MBA-DbMAC: A Random-Access MAC Protocol for MBAs

Jean-Daniel Medjo Me Biomo(✉), Thomas Kunz, and Marc St-Hilaire

Department of Systems and Computer Engineering, Carleton University,
Ottawa, ON, Canada
{jemeda,tkunz}@sce.carleton.ca, marc_st_hilaire@carleton.ca

Abstract. Ad hoc networks are infrastructureless and self-organizing networks that consist of static/mobile nodes with limited bandwidth, computing ability and energy. These networks are deployed for civilian/military applications. Having an efficient/reliable routing protocol for communication between the nodes can be critical. A current research avenue involves exploiting Multi-Beam directional Antennas (MBA) to significantly reduce the end-to-end delay in multi-hop ad hoc networks that service multiple traffic flows. To tackle such an issue at the Network level, there is a need for a suitable MAC protocol underneath. In this paper we propose MBA-DbMAC, a MAC protocol for MBAs. MBA-DbMAC is a generic MAC protocol that has the basic functionalities of a MAC protocol and renders possible the basic operation of MBA-equipped nodes in static/mobile ad hoc networks. We adopt a two-tier processing approach whereby the MAC layer is split into two artificial sub-layers: a controller sub-layer (materialized by one node-wide parent process) and a sector sub-layer (materialized by N child processes, 1 child process for each of the N sectors). Other novel aspects of this protocol are the decoupled broadcasting and the time window policy that we adopt to avoid Critical Chain Transmission/Reception. We use Opnet for the implementation/simulations. It is shown that MBA-DbMAC perfectly performs key functions such as unicasting, broadcasting, and concurrent packet transmission/reception.

Keywords: Directional MAC protocol · Multi-beam antenna · Ad hoc networks · Opnet/Riverbed

1 Introduction

As researchers are turning some attention into exploiting Multi-Beam directional Antennas (MBAs) to significantly reduce the end-to-end (E2E) delay in multi-hop ad hoc networks that service multiple traffic flows, the need for a suitable medium-access protocol is growing. In this paper we propose MBA-DbMAC, a generic MAC protocol that has the basic functionalities of a distributed IEEE 802.11 MAC protocol and renders possible the operation of MBA-equipped nodes

J. Zheng et al. (Eds.): ADHOCNETS 2019, LNICST 306, pp. 67–86, 2019.
https://doi.org/10.1007/978-3-030-37262-0_6

both in static and mobile ad hoc networks (MANETs). MBAs represent a major technology improvement over the more traditional single-beam directional antennas (SBAs) whereby only one antenna beam is active at a time. With MBAs, many beams can transmit or receive concurrently; opening up a range of possibilities for network performance improvements.

There is a choice to be made between a TDMA approach and a CSMA/CA approach. We are not going to use TDMA, but rather CSMA/CA. This is because enforcing time synchronization between nodes in MANETs can be very challenging. As noted by Luo et al. in [9], current TDMA-based MAC protocols cannot provide the rapidness and agility to deal with the rapid mobility and varying densities of vehicles in Vehicular Ad hoc NETworks (VANETs)[1] for instance. Among others, Abolhasan et al. [2] also point out the fact that CSMA/CA is a practical MAC protocol for wireless distributed network (WDNs), because it does not require time synchronization and there is no centralized coordination. CSMA/CA has been extensively implemented in WDNs.

In addition to the typical problems inherent to the design of MAC protocols in wireless networks, MBAs introduce new challenges. As already noted, one of these challenges is synchronization. To harness the Concurrent Packet Transmission (CPT) and Concurrent Packet Reception (CPR) capabilities of MBAs, there needs to be some synchronization to ensure that as many transmissions as possible happen while the concerned node is in transmission (Tx) mode before switching to reception (Rx) mode, and vice-versa. Another challenge is Head of Line (HOL) blocking. It needs to be ensured that a packet whose next hop lies in a given beam that is free (medium cleared) is not blocked in a queue behind another packet whose next hop lies in another beam that is waiting for the medium to clear out in that direction. The above challenges and a few more are addressed in the next section.

2 MBA Challenges

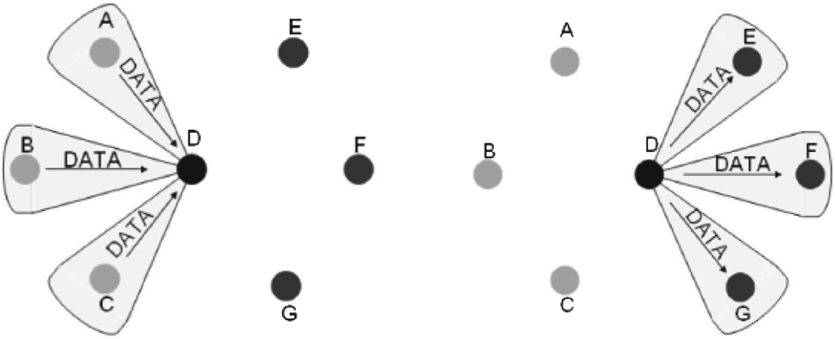
2.1 Synchronization for CPT and CPR

Transmitting nodes should start their transmission concurrently so that the common receiver node can simultaneously activate multiple beams pointing toward them. In Fig. 1, assume that nodes A, B, and C need to send data to nodes E, F, and G, respectively, via node D. In the absence of any localized synchronization, the possibility that any two or all three of them will start transmission at the same time using a random access protocol is rare. This is due to the fact that, before initiating transmission, each node waits for a random duration after sensing the channel idle for DIFS duration. Therefore, node D could start receiving a packet from node A. Before that reception is over, nodes B and C could begin their transmissions to D as well. Any of these transmitting nodes might have many packets to transmit and start transmitting those packets before the ongoing transmissions are over. This chain of events would trigger what is

---

[1] Another type of MANETs.

known as transmission starvation on node *D* which would be locked in successive receptions. To avoid such starvation, [4] proposes that CPR and CPT occur in succession at the bottleneck node (node that is common in two or more routes). To us, even without enforcing that all transmitters start their transmissions at the same time (as is the case in [4]), this scheme would translate into only accepting one packet per transmitter per cycle. That way, after having received packets from all sources, node *D* would switch to CPT and transmit them all together to their respective destinations, as shown in Fig. 1. We shall present our full strategy on this CPR/CPT issue in a subsequent section.



(a) CPR (b) CPT

Fig. 1. CPR and CPT [4]

2.2 Head-of-Line Blocking

A packet at the head of the data queue may block other packets behind it indefinitely if its intended outbound beam is busy. This phenomenon is known as Head of Line blocking [8,10]. This can be prevented by having a dedicated data queue for each beam as is the case in HMAC [4].

2.3 Deafness

In [4], the authors point out deafness as a problem, invoking beamforming. However, beamforming is not applicable to MBAs. It is only a reality for SBAs. Therefore, we believe that deafness is a non-issue in the context of MBAs. A source node experiences the deafness problem when it fails to communicate with its intended destination node that is pointing towards a different direction for transmission or reception. In fact, we should only consider the case when the intended destination is pointing to another direction for reception. That is because in the case of transmission, even if the destination was pointing to the source, there would be a mode mismatch and the packet from the source would not be received, as the destination cannot be transmitting and receiving at the same time. Now, in the case of MBAs, if the destination node is receiving from a different direction for reception, the beam that is pointing to the source is also able to receive, since the reception mode applies to the entire node (i.e. all beams) and multiple beams can receive concurrently. In SBAs, only one beam can be active at a time.

2.4 Hidden-Terminal

A given scenario experiences the hidden-terminal problem when transmissions from two nodes which cannot hear each other collide at a third node. In the context of directional antennas, all nodes that are located within the destination node's coverage area of interest (covered by a specific beam) and are away from the source node's coverage area of interest are hidden terminals. The shaded area A_h in Fig. 2 indicates the area in which hidden terminals may exist, from the perspective of node S. A node located at any other area where it cannot hear S is not a hidden terminal because, even if it points toward D, its signal either will reach D from a different beam (therefore allowed to be processed thanks to CPR capability) or it will be undetectable (because too far away) by D. Unfortunately, as pointed out in [1], the standard RTS/CTS mechanism fails to completely solve the hidden-terminal problem, as nodes in A_h may initiate transmissions to D during the time the source node S transmits the RTS to D. This problem has not been solved in current literature using a single-channel and single-radio interface [12].

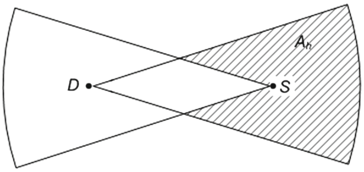


Fig. 2. Hidden-terminal problem [1]

2.5 Exposed-Terminal

A node experiences the exposed-terminal problem if it assumes a busy medium and defers its transmission even though it could be transmitting without impeding ongoing transmissions by other nodes. This is a problem mostly for omnidirectional antennas, as pointed out in [7]. In the context of directional antennas, the exposed-terminal problem can easily be solved by taking advantage of directionality. In effect, the exposed-terminal problem happens when a node receives an unintended RTS/CTS and blocks its antenna for the duration of the announced transmission, preventing the node from participating or initiating concurrent transmissions that would not impede the ongoing one. Simply using directional (per sector) NAV timers narrows the scope of this issue, since only the sectors receiving the unintended RTS/CTS will be "blocked". Notice that, even with directional antennas, there is still the possibility of an exposed-terminal if the parallel transmission falls within the blocked beam (See Fig. 3 where the transmission between C and D is prevented by the transmission between A and B that has blocked C's beam that points to D); but we assume that this is infrequent enough to not justify a separate mechanism to deal with it.

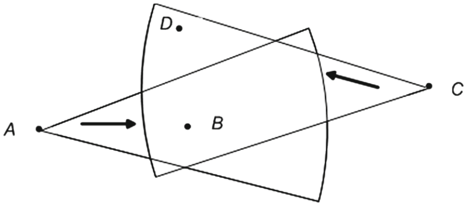


Fig. 3. Exposed-terminal problem with directional antennas

2.6 Random Backoff: Beam-Based or Node-Based

In MBA, each node has several beams, so the problem is how to control the random backoff for each beam after DIFS during transmission or retransmission attempts. One solution is to maintain a separate contention window (CW) for each beam, referred to as beam-based backoff. The other solution is to have a common CW for all beams that increases or decreases depending on the collisions or successful transmissions in the transmitting beams. This scheme is referred to as node-based backoff, where a node can transmit in multiple beams simultaneously. Now all the beams wait for the same random duration after DIFS. This is the approach adopted in HMAC and MMAC-NB [6], where the authors claim that it is conducive to high shares of CPT. This approach implies that if the medium in one beam becomes active, the backoff is suspended in all beams; which, we believe, would not be efficient. Therefore, we do not intend to pursue this approach. Instead, the beam-based backoff approach seems more appropriate.

3 Antenna Model

Similar to the approach proposed in [4], we focus on gains from spatial reuse exclusively and not from range extension of directional beams. Therefore, the range of each beam is constant and is kept equivalent to the omnidirectional range. The antenna is assumed to have a power control mechanism that feeds each beam with a power of P_{TOTAL}/M; M being the total number of beams.

Since the transmission range is kept constant, the implication is that the amount of power used will vary depending on the number of beams that are activated for a given transmission cycle. Broadcasting and multicasting (using multiple beams to send multiple different unicast packets) on all beams will consume the most power while unicasting will consume the least power. In the end, gains in energy efficiency are expected inasmuch as it is anticipated that we will not use all the beams at every single transmission cycle.

The beam shape is assumed to be a disk slice (in 2D). Sidelobes' interferences are not considered for simplicity and simulation tractability. Carrier sensing is performed directionally, that is, before transmission, the medium is sensed only in the desired beam(s). In idle mode, the receiver listens on all its beams (omni mode). A collision occurs only if a node receives interfering energy on the same beam in which it is actively receiving a packet. We use a switched-beam antenna model. A switched-beam antenna requires only activating one of the predefined beams that concentrates in the direction of the user. An adaptive beam antenna requires complex beamforming algorithms to point in the direction of the user. Thus, adaptive beam antennas are more complex to design and are not generally considered for commercial wireless networks [4]. Hence, in the remainder of this work, an MBA refers to a switched-beam antenna capable of switching multiple beams simultaneously. A node can either transmit or receive data, but not both, on multiple beams at the same time.

4 Review of Related Work: A Hybrid MAC Solution

MMAC-NB [6], ESIF [5], and HMAC [4] were all proposed by the same authors. In this section, we summarize only HMAC because it is considered an upgrade of the former two. In effect, although ESIF and HMAC deliver comparable performance, HMAC fares better; owing to its simpler design, reduced cross-layer dependence, and backward compatibility with IEEE 802.11-DCF-based protocols. The reactive mechanism for handling deafness and p-persistent CSMA employed by ESIF requires modifications in the Network layer to store the count of potential transmitters in every beam and message piggybacking among the neighboring nodes, which increases the overhead and complexity of the protocol. On the other hand, MMAC-NB has a poorer Concurrent Packet Reception capability, which leads to the underutilization of multiple beam antennas at the bottleneck (star) nodes.

As discussed in Sect. 2.3, we do dispute the existence of the deafness problem when MBAs are used. In effect, the authors in [4] claim deafness as follows. In Fig. 4, it is assumed that nodes A and B are engaged in communication, and control packets are transmitted directionally. Hence, nodes X and Y are oblivious to the ongoing communication between A and B. The claim is that they continue transmitting RTS messages to node A who is deaf to their messages and hence does not send back any CTS messages to nodes X and Y (which, consequently, go into backoff mode). In fact, in an MBA scenario, A cannot be deaf to the transmissions from X and Y. It will simply receive those transmissions on different beams at the same time as it receives the transmission from B.

HMAC is a cross-layer protocol that uses information from both the Network and the Physical layers for its operation. Similar to MMAC-NB and ESIF, HMAC uses a separate queue for each beam to avoid HOL blocking. It also uses a scheduling (SCH) control packet, which is sent in all desired beams other than the ones being negotiated via RTS/CTS. The purpose of these additional SCH packets is to further mitigate the deafness problem by letting potential transmitters know that the current node is pointing somewhere else. The novel features

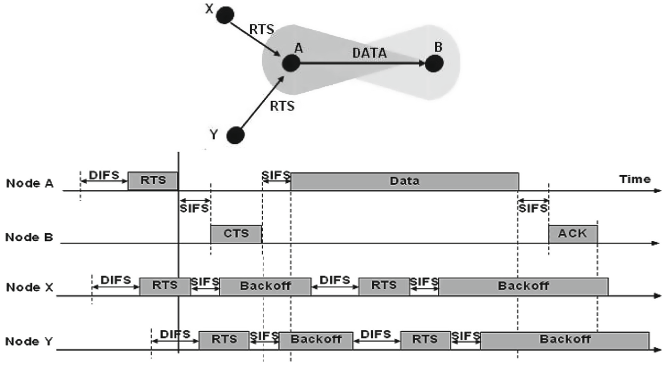


Fig. 4. Illustration of deafness as per [4]

of HMAC include: its channel access mechanism, algorithms for mitigating deafness and contention resolution, as well as jump backoff and role priority switching mechanisms for enhancing throughput. HMAC is also backward compatible with IEEE 802.11 DCF.

To minimize queuing delays in the network by facilitating successive cycles of CPR and CPT, a mechanism similar to hot-potato routing [3] is installed at every node. Thus, depending on the packets in its buffer, a node switches between transmitter and receiver modes. As long as a data packet exists in the queue, the node gives priority to the transmission mode; otherwise, the reception mode supersedes. Moreover, depending on the available neighbor and beam schedules, a node can determine whether it can actually initiate data transmission. If not, the reception mode gets priority.

5 Our Solution: The MBA-DbMAC Protocol

The IEEE 802.11b DCF MAC protocol (omnidirectional) is contention-based. It employs a CSMA/CA mechanism by means of the DCF (Distributed Coordination Function). In [11], this standard MAC protocol was adapted to work with single-beamed directional antennas. The adapted MAC protocol was called the IEEE 802.11b-based Directional MAC (or DbMAC) protocol. DbMAC works mostly like the standard IEEE 802.11b DCF MAC protocol but on a per-antenna-sector basis. At this juncture, we are going to adapt DbMAC to work with MBAs. We call this new protocol the MBA-DbMAC protocol. This new protocol is going to adopt, to a certain extent, some of the approaches of HMAC (thus of MMAC-NB and ESIF as well) to solve some of the MAC protocol challenges that come with the introduction of MBAs as discussed earlier. In the spirit of DbMAC versus existing single-beam directional MAC protocols, MBA-DbMAC combines and/or slightly modifies the common/key features of existing MBA MAC protocols without much of a claim of being a superior protocol. Nevertheless, we do propose a novel decoupled broadcasting scheme whose goal is to give all the beams a chance to transmit when some are not ready during the first broadcasting attempt. The details of MBA-DbMAC are summarized in the next few paragraphs.

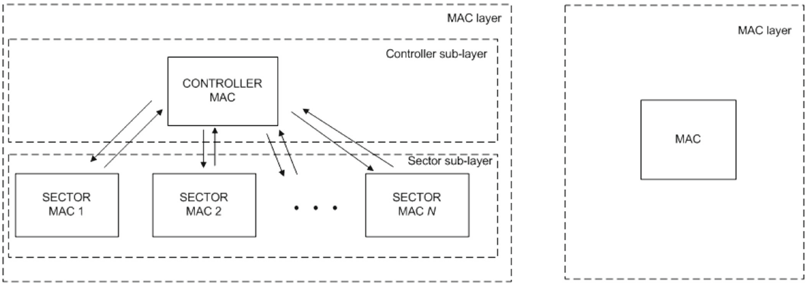
5.1 Design

For the design of the MBA-DbMAC protocol, we adopt a two-tier processing approach. In effect, we split the MAC layer into two artificial sub-layers: the controller sub-layer and the sector sub-layer. Figure 5 depicts the contrast between this approach and the traditional single-tier approach. In practice, in the development environment that we use (OPNET), these two sub-layers are materialized with processes: each node has one controller (parent) process that spawns N identical sector (child) processes at run-time; N being the number of antenna sectors. Traditional MAC-layer mechanisms are applied at the sector sub-layer. The controller sub-layer manages: (a) the neighbor table, (b) the assignment of a high-layer packet to the appropriate sector, and (c) the switch between the different operation modes (Tx mode, Idle mode, and Rx mode). The operation mode applies to the whole node. For instance, if a given sector is permitted to transmit at a given time, it is the job of the controller process to instruct all the other sectors via their respective processes to now switch to Tx mode, regardless of whether or not there are packets to be transmitted by these other sectors. Our two-tier design approach is an elegant way of enforcing mode switches in the node while still leaving the full autonomy of medium access to sectors in their respective direction of competence.

5.2 Ensuring CPT and CPR

Unlike DbMAC, we now allow multiple packets to be received or transmitted at the same time. This also means ensuring the processing of such packets by the node, as long as this is happening on different beams. The reception of two or more packets on the same beam at the same time is considered a collision. Likewise, the transmission of two or more packets on the same beam at the same time is forbidden.

We do not enforce that all transmissions on all beams of interest start at the exact same time. However, once a transmission starts on a given beam, the node is now in transmission mode (no reception can occur in any beam at this point). Any other beam that is scheduled to start a transmission while the first is still going on can start its transmission, provided a certain condition that we discuss later in this paragraph. The transmission mode ends when all beams have ended their respective transmissions. To ensure fairness and avoid reception starvation, we limit each beam to only transmit one packet per cycle. Backoff and IFS decisions are made on each beam independently of the others. This means we opt for a beam-based backoff approach, unlike HMAC and ESIF. Once the transmission mode ends, the node goes to Idle mode, or goes to reception mode if a signal is sensed right at that point in time. The node can also start a new transmission cycle if there is no signal sensed and there are packets ready to



(a) Two-Tier MAC Approach (b) Traditional Single-Tier MAC Approach

Fig. 5. Two-tier MAC approach vs traditional single-tier MAC approach

be sent instead; with the medium sensed idle long enough (has been clear for an IFS duration). We call Critical Chain Transmission (CCT) a situation where all antenna sectors/beams start transmitting one right before the end of the transmission at another sector. All the sectors are therefore stuck in transmission mode until the last sector of the chain has finished, even though the first transmitting sector has finished its transmission a while ago. CCT might result in spending a disproportionate amount of time in one transmission mode that quickly becomes useless to most sectors. This will then incur some MAC delays. To avoid CCT, we set up the transmission mode such that any sector can only start its transmission within a certain time window after the first sector has started its transmission. Past this time window, no transmission is allowed to start until the next transmission cycle (after the current transmission mode has completed). We set the time window to be half the anticipated transmission time of the first transmitting sector.

We ensure CPR using the same philosophy as with CPT. A node simply stays in Rx mode until all concerned beams have completed their respective receptions. The rule of one packet per beam per cycle also applies to avoid transmission starvation if there are too many packets coming in. Once a given beam has finished receiving one packet, it deactivates itself from further reception for the remainder of that Rx cycle. All packets (unicast and broadcast) arriving during this deactivation period will be lost. The senders of those unicast packets will have to retry sending if applicable. The Rx mode ends when all beams that have sensed a packet before the end of the first reception finish their respective receptions. The node then switches to Tx mode if there is a packet in any of the queues and the medium is ready to be used in the concerned direction. If there is no packet ready to be sent (medium also ready/available), the node goes into Idle mode or a new Rx mode if an incoming signal is sensed right at that time. From Idle mode, a node can go into Tx or Rx mode, depending on which one occurs first: a packet ready to be transmitted and the medium free, or an incoming packet sensed at the Physical layer. We choose to not adopt the approach to synchronize potential transmitters, because that would be a difficult and overhead-inducing undertaking. Similar to the CCT, the Critical Chain Reception is a situation where antenna sectors/beams start receiving a packet right before the end of the reception at another sector. We similarly adopt a time window policy to avoid CCR. In effect, we set up the reception mode such that any sector can start reception only within a certain time window after the first sector has started its reception. Past this time window, no reception is allowed to start until the next reception mode in the next cycle (after the current reception mode has completed). We set the time window to be half the anticipated reception time of the first receiving sector. In this case, the anticipated reception time is approximated to the time it would take the current node to transmit a packet the size of the RTS-threshold (the threshold for performing RTS/CTS frame exchange preceding the transmission of the data frame.) to the neighbor.

5.3 Switch from Tx to Rx Mode

As already mentioned, we set Rx and Tx modes to alternate, with one packet reception/transmission allowed per sector per cycle. The default mode is the Idle mode, where a node is neither transmitting nor receiving on any of its sectors. The switch to Tx/Rx mode is interrupt-driven. In the case of the switching-on of the Tx mode, a node will know when a packet is passed down the protocol stack or has been received and now has to be forwarded. The node checks the medium on the beam of interest. When the medium is ready (after SIFS/DIFS and backoff waiting if applicable) for transmission, a switch to Tx mode occurs, provided that the node is in Idle mode. The node stays in this mode until the transmission is complete and there is no other transmission in progress on any other beam. It then switches back to Idle mode. From the Idle mode, the node can now switch to Rx mode when an incoming packet is sensed by any sector. If no incoming packet signal is sensed at the Physical layer and there is a packet to be transmitted and the medium is free, then a switch to Tx mode occurs. As already mentioned, each node features a controller process that instructs the switch to Rx, Tx, or Idle mode to all the sectors. The state diagram of the controller is depicted in Fig. 6 and the state diagram of the sector/beam is shown in Fig. 7. For instance, while in Idle mode, if an incoming packet is sensed on a sector, that sector starts the reception and immediately reports it to the controller process. The controller process then instructs all the other sectors to switch to Rx mode. At this point, only receptions are allowed. Similarly, while in Idle mode, if the medium becomes available in a certain direction where a packet is waiting to be sent, the concerned sector starts the transmission and informs the controller. The controller then instructs all the other sectors to go into Tx mode. These other sectors can now start their own transmissions, if any, as long as they start within a certain time window as discussed earlier.

Once in Tx mode, all the sectors involved have different transmission end times. The node ends its Tx mode when the latest transmission end time occurs. The same applies to the Rx mode and the different reception end times.

In the special case of a Backoff period ending (for a transmission to start) at the same time as a packet reception starts on a given sector, preference is given to the transmission, and the packet reception is aborted.

5.4 Per-Beam Queues

As is the case for ESIF and HMAC, we set up a separate packet queue for each beam. This prevents HOL blocking. When a higher layer packet arrives at the MAC level, the controller sub-layer, which is hosted by the parent process, checks its address, and forwards it to the appropriate child process (at the sector sub-layer) that then queues the packet on its own queue. For that to be feasible, the parent process maintains a neighbor table as we explain in Sect. 5.8.

5.5 Broadcasting

Broadcasting is performed in a “decoupled” manner as follows. The controller sub-layer makes $N-1$ copies (N being the number of sectors) of the packet to be broadcast. The controller forwards each copy to a different sector process. The latter places the packet copy on its queue. That packet is eventually transmitted when it is its turn (as there might be some other packets ahead of it in the queue) and the medium is free in the concerned direction. No RTS/CTS or IFS are required. With this scheme, we can have broadcast packets be transmitted on some beams while unicast packets are transmitted on other beams during a given Tx mode. We call it “diversity-casting”, and it is a novelty of MBA-DbMAC. The mechanism is different than in DbMAC since there is no beam-sweeping involved here, and each beam has its own queue. Moreover, in DbMAC, we provided the nodes with the ability to attempt broadcast packet transmissions twice in a given direction (if the medium was found to be busy the first time around). We do not keep this feature in MBA-DbMAC. Rather, as mentioned, each beam sends its broadcast packet copy independently whenever the medium clears in that direction. With diversity-casting, there is no giving up after one retry as is the case in DbMAC. Therefore, we ensure that all broadcast packets are sent in all directions, unless the normal max retry limit (same as for unicast packets) is reached on a given direction. This should, among others, have a positive impact at the Network layer with route discoveries. The authors of MMAC-NB, ESIF, and HMAC do not specify how broadcasting is handled. But, judging from the emphasis they place on enforcing strict CPT, we can only assume that broadcasting is not decoupled, and that it is done in a synchronous manner; with all the delays that this may impose when confronted with busy medium in certain directions.

The cost of diversity-casting in terms of energy is the same as for broadcasting with omnidirectional antennas when we set the transmission range of the MBAs to be the same[2] as for omnidirectional antennas. MBAs, however, allow spatial reuse. Moreover, diversity-casting ensures that the packet is sent to a specific

---

[2] For fairness in comparison.

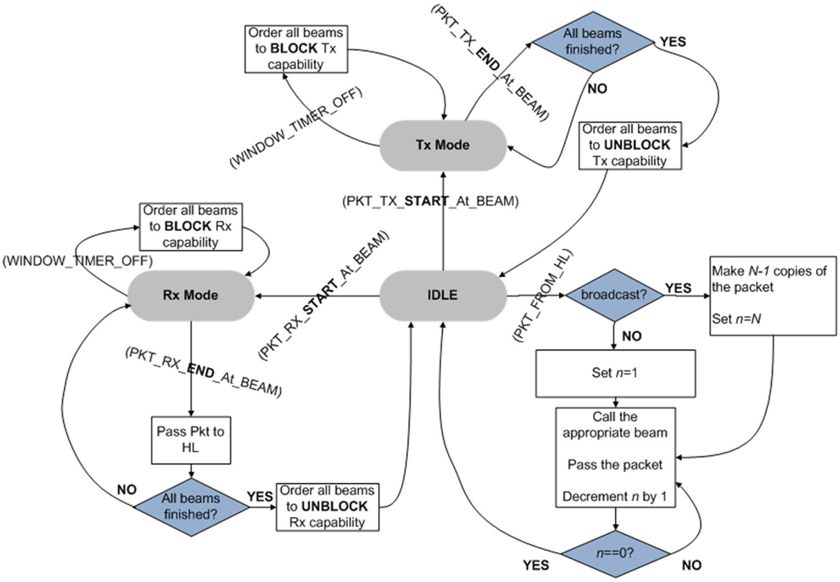


Fig. 6. Controller/parent diagram

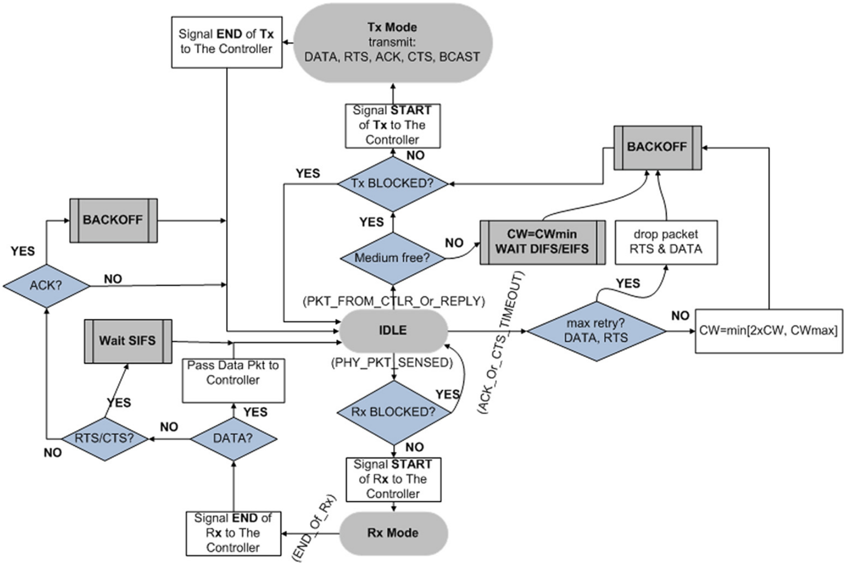


Fig. 7. Beam/child diagram

direction as soon as possible instead of waiting for all directions to be clear. This should have a positive impact in delay reduction.

5.6 RTS/CTS

Similar to DbMAC, RTS and CTS are sent directionally. As the controller sublayer forwards the packet to the appropriate sector process, the latter takes care of it (including the RTS/CTS exchanges) in its own direction, independently from the other sectors/directions. This mechanism combats the hidden-terminal problem. Moreover, having the RTS/CTS exchange occur directionally solves another problem that might arise in the context of MBAs. A source node might be engaged in an RTS/CTS exchange with a given neighbor in one direction/beam. Another neighbor that can reach this source node on a different beam does not need to know about this exchange since it can engage in a parallel exchange with the same source node thanks to the CPR capability. Having RTS/CTS exchanged omnidirectionally would unnecessarily block exchanges with neighbors from other beams. This can be seen as an exposed-terminal problem in reverse, specific to MBA environments. Restricting RTS/CTS exchanges to occur directionally solves this potential problem.

5.7 DNAV

The Directional Network Allocation Vector (DNAV) is similar to the NAV described in the original IEEE 802.11b DCF MAC; except now the NAV is kept on a per-antenna-beam basis. The exposed-terminal problem is taken care of through the DNAV. Only the sector that receives the unintended RTS/CTS will be "blocked" for transmission for the duration of the neighboring/overheard transmission. Any other sector of the neighbor can engage in a concurrent transmission/reception. As already noted, there is still the possibility of an exposed-terminal if the parallel transmission falls within the blocked beam (Fig. 3). But we assume that this is infrequent enough to not justify a separate mechanism to deal with it. For instance, assuming an N-sector antenna, assuming that the nodes are uniformly distributed in the network, another node (the one sending the unintended RTS/CTS) has to be in the same sector as the one we would transmit in. That cuts down the probability of this occurring by a factor of $1/N$. And the impacted node would need to, presumably, use the impacted sector for its transmission, and that would cause another $1/N$ reduction. Moreover, the sender of the RTS/CTS would also have to use a specific sector (out of its own N) that points to the impacted node. That is yet another $1/N$ reduction. In the end, we can see that would imply a reduction by $1/N^3$ of the probability of occurrence of the exposed-terminal problem. Therefore, that probability drops considerably with an increase in the number of sectors N.

5.8 Neighbor Table

Broadcasting is an important and frequently exploited communication primitive. In the broadcasting mode, nodes send packets on all sectors without having to know where neighbors are. In fact, numerous network layer protocols (routing) do perform neighbor discovery through periodic HELLO message exchanges which are one-hop broadcast messages. Moreover, on-demand protocols broadcast Route Discovery messages before sending unicast packets. Broadcasting messages offer the opportunity to a node to "advertise" itself to all the nodes within its transmission range, and hence be included in their neighbor tables.

Similar to DbMAC, each node maintains a neighbor table. This table is maintained by the controller process with updates coming from individual sector processes. The neighbor table is initially empty and is progressively populated as follows. Whenever a sector receives a frame from the Physical layer, the process responsible for that sector informs the controller sub-layer of the address of the sender of that frame. The controller then updates the neighbor table, if needed, with information that says "Neighbor A lies on beam i". That way, if a frame is later to be sent to that neighbor, the node (controller) knows which antenna sector to use. Therefore, we do not have to calculate the direction of arrival; we know it as soon as the frame arrives on a given sector (that represents a good-enough approximate direction).

6 MBA-DbMAC Functionality Testing

We performed functionality checks of uni-packet transmission (UPT) and broadcasting, and they all worked perfectly as expected. In this section, we show the tests on concurrent-packet transmission and concurrent-packet reception. Furthermore, we examine the time spent in transmit mode and in receive mode. Finally, we assess the performance of our protocol against that of IEEE 802.11b (omnidirectional antenna) and DbMAC (single-beam directional antenna). This performance is evaluated in terms of delay, throughput, and goodput. For these tests, we use the static star-topology scenario depicted in Fig. 8 where node 1 is the central node and all other nodes are peripheral. The default data packet generation rate is set to 5 packets/sec. However, we will increase (and specify) this rate occasionally for some of the tests. All the tests are conducted using OPNET/Riverbed Modeler 16.0. The channel capacity is 11 Mbps, and the default simulation time is 1800 s.

6.1 CPT-CPR Test

To test CPT-CPR, node 1 sends and receives unicast data packets from nodes 2, 3, 4, 5, 6, and 7. These nodes all start sending data packets to node 1 around the same time. For the CPT functionality, the results are as shown in Fig. 9.

The first sub-graph shows that a little over 5 packets per seconds are sent from node 1. This rate is actually the rate at application-layer level. However, in order

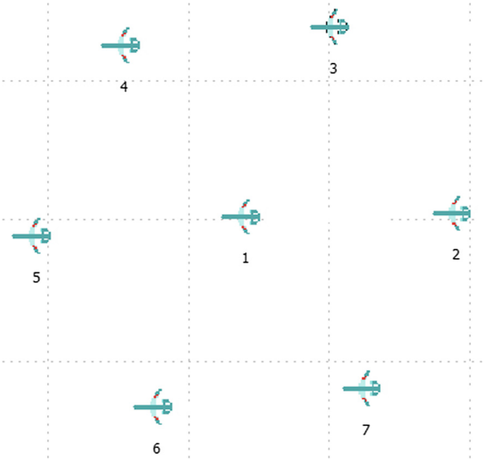


Fig. 8. MAC test scenario topology

to test CPT at MAC level, five copies are made for each application-layer packet (making it a total of 6 packets to be sent at MAC level, one for each sector). For each copy, a different destination address (corresponding to a different neighbor) is set, and the packet is sent via the sector that points to the intended neighbor. Therefore, the application-layer rate becomes the per-sector rate at the MAC level. We implemented this scheme this way because OPNET does not allow us to issue application-layer-level packets with different destination addresses. The remaining sub-graphs clearly show that all nodes, each lying on a different beam of node 1, receive 5 packets per second sent by node 1 on the concerned beam. This is the first step in showing the CPT functionality of MBA-DbMAC.

For the CPR functionality, we obtain the results shown in Fig. 10. The first sub-graph shows that about 35 packets per seconds are received by node 1. The remaining sub-graphs show that nodes 2, 3, 4, 5, 6, and 7 each send a little over 5 packets per second; thus the total of about 35[3] packets received by node 1. This is the first step in showing the CPR functionality of MBA-DbMAC.

6.2 Rx Mode vs Tx Mode

We measure the time spent in Rx and Tx mode in the situation of CPT/CPR. The traffic rate is still 5 packets per second. For a peripheral node, the time spent in Rx mode (18 s) and in Tx mode (18 s) is balanced since only one beam is involved. However, for the central node (node 1), we obtain that more time is spent in Rx mode (79 s) than in Tx mode (48 s). This imbalance can be explained by the fact that, with our current rate of packet generation (5 packets

---

[3] Each of the 6 senders sends a little over 5 pkts/sec, more like a little under 6 pkts/sec, therefore the total received at the common receiver is greater than 30 packets, but still less than 36 packets (as it would be if the senders' rate were exactly 6 pkts/sec).

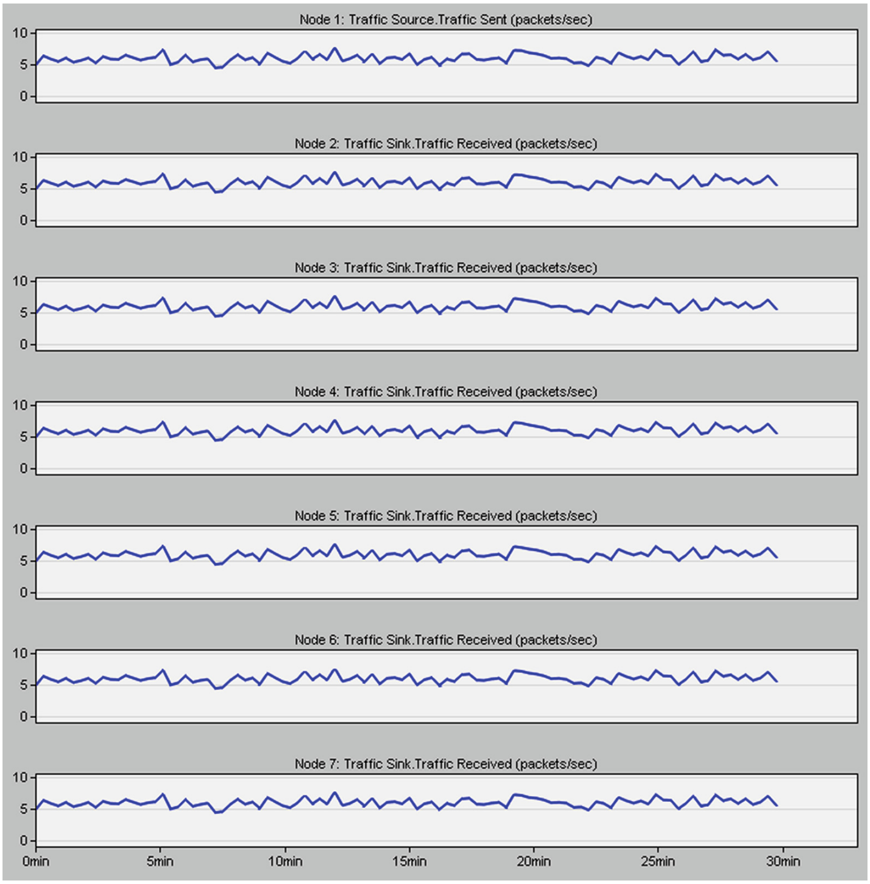


Fig. 9. CPT test

per second), there are more instances of the central node not having packets to transmit than there are instances of at least one of the six peripheral nodes not having packets to transmit (hence the central node not receiving).

One way to address/overcome this asymmetry is to increase data packet generation so that the output buffer of all the nodes is always full. For example, with a higher packet generation rate of 20 pkts/sec, we obtain a more symmetric result for the central node of 304 s spent in Rx mode and 298 s spent in Tx mode.

These time results are the final step in showing that we do, in fact, have CPR/CPT functionality. If the 6 peripheral nodes spend collectively 6×18 s to transmit, that is 108 s. But, as we saw earlier, the central receiver (node 1) spends only 78 s receiving. This means that some of the receptions need to happen in parallel; because, after all, all packets are received, as we saw in Fig. 10. This therefore shows the CPR functionality of MBA-DbMAC. Similarly, if the central node sent its packets sequentially (one beam at a time) it would behave as a

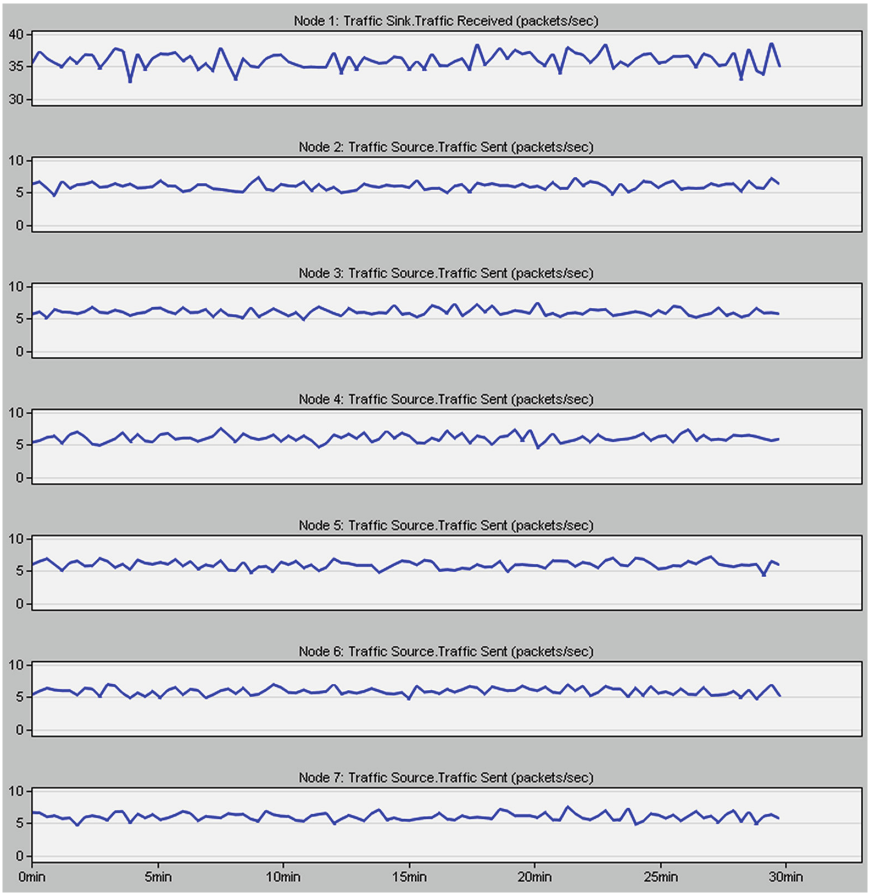


Fig. 10. CPR test

peripheral node (single beam used) in that regard; hence it would take a total of about 18 × 6 = 108 s to complete its transmissions. But, as seen earlier, it only takes 47 s, with no packet lost (Fig. 9); which means that some transmissions have been done in parallel. This therefore shows the CPT functionality of MBA-DbMAC.

6.3 Performance Comparison: Delay, Throughput, and Goodput

With the same star topology, we compare the MAC delay, throughput, and goodput when three types of antennas are used: MBAs, SBAs, and omnidirectional antennas. In any given scenario, all nodes are equipped with the same antenna type. MBAs work with the MBA-DbMAC protocol, SBAs work with the DbMAC protocol, and omnidirectional antennas work with the IEEE 802.11b DCF MAC protocol.

The delay here is measured from the time when a packet is inserted into the transmission queue until it is received. It includes the period for the successful RTS/CTS exchange, if this exchange is used prior to the transmission of that frame.

The network throughput is defined as the total number of bits (in bits/sec) forwarded from wireless LAN layers to higher layers in all the WLAN nodes of the network. The network goodput is defined as the application-level throughput (i.e. the number of useful information bits delivered by the network to a certain destination per unit of time). The amount of data considered excludes protocol overhead bits as well as retransmitted data packets.

The performance in terms of the delay comes as follows: the smallest delay is experienced with MBA antennas, 0.5 ms, compared to 3.5 ms for Omni, and 5.5 ms for SBA antennas. MBA antennas reduce the delay by about a factor of 10. The delay of SBA being significantly worse than Omni illustrates that deafness is a more severe issue than the exposed-terminal problem in this configuration. In effect, we know (from [11]) that SBAs are more affected by deafness than any of the other two types, whereas omnidirectional antennas are more affected by the exposed-terminal problem.

The throughput showed to be similar for all three types of antennas. This can be explained by the fact that the nodes spend most of their time in Idle mode (no transmission, no reception). For instance, in the case of MBA-DbMAC testing, we saw earlier that the central node spends a total of about 127 s in transmission and reception combined (out of 1800 s of simulation). Peripheral nodes spend about 36 s out of 1800 in transmission and reception combined. These times in active mode are expected to be higher in the case of SBA and omni antennas (since there is no concurrence of multiple transmissions or receptions). But since we still receive all the sent packets notwithstanding the type of antenna used, it makes sense to have the same throughput.

If we increase the packet generation rate to 105 pkt/sec, we now have a total network load of at least 105 pkt/sec × 1024 Bytes/pkt × 8 bits/Byte × 12 nodes[4] = 10.3 Mbps (recall that the channel capacity is set to 11 Mbps). Throughput-wise, we observed that SBA fails completely with this high packet rate, and that MBA has a lower throughput than Omni. However, MBA actually shows a higher goodput (actual data packets received at the peripheral nodes); even though, in the absolute, both types of antennas (and their corresponding MAC protocols) perform poorly with packet delivery ratios (goodput) of 40% and 50% respectively.

7 Summary

The proposed MBA-DbMAC protocol is a generic MAC protocol that has the basic functionalities of a MAC protocol and renders possible the basic operation of MBA-equipped nodes. MBA-DbMAC is an adaptation/extension of DbMAC

---

[4] 12 nodes = 6 peripheral nodes + 1 central node. However, the central node acts as 6 sources (each of its 6 sectors sends traffic to a neighbor) of traffic.

to the MBA environment. ESIF, HMAC, and MMAC-NB are all MAC protocols designed to work with MBAs. We used the design of these protocols as a starting point in the identification of MBA-specific issues inherent to MAC design. We have adopted some ideas from these protocols, but we are also proposing different, innovative, and fairly simple solutions to some MBA-specific issues. For instance, for the design of the MBA-DbMAC protocol, we adopt a two-tier processing approach whereby the MAC layer is split into two artificial sub-layers: the controller sub-layer (materialized by one node-wide parent process) and the sector sub-layer (materialized by N child processes, 1 child process for each of the N sectors). The sector sub-layer implements traditional MAC-layer mechanisms while the controller sub-layer manages: (a) the neighbor table, (b) the assignment of a high-layer packet to the appropriate sector, and (c) the switch from/to the different operation modes (Tx mode, Idle mode, and Rx mode). This two-tier design approach is an elegant way of enforcing mode switches in the node while still leaving the full autonomy of medium access to sectors in their respective direction of competence. Other novel aspects of our solution are the decoupled-broadcasting or diversity-casting and the time window policy that we adopt in order to avoid Critical Chain Transmission and Critical Chain Reception. In the end, our goal was to have a functional MBA-specific MAC protocol.

References

1. Abdullah, A.A., Cai, L., Gebali, F.: DSDMAC: dual sensing directional MAC protocol for ad hoc networks with directional antennas. IEEE Trans. Veh. Technol. **61**(3), 1266–1275 (2012). https://doi.org/10.1109/TVT.2012.2187082
2. Abolhasan, M., Lipman, J., Ni, W., Hagelstein, B.: Software-defined wireless networking: centralized, distributed, or hybrid? IEEE Netw. **29**(4), 32–38 (2015). https://doi.org/10.1109/MNET.2015.7166188
3. Acampora, A.S., Shah, S.I.A.: Multihop lightwave networks: a comparison of store-and-forward and hot-potato routing. IEEE Trans. Commun. **40**(6), 1082–1090 (1992). https://doi.org/10.1109/26.142798
4. Jain, V., Gupta, A., Agrawal, D.P.: On-demand medium access in multihop wireless networks with multiple beam smart antennas. IEEE Trans. Parallel Distrib. Syst. **19**(4), 489–502 (2008). https://doi.org/10.1109/TPDS.2007.70739
5. Jain, V., Gupta, A., Lal, D., Agrawal, D.P.: A cross layer MAC with explicit synchronization through intelligent feedback for multiple beam antennas. In: GLOBECOM 2005. IEEE Global Telecommunications Conference, vol. 6, pp. 3196–3200, November 2005. https://doi.org/10.1109/GLOCOM.2005.1578365
6. Jain, V., Gupta, A., Lal, D., Agrawal, D.P.: IEEE 802.11 DCF based MAC protocols for multiple beam antennas and their limitations. In: IEEE International Conference on Mobile Adhoc and Sensor Systems Conference, pp. 8–474, November 2005. https://doi.org/10.1109/MAHSS.2005.1542833
7. Jayasuriya, A., Perreau, S., Dadej, A., Gordon, S.: Hidden vs. exposed terminal problem in ad hoc networks. In: Proceedings of the Australian Telecommunication Networks and Applications Conference (2004)

8. Kolar, V., Tilak, S., Abu-Ghazaleh, N.B.: Avoiding head of line blocking in directional antenna MAC protocol. In: 29th Annual IEEE International Conference on Local Computer Networks, pp. 385–392, November 2004. https://doi.org/10.1109/LCN.2004.30
9. Luo, G., Jia, S., Liu, Z., Zhu, K., Zhang, L.: sdnMAC: a software defined networking based MAC protocol in VANETs. In: 2016 IEEE/ACM 24th International Symposium on Quality of Service (IWQoS), pp. 1–2, June 2016. https://doi.org/10.1109/IWQoS.2016.7590403
10. Luo, Q., Wang, J., Liu, S.: AeroMRP: a multipath reliable transport protocol for aeronautical ad hoc networks. IEEE Internet Things J. **6**(2), 3399–3410 (2019). https://doi.org/10.1109/JIOT.2018.2883736
11. Medjo Me Biomo, J.D., Kunz, T., St-Hilaire, M.: Exploiting multi-beam antennas for end-to-end delay reduction in ad hoc networks. Mob. Netw. Appl. **23**(5), 1293–1305 (2018). https://doi.org/10.1007/s11036-018-1037-8
12. Subramanian, A.P., Das, S.R.: Addressing deafness and hidden terminal problem in directional antenna based wireless multi-hop networks. Wireless Netw. **16**(6), 1557–1567 (2010). https://doi.org/10.1007/s11276-008-0138-x