SpotMAC: A Pencil-Beam MAC for Wireless Mesh Networks

Kwan-Wu Chin
School of Electrical, Computer and Telecommunications Engineering
University of Wollongong
Northfields Avenue, NSW, Australia
kwanwu@uow.edu.au

Abstract— **Deafness is a key problem. It erodes the performance gains provided by directional antennas, and introduces a new hidden terminal problem. To address deafness, and hence the hidden terminal problem, we propose SpotMAC. By exploiting narrow or pencil beams, SpotMAC achieves high spatial reuse, throughput and fairness. In addition, pencil beams simplify the collision avoidance process and constrain the hidden terminal problem to a linear topology which can be solved using an inverse RTS/CTS exchange. From extensive simulation studies, we confirm nodes using SpotMAC have several orders of magnitude higher throughput than those using the IEEE 802.11 MAC with omni-directional antenna.**

I. Introduction

A promising but challenging approach to increase mesh networks capacity is to equip nodes with directional antennas. Nodes using such antennas have the ability to focus their transmission energy or electromagnetic beam on a specific node. This reduces interference and increases signal quality since most energy is constrained to a given geographical area. As a result, nodes experience higher per-node throughput and less interference than those using omni-directional antenna.

Despite its many advantages, directional antennas introduce a set of challenging problems [12][16]. One of which is deafness, and the resulting new hidden terminal problem. To date, various approaches have been proposed, ranging from new signaling techniques to physical layer solutions. Unfortunately, these approaches either work only for a restricted set of scenarios, require additional channels, a dedicated time period, or they involve a substantial amount of signaling overheads.

In this paper, we propose SpotMAC, a MAC that exploits narrow or pencil beams to minimize the impact of the hidden terminal problem. We show the benefits of pencil beams and how their use simplify the collision avoidance process. More important, pencil beams restrict the occurrence of the hidden terminal problem to a linear topology, which can be solved using an inverse RTS/CTS exchange. Lastly, nodes using pencil beams are less affected by increasing node density and the performance anomaly problem [2].

This paper is organized as follows. We first discuss the new hidden terminal and deafness problem in Section II, followed by existing solutions in Section II-B. We then present our system model and the protocol details of SpotMAC in Section III and IV respectively. Then in Section V, we present our simulation environment. Section VI presents results from our studies on beamwidth, node density, fairness, deafness and hidden terminals. Section VII concludes the paper.

II. Background

A. The Problems

Deafness leads to two main problems, (i) non-optimal contention window, and (ii) hidden terminals. Consider Figure 1(a). Assume Node-B is communicating with Node-C at time t. Then at time $t+1$, Node-A initiates a communication with Node-B. Notice that Node-A is unaware node-B is busy. As a result, it deems there is contention and backs off accordingly. However, once node-B finishes, it is likely to re-capture the channel again due to its low backoff value. Unfortunately, once node-A is ready to transmit, it finds node-B to be busy again. To overcome the perceived "contention", node-A increases its contention window to a bigger value. From this example, it is clear that a packet will have to undergo multiple retransmissions and will be dropped after reaching the maximum number of retransmissions.

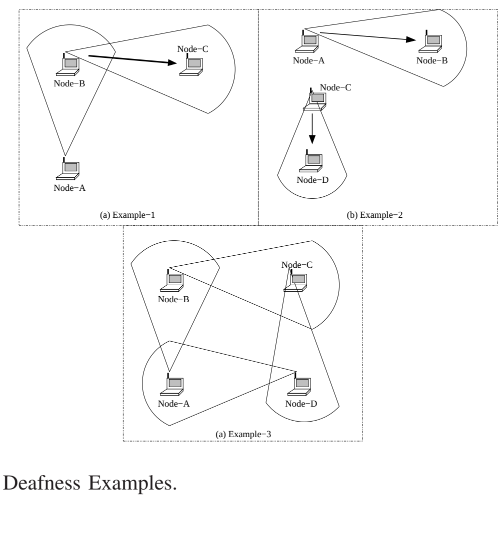

Fig. 1. Deafness Examples.

Figure 1(b) shows a different scenario where the node pairs A-B and C-D are actively communicating. Assume at time t, Node-A's communication ends and it decides to initiate a connection to Node-D. Since Node-A had its antenna pointed at Node-B previously, it is unaware Node-C has initiated communication with Node-D. It therefore concludes the channel is free to transmit. As a result, any communication attempt by node-A will result in a collision at node-D.

Deafness also causes deadlock. Consider Figure 1(c). In this example, all nodes are pointing their antenna at their respective target destination and are unaware of communication attempts made by another neighbor. As a result, every RTS sent is left un-answered, resulting in exponential backoffs. The behavior repeats itself after each node returns from backoff, thus creating a deadlock. This can be avoided if nodes switch to omni-directional mode when they are in backoff mode to receive incoming RTS packets [3].

B. Existing Solutions

The fundamental cause of the hidden terminal problem is inconsistent state. This is because when a sender is engaged in directional communication, its neighbors may transition to the transmission, reception or idle state without it knowing.

To date, many approaches have been devised to ensure all nodes have a consistent channel state. In general, these approaches are based on (i) modified RTS/CTS exchange, sender or receiver initiated, (ii) time division, or (iii) multiple channels/antennas. In this paper we omit works related to time division since SpotMAC is a contention-based MAC.

1) RTS/CTS:

Sender-Initiated: Nasipuri et al. [15] propose to send RTS and CTS messages omni-directionally, and transmit data and acknowledgment packets directionally. In [11], Ko et al. propose two schemes. The first sends CTS omni-directionally, whereas the second sends RTS directionally. Omni-directional transmission is used if the receiver's location is unknown. In both works, sending control messages omni-directionally sacrifices spatial reuse and erodes the performance benefits of directional antennas [20]. In [4], the authors propose directional RTS and CTS, and directional network allocation vector (NAV). Unfortunately, all these works suffer from the hidden terminal problem since no effort is placed on ensuring nodes have a consistent state.

Korakis et al. [12] propose a MAC that transmits RTSs circularly over all M beams. Each node that receives a RTS records the beam which the RTS arrives on, and blocks itself from transmitting on that beam for a given duration; as specified in the RTS message. The receiver then sends a CTS message directionally to the sender. Although the MAC exploits the increased range of directional antennas, its main limitation is the non-negligible delays and costly signaling overheads incurred by the RTS sweep.

The above works assume all neighboring nodes are idle to receive RTS or CTS messages, and hence block themselves from transmitting in the direction which a RTS or CTS originated from. This assumption is unrealistic given that a neighbor may be deaf when a RTS or CTS message is sent; thereby resulting in the hidden terminal problem.

Receiver-Initiated: Lal et al. [13] present a receiver initiated MAC where nodes send a RTR (ready to receive) message omni-directionally to inform neighbors whenever it is free to receive. Neighboring nodes transmit their RTS message directionally if they have a packet pending for the receiver. In [19], the authors propose a hybrid approach which involves the sender switching to the receiver-initiated mode after failing to receive a CTS after n RTS transmissions. Once communication is established, the receiver polls the sender using CTS messages.

To some extent, the hidden terminal problem can be solved using a receiver-initiated approach. The premise is that collisions happen at the receiver, and only the receiver knows its instantaneous channel status. This means, if a receiver is not ready, a sender will not attempt communication. For example, in Figure 1(b), Node-A will not initiate a connection to Node-D, thus avoiding collision with the packet from Node-C. Unfortunately, the invites, e.g., RTR, sent by nodes cause collisions. For example, in Figure 1(b), an invite message sent by node-A will cause a collision at node-D. Therefore, the hidden terminal problem remains unsolved.

2) Multi-Channel/Antennas: One approach to avoid deafness is allowing nodes to simultaneously transmit and receive [21]. Alternatively, a signal can be transmitted omni-directionally on an out-of-band channel.

Choudhury et al. [3] propose ToneDMAC. A node transmits a tone after a directional transmission to inform its neighbors that deafness was the reason for not replying to their RTSs. After receiving the tone, neighbors that have sent a RTS to the node reset their backoff counter. Hence, ToneDMAC addresses the unnecessary backoff caused by deafness. However, the hidden terminal problem is not addressed.

Elbatt et al. [5] highlighted the fact that deaf nodes may cause collisions since they are not aware of RTS/CTS messages sent by their neighbors when they are transmitting or receiving. To overcome this problem, Elbatt et al. propose to transmit RTS/CTS message on a separate channel in the direction of a blocked beam to inform idle nodes to set their NAV. On the othe hand, for active users, the authors propose that, (i) a node suspends its current data transmission and sends a RTS/CTS message when a neighbor has finished transmission, or (ii) to equip nodes with multiple directional antennas so that a RTS or CTS message can be sent during transmission to a neighbor that has just finished transmission or reception.

In [18], the authors propose BeamMAC. A sender first transmits an announcement directionally on a signaling channel to test whether an impending data transmission is permitted. Nodes that deem the data transmission to be destructive to its reception sends an objection to the sender. Alternatively, a node can determine whether a higher coding gain can be used to overcome the interference before sending an objection. The sender proceeds if no objection is received.

The aforementioned works require a separate channel or antennas to ensure nodes have a consistent state. As we will show later, pencil beams enable nodes to set up non-overlapping communication links that obviate the need to inform all neighboring nodes of an impending transmission or reception.

III. System Model

Before describing how SpotMAC works, we first present our antenna model and the benefits of using narrow beams.

We assume nodes are equipped with an adaptive antenna array of m elements, placed half wavelength apart and arranged in a circular geometry to facilitate $360°$ beam steering. We assume the antenna array can generate m ideal beams, each having a beamwidth of $2\pi/m$ with omni-antenna gain.

Figure 2 shows an example system. In addition to the narrow main lobe, we also see four side lobes. In the inset, we see a block diagram of an antenna array. The responses of each element is represented by a steering vector or array response vector. This vector is used to compute the desired weights, w_i, that maximizes the SNR of a signal to/from a neighbor. In practice, a training sequence is used to help compute the weights [10]. Note that, adaptive antenna systems work well in multi-path environments. It is important to note that a node remains in omni-directional mode when idle, and only beamforms to transmit and receive packets.

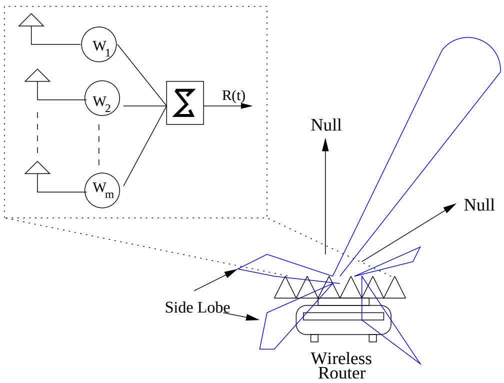

Fig. 2. Smart antenna system.

The ability to form nulls is an important property of smart antennas. In fact, a smart antenna has the ability to suppress $m-1$ interfering nodes. This minimizes the effect of neighboring nodes' side lobes and also the detrimental effect caused by a neighbor inadvertently transmitting in the direction of a receiving node due to the hidden terminal problem [10].

We like to point out that researchers have used pencil beams to increase the capacity of cellular networks by forming mini-cells that cover a lone or group of mobile terminals. Example systems include [17], [1] and [6]. To the best of our knowledge, SpotMAC is the first protocol that exploits pencil beams in wireless mesh networks.

A. Narrow Beam Benefits

Pencil beams have the following properties, (i) high spatial reuse, (ii) small number of hidden and exposed terminals, and (ii) they simplify the collision avoidance (CA) process.

In Figure 3 (a), we see that the directional transmission range of Node-A and Node-D overlaps. This means only one of them can transmit at a given point in time. Notice that, when either one of them transmits, the other will be exposed, i.e., prevented from transmitting. However, when pencil beams are used, see Figure 3(b), nodes A and C can initiate a connection to their respective neighbor simultaneously, thereby increasing spatial reuse and avoiding the exposed terminal

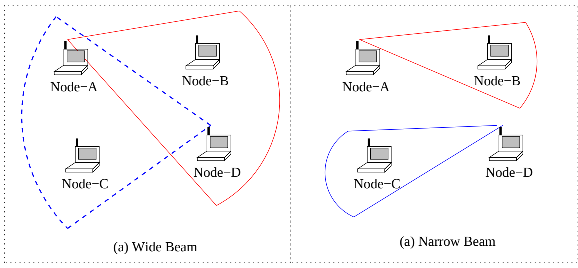

Fig. 3. Wide versus narrow beam.

problem. Moreover, the respective receivers will benefit from the improved gains due to the narrow beamwidth.

Pencil beams simplify the CA process. Current systems rely on CA to inform neighboring nodes, via RTS/CTS messages, that the channel is reserved for a given period of time, and that they should defer their transmission to prevent collisions. This changes when pencil beams are used. Control messages are only used to determine whether the receiver is free to receive. This is because the narrow beam and nulls effectively create a virtual "wire" that connects the sender and receiver. The resultant communication link not only does not cause interference to neighbors but protects the sender and receiver from interferences generated by neighboring nodes. As a result, control messages are no no longer needed to reserve the communication "floor". Having said that, we will show in Section IV-B that this is not entirely true. Nevertheless, pencil beams simplify the CA process and yield a simple solution to the hidden terminal problem.

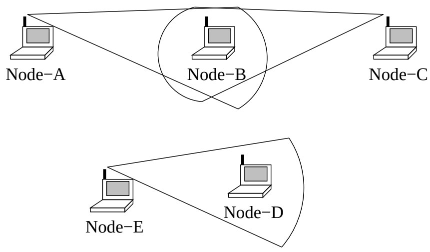

Fig. 4. Collision avoidance using narrow beams.

IV. SpotMAC

SpotMAC is based on the IEEE 802.11 MAC but uses an additional inverted RTS/CTS exchange to overcome the hidden terminal problem. It uses narrow beamwidth, rely only on in-band signaling, exploits null steering and does not require a separate channel or multiple directional antennas.

In the following sections, we first present the state maintained by each node before explaining how SpotMAC addresses the hidden terminal problem. In our discussions we assume nodes are stationary.

A. Node State

Each node is assigned a unique training sequence. This sequence is used by a receiving node to find the best antenna element weights to use using algorithms like Recursive Least

Squares (RLS) [10]. Nodes make their sequence known to all neighboring nodes via RTS and CTS messages, which increases the size RTS and CTS message by 16 octets.

All nodes record the angle of arrival (AoA) of each packet. This AoA information is then used to form nulls to interfering neighbors before communication takes place. Since we assume nodes are stationary, each neighbor's AoA does not change significantly over time. Lastly, each node maintains a separate contention window (CW) for each neighbor. Table I summarizes the state maintained by each node.

State	Description
Neighbor IDs	MAC addresses of neighbors.
Training Sequences	Each neighbor's unique 128-bit Walsh Hadamard code.
AoA_i	The AoA of neighbor i.
CW_i	Contention window for neighbor i.
NAV^{Θ}	The NAV for direction Θ.
C_i	Transmission credits for neighbor i.

TABLE I
STATE MAINTAINED BY EACH NODE RUNNING SPOTMAC.

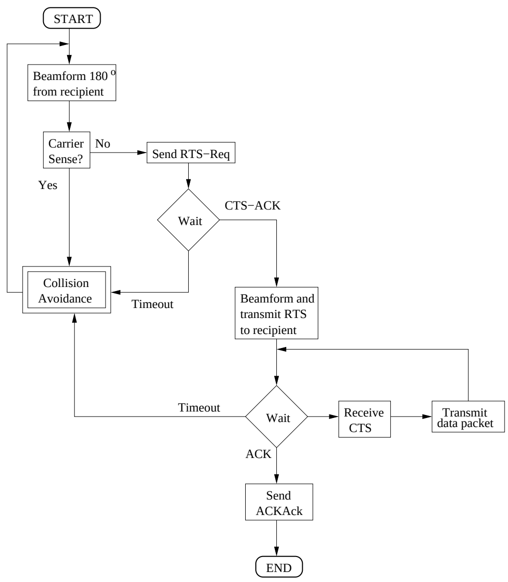

Fig. 5. Flowchart for the inverted RTS/CTS exchange.

SpotMAC also introduces three new control messages, RTS-Req, CTS-ACK and ACKAck; see Section IV-B. These messages have similar fields to the IEEE 802.11 RTS, CTS and ACK messages, except for a new type and subtype value.

B. Inverse RTS/CTS Exchange

Nulls cannot be steered toward a transmitting or receiving direction. As a result, when nodes are placed linearly, as in nodes A, B and C of Figure 4, the hidden terminal problem occurs when node A or B communicates with C. This is because when Node-B is transmitting to C, node-A is unaware that node-C is busy receiving; since it detects no carrier. When node-A transmits to node-C, a collision occurs. Bear in mind, node-A may be deaf when node-B sends its RTS message.

To address this problem, we invert the RTS/CTS exchange. A node that wants to transmit to a "downstream" neighbor must first ask its "upstream" neighbor for permission. Hence, blocking the "upstream" neighbor from transmitting and interfering with the "downstream" node's reception. In Figure 4, node-A and node-C is the upstream and downstream node of node-B respectively.

Figure 5 illustrates the inverse RTS/CTS exchange. To explain this exchange, we use Figure 4 and assume nodes A and B have packets for node-C. First, node-B beamforms to node-A and sends a RTS-Req message to block node-A from transmitting to node-C. Assuming node-A is idle and not deaf, it returns a CTS-ACK message to node-B, thereby confirming it will not transmit to node-C. Node-B then initiates a communication to node-C as usual, i.e., using the RTS/CTS/DATA/ACK exchange. Node-B then transmits an ACKAck message to node-A once its data packet is acknowledged by node-C, hence releasing node-A from transmitting in node-C's direction. In the scenario above, node-B will not receive a CTS-ACK if node-A is deaf. In this case, node-B backs off and retransmits the RTS-Req message once its backoff counter reaches zero.

The justification for sending the ACKAck message is as follows. Node-B does not know the exact time required to send the data packet to node-C. This is because a rough estimate of the time required is known only when node-C informs node-B of the best data rate to use via its CTS message; see RBAR [7]. In addition, the transmission time is affected by the number of retransmissions before the RTS and data packets are received correctly. These factors mean the actual time required by node-B to transmit a packet to node-C is highly variable. An inaccurate time leads to sub-optimal performance where node-A is blocked for an unnecessary long period, or it may result in node-A transmitting to node-C before node-B has finished its transmission.

Node-A will be blocked indefinitely if it did not receive an ACKAck message. To prevent this from happening, node-A sets its timeout timer to twice the transmission time of the maximum frame size using the base data rate after receiving a RTS-Req message. Once this timer timer expires, node-A is free from transmitting to node-C even though it has not received an ACKAck message.

Inverting the RTS/CTS exchange addresses the hidden terminal problem effectively. Consider Figure 4. Node-B will not communicate with node-C unless it has gained node-A's approval. If node-A is deaf, node-B will defer its transmission, hence when node-A returns from its directional transmission it can transmit to node-C safely. In a different scenario, if node-A is transmitting to node-C, node-B will detect node-A's transmission and backs off. In both cases, collision is averted. We note that the returning acknowledgment message is prone to the hidden terminal problem too. However, given its small size and the specific scenario which the hidden terminal problem occurs, the probability of collision is negligible.

1) An Optimization: Having to request permission from an upstream node before each transmission is overly conservative, especially if the upstream node has no packets for any nodes in the downstream node's direction. Further, the upstream node may be located sufficiently far that it's transmission does not interfere with the downstream node's reception, especially if the payload is coded appropriately.

To remedy the above limitation, SpotMAC uses the inverse RTS/CTS exchange only when it detects persistent interference from an upstream node's transmissions. In Figure 4, if node-B experiences a data transmission failure when communicating with node-C, node-B listens to the channel for CW_{min} slots. Once a packet is received, node-B records its direction. If the transmission is towards node-C's direction, node-B records the number of times, I_A, that node, i.e., node-A, has interfered with its transmission to node-C. Once node-B deems there is persistent interference from node-A, i.e., when I_A exceeds a threshold ω, node-B invokes the inverse RTS/CTS exchange whenever it wants to communicate with node-C. However, this only lasts for a given period of time. Once node-A determines that it has no packets for node-C, it informs node-B by setting the RELEASE flag in the next CTS-ACK message, thereby causing node-B to stop using the inverse RTS/CTS exchange when transmitting to node-C. We like to point out that the RELEASE flag is also set when node-A finds that node-C is not its neighbor after receiving a RTS-Req from node-B.

A receiver could also play an active role in helping the sender identify an interfering neighbor. For example, after detecting a collision, the receiver starts recording the list of neighbors that are trying to transmit to it from the direction of the sender for a short period of time. At the end of this period, the receiver returns the address of the neighbor that has the strongest signal strength to the sender.

C. Collision Avoidance

The CA process must ensure multiple nodes contending for a receiver is resolved quickly. In addition, it needs to backoff when an upstream node fails to respond with a CTS-ACK. In both scenarios, due to the use of pencil beams, we can afford to be aggressive in order to reduce channel access delay.

Figure 6 depicts the CA process. There are two notable features. Firstly, the CW of a neighbor is adjusted only after a given threshold is met. Initially, $CW = CW_{min}$, with $CW_{min} = 8$. Whenever a failure is encountered, the CW is increased exponentially once FailCount exceeds FailMAX. This ensures the sender contends for the channel quickly in case deafness was the reason the receiver did not respond to a RTS message before. The CW is decreased exponentially, bounded to $MAX(CW_{min}, CW)$, after a sender experienced $SCount$ successful transmissions. This ensures that the CW reflects the contention at the receiver.

Secondly, a sender must sense whether there is a carrier from both its upstream neighbor and the intended receiver, and only decrements the backoff counter when the channel is idle in both directions. If a carrier is detected, the backoff counter is suspended. Note, if no upstream neighbor interferes with the receiver, the sender only needs to ensure the receiver is not transmitting in its direction.

The CA process, as depicted in Figure 6, does not consider what happens when an upstream node fails to return a CTS-ACK. In our implementation, a node always use a CW of

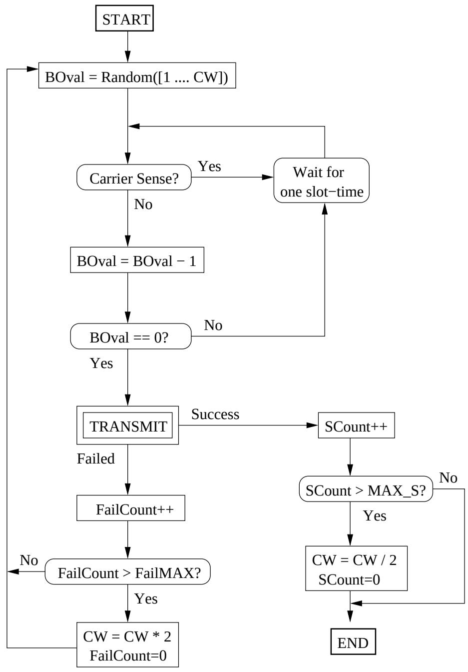

Fig. 6. Collision avoidance.

CW_{min} when sending a RTS-Req to an upstream node. This minimizes any delays due to deafness. However, the small CW may lead to unfairness if the upstream node needs to access the channel in the downstream node's direction. To improve fairness, each node maintains a credit counter, C^{Θ}, for each direction. This counter is incremented whenever it receives a RTS-Req message. When the upstream node has a packet for a neighbor, and at the same time receives a RTS-Req message from another neighbor, it checks whether its credit counter is positive. If it is, the upstream node does not respond to the RTS-Req. Hence, the RTS-Req sender is prevented from transmitting to its downstream node. To prevent the upstream node from accumulating a large number of credits and dominating the channel with them, we bound C^{Θ} to a maximum value of η.

V. SIMULATION

We augmented *ns-2* (v2.29) [14] with the IEEE 802.11a [8] MAC. Each node has eight data rates, 6 to 54 Mbps, with the best choice chosen by the RBAR [7] link adaptation algorithm. Our simulation considers the interference caused by simultaneous transmissions, which affects the signal-to-interference of each packet and hence its packet error rate. We use the shadowing radio propagation model included in *ns-2*; using a standard deviation value (*std_db*) of 4.0 and path-loss exponent of 2.0. All flows generate a maximum of 300 packets at a constant bit rate (CBR) of 1000 packets/second, each of size 512 bytes.

Nodes are equipped with adaptive antennas. Nodes record the AoA and the sender's address of each packet in a table called *AngleTable*. A node only beamforms when an incoming packet is detected. This means a node is in directional mode only when it is transmitting or receiving. At other times, it listens omni-directionally. Before each transmission, a node looks up the AoA of a receiver from the *AngleTable*. If none is found, the packet is transmitted omni-directionally. Otherwise, the node beamsforms in the AoA direction and transmits the packet. Given the AoA, both the sender and receiver can communicate directionally at maximum gain. Note, the gain is zero outside of the main and side lobes.

Figure 7 shows the topologies used to investigate the performance of SpotMAC and also to study the performance degradation due to deafness, and the hidden terminal problem. Further details are presented in Section VI.

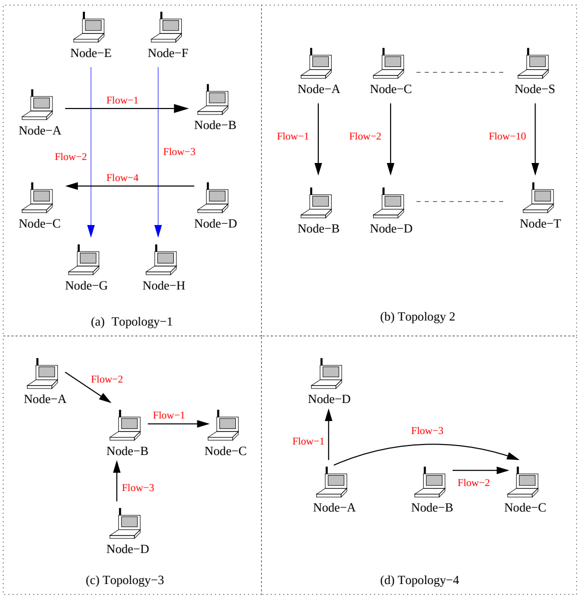

Fig. 7. Topologies used in simulations.

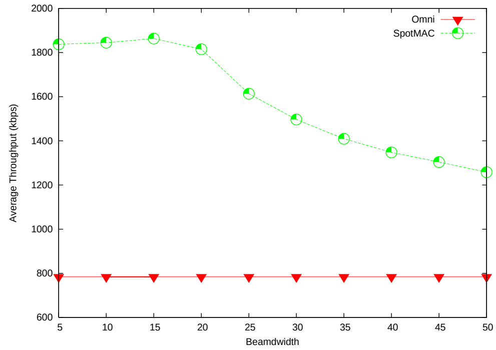

Fig. 8. Beamwidth vs. average throughput. This plot confirms that small beamwidth increases throughput due to increased spatial reuse and reduced collisions.

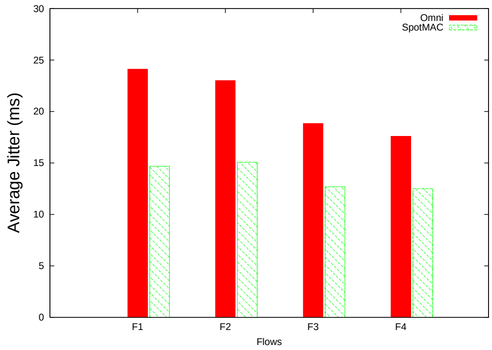

Fig. 9. Average jitter. We see up to 10ms difference in jitter experienced by flows.

VI. Results

The following sections present results from our studies on beamwidth, fairness, node density, deafness and the hidden terminal problem. Note, we assume nodes using the IEEE 802.11 MAC are equipped with an omni-directional antenna. Further, we label IEEE 802.11 MAC results as "Omni".

A. *Topology-1*

Using Figure 7(a), we study the impact of beamwidth on throughput and jitter. We also study the throughput fairness of each flow.

1) Beamwidth: In the first experiment we plot the aggregate throughput of all flows with increasing beamwidth. From Figure 8, we see that each flow maintains its throughput for beamwidth up to 20 degrees. After that, throughput drops quickly due to increased contention. Figure 9 plots the jitter experienced by each flow, using a beamwidth of five degree. Given the absence of contention, each flow's jitter is affected by channel errors only, thereby resulting in varying number of retransmission attempts.

2) Fairness: In this experiment, we calculate the throughput fairness of each flow using Jain's fairness index [9] over 50 milliseconds intervals. Note, in order for all flows to have on average a similar data rate, we ensure the sender and receiver of all node pairs have equal distance. Comparing Figures 10 and 11, we see that SpotMAC achieves significantly higher throughput fairness than the IEEE 802.11 MAC.

A key observation from our experiments is that SpotMAC minimizes the performance anomaly problem [2]. Using narrow beams, each flow is effectively shielded from all other flows. Thus, a low data rate flow that uses an unfair amount of "air-time" has little effect on a high data rate flow. Moreover, nodes are unlikely to have a large backoff value, thereby avoiding the short term unfairness of the binary exponential backoff (BEB) algorithm.

B. *Topology-2*

In this experiment we study the impact of node density on throughput using Figure 7(b). After 20 simulation runs, we add another node pair and repeat the experiment another 20 times. Figure 12 shows the aggregate throughput for varying node pairs. Due to the superior spatial reuse from using pencil beams, SpotMAC's throughput remains unchanged as we add node pairs.

C. *Topology-3*

To study the effect of deafness, we use the topology shown in Figure 7(c), where node-B is deaf to two other neighbors

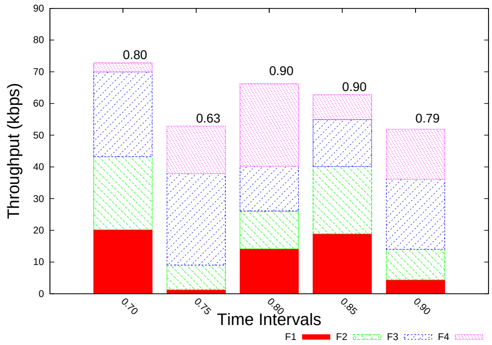

Fig. 10. Fairness. Nodes using IEEE 802.11 MAC.

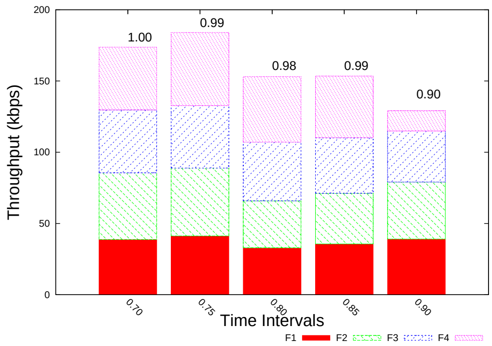

Fig. 11. Fairness. Nodes using SpotMAC.

whenever it is transmitting or receiving.

From Table II, the average throughput of each flow using SpotMAC is higher or on par with the standard IEEE 802.11 MAC. These results can be explained as follows. Whenever node-B communicates with one of its neighbors, the resulting deafness causes two other neighbors to backoff. Similarly, for the omni-directional antenna case, each node will backoff when node-B is communicating. This is because each neighbor would have detected a carrier or received node-B's RTS/CTS message. In effect, SpotMAC has a similar behavior to the IEEE 802.11 MAC. The slight throughput increase experienced by flows one and three is due to the lower packet error rate resulting from the superior gain provided by adaptive antennas.

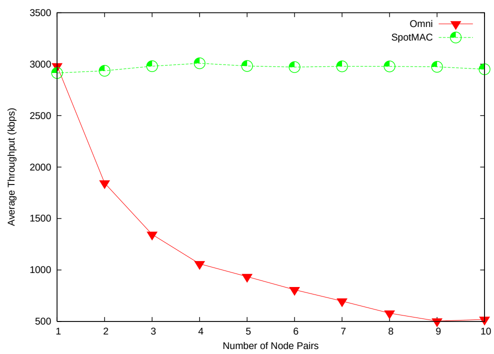

Fig. 12. Number of node pairs versus throughput. This shows that increasing node density has little effect on the performance of SpotMAC.

	Throughput (Kbps)		
	Flow-1	**Flow-2**	**Flow-3**
Omni	1310.89	1394.35	1337.37
SpotMAC	1420.49	1368.50	1551.98

TABLE II
THE IMPACT OF DEAFNESS ON SPOTMAC AND IEEE 802.11 MAC.

D. *Topology-4*

This last experiment uses the topology shown in Figure 7(d). In this topology, node-A becomes deaf to the communication between nodes B and C when it communicates with node-D, and is likely to cause a collision when it transmits to node-C.

To quantify the throughput degradation due to the hidden terminal problem, and gains due to the inverse RTS/CTS exchange, we conduct the following experiments. First, all nodes transmit omni-directionally. In this scenario, the throughput values of each flow is shown in the *Omni* column of Table III. Second, we disable the inverted RTS/CTS exchange. We repeat the previous experiment, but using adaptive antennas. Notice that, nodes A and B have the same receiver, thus causing frequent collisions at node C, and reducing all flows' throughput significantly. In fact, by approximately 335 Kbps when compared to the *Omni* case; cf. the first and second column of Table III. In the third experiment we rotate nodes A and D 90 degree anti-clockwise, thus placing them below node-C and removing the hidden terminal problem. In this modified topology, the throughput for each flow is shown in the *Deaf=No* column. Without deafness, each flow experiences a significant throughput gain; approximately 725 Kbps over the second experiment.

Flow	**Omni**	**Deaf=Yes**	**Deaf=No**	**SpotMAC**
1	1456.92	1185.24	1829.38	1701.66
2	1327.97	1054.37	1999.14	1679.79
3	1694.40	1233.07	1823.91	2917.40

TABLE III
THIS TABLE QUANTIFIES AND COMPARES THE THROUGHPUT (IN KBPS) OF EACH FLOW WHEN DEAFNESS IS PRESENT, AND THE EFFECTIVENESS OF THE INVERTED RTS/CTS EXCHANGE.

We now enable the inverse RTS/CTS exchange and reset nodes A and D to their original position. The throughput achieved by each flow using *SpotMAC* is shown in the last column of Table III. We see SpotMAC achieves a higher throughput than nodes using the IEEE 802.11 MAC. More important, it resolves the performance degradation due to deafness. Comparing the second and fifth column, on average, flows obtained a throughput increase of 942 Kbps.

SpotMAC is unable to match or better the throughput obtained when there is no deafness, with the exception of Flow-3 which experienced an improvement of approximately 1024 Kbps. Other flows experience a reduction of around 223 Kbps. This reduction is primarily attributed to the delay incurred by the inverted RTS/CTS exchange. Moreover, this delay is exacerbated when node-A is deaf, thus it is oblivious to node-B's RTS-Req message. This causes node-B to backoff and leave the channel idle. After transmitting to node-D, node-A may initiate a transmission to node-C, causing node-B to defer its transmission further. Despite these limitations, SpotMAC performs well and approximates the throughput of the third experiment, but at the expense of fairness. For example, Flow-3 has a disproportionate high throughput compared to other flows. This is due to node-A taking full advantage of the idle channel when node-B backs off. To ensure fairness, node-A must be aware of its share of the channel, and also that of its respective neighbors. We leave this as future work.

VII. Conclusion

We have shown how SpotMAC exploits pencil beams to achieve several orders of magnitude higher throughput than the IEEE 802.11 MAC. This is achieved by exploiting the following key benefits, high spatial reuse, simplification of the collision avoidance process, and restricting the hidden terminal problem to a linear topology which can be solved using an inverse RTS/CTS exchange. Further, pencil beams minimize the negative effect of the performance anomaly experienced by nodes using a disproportionate amount of "air-time" due to varying data rates.

VIII. Acknowledgment

I like to thank Ricardo Gandia-Sanchez for many fruitful discussions, and for implementing an interference aware physical layer and incorporating directional antennas into the *ns-2* simulator.

References

[1] A. S. Acampora, T.-S. Chu, C. Dragone, and M. J. Gans. A metropolitan area radio system using scanning pencil beams. *IEEE Transactions on Communications*, 39(1), Jan. 1991.
[2] G. Berger-Sabbatel, F. Rousseau, M. Heusse, and A. Duda. Performance anomaly of 802.11b. In *Proceedings of IEEE INFOCOM'2003*, San Francisco, USA, 2003.
[3] R. R. Choudhury and N. H. Vaidya. Deafness: A MAC problem in ad hoc networks when using directional antennas. In *Proceedings of the 12th IEEE ICNP*, Berlin, Germany, Oct. 2004.
[4] R. R. Choudhury and N. H. Vaidya. On designing mac for wireless networks using beamforming antennas. *IEEE Transactions on Mobile Computing*, 5(5), May 2006.
[5] T. Elbatt, T. Anderson, and B. Ryu. Performance evaluation of multiple access protocols for ad-hoc networks using directional antennas. In *IEEE Wireless Communications and Networking Conference (WCNC'03)*, New Orleans, USA, Mar. 2003.
[6] T. Gebauer and H. G. Gockler. Channel-individual adaptive beamforming for mobile satellite communications. *IEEE Journal on Selected Areas in Communications (JSAC)*, 13(2), 1995.
[7] G. Holland, N. Vaidya, and P. Bahl. A rate-adaptive MAC protocol for multi-hop wireless networks. In *ACM MOBICOM'2001*, Rome, Italy, Oct. 2001.
[8] IEEE. Part 11a: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications: High speed physical layer in the 5 GHz band, standard specification. IEEE Std 802.11a-1999, 1999.
[9] R. Jain, D.-M. Chiu, and H. Hawe. A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer Systems. DEC RR TR-301.
[10] J. C. L. Jr and T. S. Rappaport. *Smart Antennas for Wireless Communications: IS-95 and Third Generation CDMA Applications*. Prentice-Hall, 1999.
[11] Y. Ko, V. Shankarkumar, and N. H. Vaidya. Medium access control protcols using directional antennas in ad hoc networks. In *IEEE Infocom*, Tel Aviv, Israel, 2000.
[12] T. Korakis, G. Jakllari, and L. Tassiulas. A MAC protocol for full exploitation of directional antennas in ad-hoc wireless networks. In *ACM MobiHOC*, Annapolis, USA, Aug. 2003.
[13] D. Lal, R. Toshniwal, R. Radhakrishnan, D. P. Agrawal, and J. C. Jr. A novel MAC layer protocol for space division multiple access in wireless ad-hoc networks. In *IEEE 11th ICCCN*, Miami, FL, USA, Oct. 2000.
[14] S. McCanne and S. Floyd. ns network simulator-2. http://www.isi.edu/nsname/ns/.
[15] A. Nasipuri, S. ye, J. You, and R. E. Hiromoto. A MAC protocol for mobile ad hoc networks using directional antennas. In *IEEE WCNC*, Chicago, IL, Sept. 2000.
[16] R. Ramanathan. Ad-hoc networking with directional antennas: A complete system solution. In *ACM MobiHOC*, USA, Aug. 2001.
[17] S. C. Swales, M. A. Beach, D. J. Edwards, and J. P. McGeehan. The performance enhancement of multibeam adaptive base-station antennas for cellular land mobile radio systems. *IEEE Transactions on Vehicular Technology*, 39(1), Feb. 1990.
[18] R. Vilzmann, C. Bettstetter, and C. Hartmann. BeamMAC: A new paradigm for medium access in wireless networks. *International Journal of Electronics and Communications*, 60, 2006.
[19] Y. Wang and J. G.-L. Aceves. A new hybrid channel access scheme for ad hoc networks. In *Proceedings of the 1st Annual Mediteranean Ad-Hoc Networking Workshop (Med-Hoc-Net)*, Sardegna, Italy, Sept. 2002.
[20] Y. Wang and J. Garcia-Luna-Aceves. Spatial reuse and collision avoidance in ad hoc networks with directional antennas. In *IEEE Globecom*, Taipei, Taiwan, 2002.
[21] C. Zhu, T. Nadeem, and J. R. Agre. Enhancing 802.11 wireless networks with directional antenna and multiple receivers. In *Proceedings IEEE Milcom'06*, Washington, DC, USA, Oct. 2006.