Tone based MAC protocol for use with Adaptive Array Antennas

Harkirat Singh and Suresh Singh
harkirat,singh@cs.pdx.edu
Department of Computer Science
Portland State University
Portland, OR 97207

***Abstract*— This paper presents a novel Tone-based 802.11b protocol for use in wireless networks where nodes are equipped with adaptive array antennas. The protocol relies on the ability of the antenna and Angle of Arrival (AoA) algorithms to identify the direction of transmitters and then beamform appropriately to maximize SINR (Signal to Interference and Noise Ratio) at the receiver. The performance of the protocol is evaluated using detailed simulation in OPNET and Matlab where we demonstrate the benefits of using adaptive antenna arrays. The impact of using different number of antenna elements is also studied for this environment.**

I. Introduction

In the past several years a great deal of research has been done to better understand the networking issues in WLAN. The predominant model used in most of this work has assumed omni-directional antennas in which the transmission energy is spread out in all the directions from the sender. In this work, on the other hand, we used adaptive array antennas. The benefit of using this type of directional antenna is that the sender can focus the transmission energy in a narrower region while reducing the effect of interferers. A receiver can run an Angle of Arrival algorithm to determine the direction of the maximum strength signal and place nulls in some other directions to cancel the interfering signals. This results in increased network capacity due to less interference, and an opportunity for reduced energy consumption overall.

To motivate the use of adaptive array antennas, it is useful to enumerate the additional capabilities provided by these antennas over and above those provided by directional antennas alone.

1) *Silencing Interferers:* If a receiver knows that there are interfering transmitters in its neighborhood, it can form a directed beam towards the sender while simultaneously placing nulls in the direction of the other transmitters. A null effectively cancels the received signal power from a transmitter (even if the interferer is more powerful than the desired transmitter) and ensures a high SINR at the receiver.
2) *Enhanced Neighbor Discovery:* Identifying the direction of neighbors is necessary for transmission (so as to beamform appropriately) as well as for reception (to silence interferers). Earlier directional antenna papers typically use some form of sequential polling to identify the direction of one-hop neighbors [1]. Thus, for a 45^0 sectored antenna, there are eight directions in which a node will periodically poll neighbors. Adaptive antennas, on the other hand, can considerably ease the complexity of this task by running a DSP module called Angle of Arrival (AoA).
3) *Flexible Beamforming:* An adaptive array system can be configured as an omni-directional antenna or as a directional antenna with variable beamwidths (limited by the number of antenna elements) and with arbitrarily precise boresight[1]. This flexibility allows us to explore the protocol space with arbitrary combinations of beamwidths for collision avoidance and data transmission.

The remainder of this paper is organized as follows. In the next section, we present an overview of adaptive array antennas. Section III summarizes the previous work in this and related areas. In section IV we describe our tone-based 802.11b protocol in detail. Section V presents our simulation results.

II. Overview of Adative Array Antennas

Adaptive Array antennas are directional antennas which can change the *direction, shape, and power* of the antenna beam thus allowing a great deal of adaptability in setting up connections. In general, the array may consist of a number of antenna elements distributed in any desired pattern; however, the array is frequently implemented as a *linear equally spaced* (LES), uniform circular, or plannar array.

The output of the array of the antenna elements is taken as input to a module called the Angle of Arrival (AoA). This module computes the *angular* direction in which it detects electromagnetic energy. The output of the AoA gives a list of signal strengths and direction (θ, ϕ) (or, elevation and azimuth) from which this signal strength is incident on the antenna array. It is important to note that the AoA only gives us the direction and not the actual location of a single source. In addition, the AoA may not be able to discriminate between two signal sources that have a small *angular separation* at the receiver.

This work is funded by the NSF under grant ANIR-0125728.

[1]Sectored antennas, for example, are relatively inflexible in this regard which can cause more collisions for nodes that lie outside the 3dB beamwidth of the main beams.

Each branch of the array has a weighing element, w_i. The *weighing element* w_i, has both a magnitude and a phase associated with it.

$$w_i = e^{j\beta i d sin\theta_i}$$

By adjusting the set of weights, w_i, it is possible to direct the main lobe (maximum of the main beam) of the array in any desired direction. The number of antenna elements changes the beamwidth of the main lobe. As the number of elements increases, the beamwidth becomes narrower. For a more detailed discussion, please see [2]. We implemented Adaptive Array antennas in Matlab and interfaced it with the radio pipeline stage of the OPNET.

III. Literature Review

Recently there have been several papers that have looked at the problem of MAC design for ad hoc networks where nodes are equipped with directional antennas. The directional antenna models used include switched beam antennas (the antenna is sectored and one of these sectors is used depending the direction of the communicating node), multi-beam antennas (here more than one beam can be used simultaneously), and adaptive antenna arrays (here the beam can be made to point in any direction as described in section II).

Among the papers that have studied the application of adaptive antenna arrays in ad hoc networking are [3], [4], [5]. [5] provides a high-level discussion of some promising research areas when using adaptive antenna arrays in ad hoc networks. They however do not provide any results. [3] examines the interaction and integration of several critical components of adaptive antenna arrays for use in ad hoc networks. This paper focuses on the design of these antennas for mobile devices with operating frequency of 20Ghz. The authors examine the impact of the antenna design on network throughput and the impact of mutual coupling[2] on the performance of adaptive algorithms. The paper reports results of detailed OPNET simulations using a TDMA version of the 802.11 protocol in a *single-hop network*. The network size studied was 55 nodes and the performance of 8x8 and 4x4 planar array antennas was studied. A maximum throughput of 9 pkts/packet-time was achieved using the 8x8 array in a single-hop network.

[4] studies the performance of Spatial TDMA and CSMA/CA in multihop ad hoc networks using adaptive antenna arrays. They examine the performance of different RTS/CTS schemes (DRTS/DCTS, ORTS/OCTS, DRTS/OCTS, etc.) on throughput. The main results indicate that narrower beamwidths (10^o) do give the highest throughput though this value is not too different from the case when using 60^o beamwidths. In addition, they also performed simulations with dense as well as sparse networks. The highest throughput was achieved in dense networks (average degree 10.2) with lower throughput in sparse networks (average degree 6.9). This paper did not exploit the benefits of *nulling and AoA* as we do in our paper.

[2]Mutual coupling results in radiation patterns that have shallower and shifted nulls, and less accurate AOA (Angle Of Arrival), thus deteriorating overall network throughput.

[6] develops slotted scheduling-based MAC protocols for nodes equipped with directional antennas. The directional antenna considered is a multi-beam adaptive array antenna (MBAA) which is capable of forming multiple beams. The protocols assume that nodes can engage in several simultaneous transmissions. The key contribution of the paper is the development of a neighbor tracking scheme that is then used to schedule transmissions by each node in a distributed way. Several recent papers have looked at MAC design using sectored directional antennas.

[7] is one of the early papers which touches upon various ad hoc networking issues when using directional antennas. The authors discuss issues such as power control, hidden terminal problem, and which antenna models to use. [8] proposes a MAC protocol that uses directional antennas where mobile nodes do not have any location information. Each node is equipped with M directional antenna elements each of which has a conical pattern, spanning an angle of $2\pi/M$ radians. The MAC protocol is assumed to be capable of switching any one or all the antennas to active or passive modes. The protocol uses omnidirectinal RTS/CTS exchange followed by directional data transfer. [9] also develops MAC protocols for this scenario. They present a MAC protocol based on Directional RTS/CTS or a combination of DRTS/O-CTS (Omnidirectional CTS). The protocol assumes that if a transmission is happening in some direction then it will defer all transmissions in that direction. Similarly, [10], [11] consider a switched beam antenna model and they use a directional antenna at the receiver. A second protocol based on DRTS/DCTS assumes two separate channels: one for data and another for signaling.

In [12], [13] the authors assume that each node maintains neighbor Angle-SINR table (AST) and they provide a link state based table-driven routing and MAC protocol. Based on AST a node calculates an affinity for an angle which provides maximum SINR. Based on this a NLS Table is formed. Nodes in the beamformed region remain in the omni mode but they make nulls in the direction of ongoing transmissions.

[14] investigates the performance of Smart Antennas in MIMO channels. [1] uses directional transmissions for control and data packets. It uses a directional-NAV table for transmission scheduling and collision avoidance. However, they do not exploit the capabilities of the smart antennas, such as beam steering and the placement of nulls in the direction of interferers.

In [15] a multi-hop RTS is proposed to establish links between distant nodes. The direction in which the main lobe is to be oriented is determined by the MAC protocol (which in turn is provided this information by the network layer which is assumed to be neighbor-aware). The authors note that node alignment negates the benefits achieved due to directional antennas, however, unaligned routes enhances the spatial reuse. They show that their protocol has a 4-5x throughput as compared with 802.11. [16] describes the performance of 802.11 when using adaptive antenna arrays. Like [4], the authors consider the omni-RTS/omni-CTS followed by directional packet transmission within the context of 802.11. The transmit power for the data packet is smaller than that used for the RTS/CTS exchange and the authors present several power control variants. It is noteworthy that [15] also used 25-node grid networks but obtained a larger relative improvement (with respect to 802.11) in throughput compared with [16].

In [17] a node caches AOA information based on signals received and nodes remain in promiscuous mode to cache signals. 802.11 specifications say that RTS needs to be transmitted 7 times, so a node will transmit 4 directional RTS and remaining the 3 as omni-directional RTS if there is no response to the directional RTS. A circular antenna with 6 elements is assumed, and a node is capable of electronically steering the boresight towards a specific direction. A constant beamwidth of 45 deg assumed. However, it was observed that as the boresight changes, the side lobe pattern changes drastically. The key insight here is that the effects of side and back lobes cannot be ignored in the evaluation of network performance with directional antennas. [17] shows that using an ideal antenna results in a maximum throughput of 2.2Mbps while using a realistic antenna has a maximum throughput of only 1.4Mbps. *This fact, that antenna patterns matter in evaluating MAC behavior, is one that has largely been ignored by a great many authors who assume ideal antenna patterns.* Our work here differs from all of the above papers in the following ways:

- Our antenna model is made up of a linear array of antenna elements and we exploit AoA information as well as nulling capability of the antenna to maximize SINR at the receiver.
- Unlike the other protocols, we do away with RTS/CTS and instead rely on a simple tone-based scheme to establish a link. *Our protocol does not explicitly silence hidden terminals.*
- Following the lesson of [17], we use realistic antenna patterns in our studies.
- We do not use separate channel for tones as is done in [11].

IV. Description of Tone-based 802.11b

Consider the case when a node a needs to transmit a packet to node b which is its one-hop neighbor. We assume that a knows the angular direction of b (as in [15]) and it can therefore form a beam in the direction of b. However, to maximize SINR, b should also form a beam towards a and form nulls in the direction of all other transmitters. In order to do this, b needs to know two things – first, that a is attempting to transmit to it, and second, the angular direction of all the other transmitters that interfere at b.

Our protocol is based on the 802.11b standard with some changes as noted below. To transmit a packet, transmitters beamform towards their receivers and transmit a short *sender-tone* to initiate communication. This sender tone is a short, pure sinusoid. Idle nodes remain in an omni-directional mode and receive a complex sum of all such tones (*note that the tones are identical for all nodes and thus we cannot identify the nodes based on the tone*) and run a AoA algorithm to identify the direction and strength of the various signals. An idle node then *beamforms in the direction of the maximum received signal strength and forms nulls in other directions* and transmits a short *receiver-tone*. The transmitter waits for a *receiver-tone* and transmits its packet upon reception of a receiver-tone. After transmission of a packet, the sender waits for the receipt of an ACK. If there is no ACK, it enters backoff as in 802.11b. We note the following specifics of our protocol:

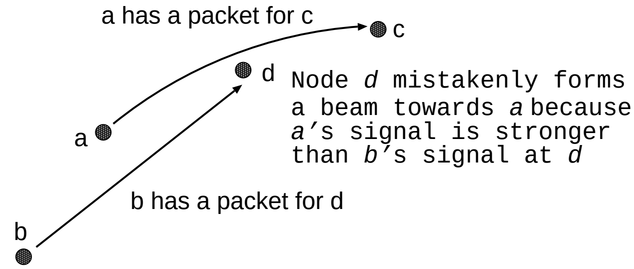


Fig. 1. False beamforming.

- The protocol is *unslotted*.
- The *sender–tone* and *receiver–tone* are pure sinusoids and thus contain no information about the sender or receiver identity (as is contained in RTS/CTS messages). The same sender–tone (and the same receiver–tone) is used by all senders (receivers). The benefit of using a tone is that its is very short (equal to the length of the preamble of a 802.11 frame).
- The intuition behind the receiver beamforming in the direction of the maximum signal is that, because of the directivity of the antenna, there is a high probability that it is the intended recipient for the packet. However, we note that in cases, as in Figure 1, the receiver d incorrectly beamforms towards a because a's signal is stronger than b's. While this is not a serious problem in most cases, we can envision scenarios where the $b \longrightarrow d$ transmission gets starved due to a large volume of $a \longrightarrow c$ traffic. An optimization we have therefore implemented is a *single-entry cache* scheme which works as follows:
  - If a node beamforms incorrectly in a given timeslot, it remembers that *direction* in a single-entry cache.
  - In the next slot, if the maximum signal strength is again in the direction recorded in the single-entry cache, then the node ignores that direction and beamforms towards the second strongest signal.
    - If the node receives a packet correctly (i.e., it was the intended recipient), it does not change the cache.
    - If it receives a packet incorrectly, it updates the cache with this new direction.
  - If there is no packet in a slot from the direction recorded in the cache, the cache is reset.

  This simple mechanism ensures that in cases similar to Figure 1, connections are not starved. However, we can construct more complex scenarios where a single-entry cache will fail to prevent starvation. In these cases, more sophisticated multiple-entry caching schemes are required. However, in our simulations, we only use the single-entry caching scheme because the probability of more complex scenarios resulting in starvation are very rare.

- Unlike previous CSMA/CA protocols, we make no attempt to combat the hidden terminal problems and do not maintain the NAV. However, this can cause problems as illustrated in Figure 2. Here, a transmits a tone to b but since c also lies in the beam, c responds with a receiver–tone. a transmits the packet which is not received by b since b is busy receiving a packet from d. As a result of this, there is the potential for interference at b. We handle this problem by the following simple mechanism:
  - In our example, node b *dynamically forms a null* towards a when it detects interference with its ongoing packet reception.
  - In addition, it recovers from the brief interference due to a sender-tone by relying on additional FEC (Forward Error Correction) bits (the idea of using FEC to recover from multiple short interferences has been explored by others, for example [18]).

Figure 3 illustrates some of the hidden terminal problems that may occur in our scheme. Here, node e is a one-hop neighbor of node a and node c is a one-hop neighbor of node b. Assume that node a is sending data to node b. Consider the following cases:

  - c wants to transmit to b. In this case, c will send a tone to b. Given that we have a linear antenna array, the beam pattern at b will form 2 lobes – one towards a and one, possibly, towards c. Thus, the sender–tone sent by c will interfere with b's reception (a different antenna such as a planar antenna may not have this problem). This problem can be handled to a degree by dynamically forming nulls and by using additional error-correcting codes in all data packets as mentioned above.
  - c wants to transmit to e. In this case, the receiver tone will interfere with a's transmission to b. Thus, b will be unable to receive a's transmission because it cannot form a null towards e while also receiving a packet from a.
  - d wants to transmit to b. In this case, either b already has a null towards d or it can form a null towards d after it receives the first tone.

There are other similar hidden terminal problems that can arise in our protocol. However, because of the ability to form nulls dynamically and the use of FEC, we can minimize the negative effect of the hidden terminals. Thus, **we are trading off protocol simplicity for some loss of performance due to hidden terminals.**

A. Details of Tone-based 802.11b

As we noted earlier, an exchange of *tones* (sender–tone and receiver–tone) is used for the initiation of a new connection. The tone format is a combination of the DSSS PLCP Preamble and PLCP header. According to IEEE 802.11 DSSS PLCP Preamble is 144 bits and PLCP header is 48 bits so the total tone length is equal to 192 bits. Figure 4 provides a state diagram of our tone-based protocol. The behavior of the protocol in various states can be summarized as follows:

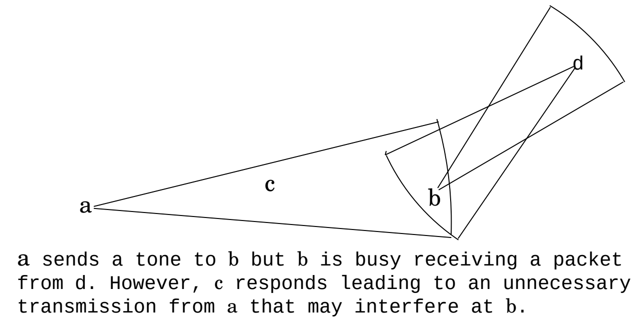


Fig. 2. Incorrect packet transmission.

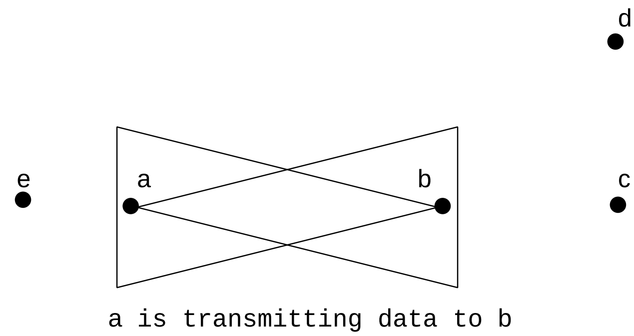


Fig. 3. Hidden terminal problems.

Idle: In case a node has no packet to send, it will remain in the Idle state and set its antenna to operate in the omni-directional mode. If it receives a sender–tone from some other node, it will move into the Data Receive Wait state. On the other hand, if it wishes to send data, it will beamform in the direction of the receiver. It chooses a random number between [0..CW] and sets the CW (Contention Window) timer[3]. When the CW timer expires, it sends a sender–tone in the direction of the receiver and moves to the ACK Wait state. If, before the CW timer expires, the node receives a sender–tone from another node, it will freeze its CW timer and move to Data Receive wait state.

Data Receive Wait: A node will move to this state in the event it receives a sender–tone. The node will beamform towards the sender and then randomly defer transmitting the receiver–tone by choosing a random waiting period between $[0..32]\times 20\mu$sec. The reason for deferring the reply is to minimize the chance of several receiver–tones colliding at the sender[4].

After transmitting a receiver–tone, the node remains in this state for 2τ (twice the maximum propagation delay+tone transmission time). If it does not hear a transmission, it returns to the Idle state. If it hears the start of a transmission, it remains in this state and receives the

[3]The random number selected is multiplied with slot-time i.e. 20μsec.

[4]Note that our tones do not carry information about the sender and the receiver so if all the nodes who receive a sender–tone (and are in idle state) respond immediately then the sender will detect a collision.

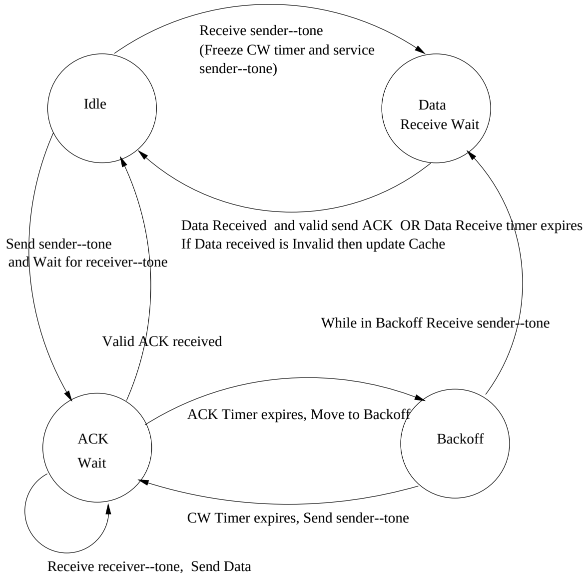


Fig. 4. State diagram of the Tone-based 802.11b protocol.

packet. It then discards the packet if the packet was meant for some other node (node c discards a's transmission in Figure 2). If, however, the packet was meant for it, then it sends an ACK.

Ack Wait: If the sender node receives a receiver–tone before the tone RTT timer goes off (which is twice the tone transmission time plus propagation delay) it will transmit the data packet. Reception of a valid ACK will move the node to the idle state, and if packets are there in the queue then it will schedule the one at the head of the queue. The node will move to Backoff state under two conditions 1) a receiver–tone did not arrive, 2) an ACK was not received following transmission of the data packet.

Backoff: The node computes a random Backoff interval (as in 802.11) and remains in backoff for this time period (it also resets its antenna to omni-directional mode). If, however, a sender-tone is received, it freezes the backoff timer and enters the Data Receive Wait state. If the node is in backoff, upon expiration of the timer, it retransmits the sender–tone, increments the retransmit counter, and enters the ACK Wait state. A packet is discarded after the retransmit counter exceeds Max_Retransmit= 7, as in the IEEE 802.11 standard.

The reception of a data packet by a node may be interfered with transmissions of sender–tones, receiver–tones, or other data packets (since our protocol does not take care of hidden terminals). A node engaged in receiving a data packet can dynamically form nulls towards new interferers, but this process takes some time (*we model this time as the length of a sender–tone*). Thus, the data packet will have errors due to this interference. We combat this error by relying on FEC (Forward Error Correcting) codes as used in IEEE 802.11e, where (224, 208) shortened Reed Solomon (RS) codes are used. In 802.11e, a MAC packet is split into blocks of 208 octets and each block is separately coded using a RS encoder.

Simulation Parameters	
Background Noise + ambient Noise	-143 dB
Propagation model	Free space
Bandwidth	1,000 kHz
Min frequency	2,402 MHz
Data Rate	2000 kbps
Carrier Sensing Threshold	+3dB
Minimum SINR	9 dB
Bit Error	Based on BPSK Modulation curve
Packet Size	512 bytes
Maximum radio range	250 m
Number of nodes	25
Area	1500x1500m

TABLE I
OPNET SIMULATION PARAMETERS.

A (48,32) RS code, which is also a shortened RS code, is used for the MAC header, and CRC-32 is used for the Frame Check Sequence (FCS). Note that any RS block can correct up to 8 octet errors.

V. PERFORMANCE STUDY

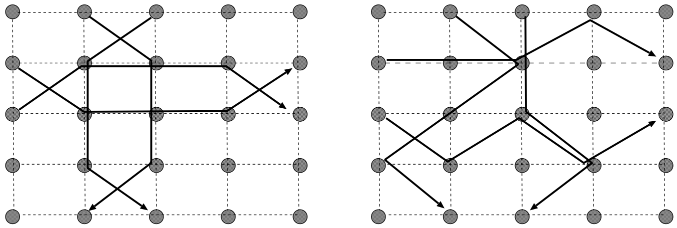
For our simulation, we used OPNET and modified the existing 802.11b implementation in OPNET to create Tone-based 802.11b. The modifications included adding the two-tones (sender and receiver) as well as changing the FEC to the 802.11e specification. We interfaced liner array (implemented in Matlab) with the *radio pipeline stage* of OPNET. The remainder of the simulation parameters are detailed in Table I.

We evaluate the performance of the protocol using 5x5 mesh with four pre-defined flows. Figure 5 shows the network topology and flows used for two of these experiments. For the third experiment, we used a random node placement on the grid where a node's position is randomly shifted in the x-axis and y-axis by adding a displacement randomly selected from [-150m , +150m] and the flows are as in Figure 5(b). The traffic is CBR (Constant Bit Rate) which increases (per flow) from 75kbps to 2Mbps. The packet size is 512 bytes. Figure 6 plots the aggregate throughput as a function of the data rate of one flow (for Figure 5(a)) for two antenna elements – one with 8 elements and one with 16 elements. Figure 7 does the same for 5(b) and 8 corresponds to the random mesh topology case. We used 10 different cases for random flows (Figure 5(b)) and randomly mesh case.

We see that using 16 antenna elements as opposed to 8 elements makes a big difference in aggregate throughput. This is not surprising because the average beamwidth when using 16 antenna elements is smaller than when using 8 elements, hence, there is a greater potential for spatial reuse with 16 antenna elements. We note that performance of our protocol is 3x–9x better than 802.11b protocol.

VI. CONCLUSION & FUTURE WORK

This paper presents a tone-based 802.11b protocol for use with adaptive array antenna systems. This protocol does not explicitly combat hidden terminals yet it shows very high



(a) Four flows (some alignment) (b) Randomly selected flows

Fig. 5. 5x5 grid topology used for the performance study.

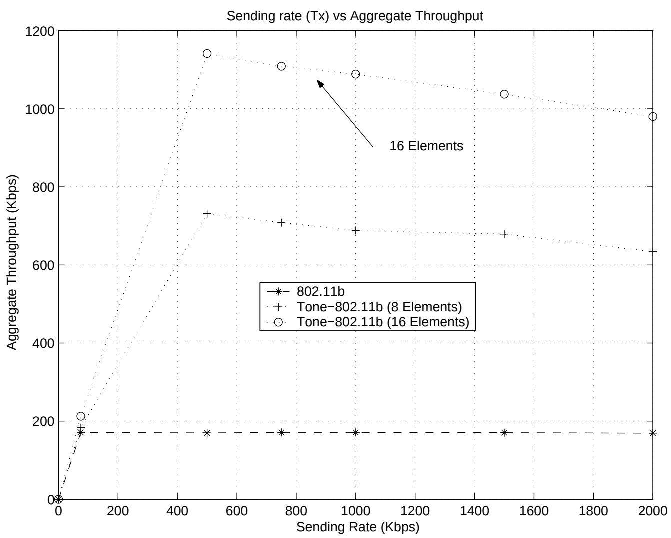


Fig. 6. Performance of our protocol in 5(a).

throughput. This happens because receivers exploit AoA information to cancel interferening transmissions dynamically. In the future we would like to extend the study to the multi-path environments.

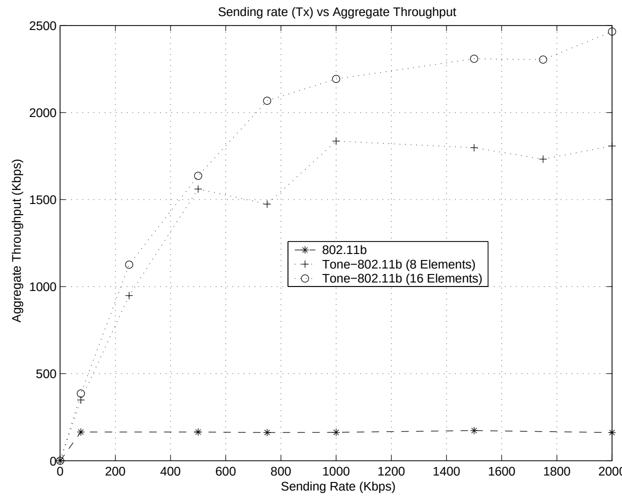


Fig. 7. Performance of our protocol in 5(b).

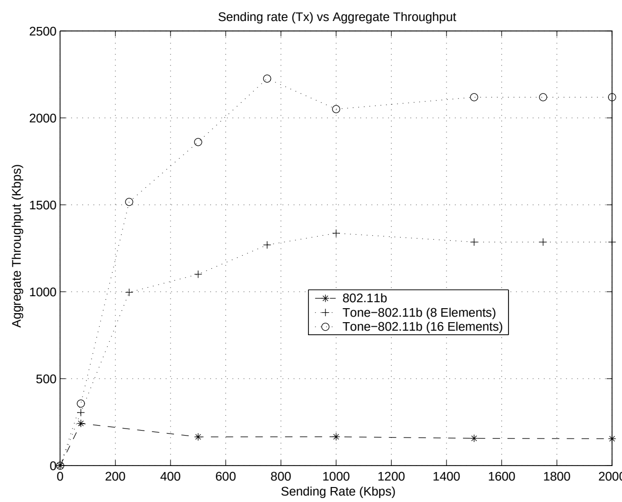


Fig. 8. Performance of our protocol when using random mesh.

REFERENCES

[1] Gentian Jakllari Thanasis Korakis and Leandros Tassiulas, "A mac protocol for full exploitation of directional antennas in ad-hoc wireless networks," in *ACM Mobihoc'03*, 1 – 3 June 2003.

[2] J. C. Liberti and T. S. Rappaport, *Smart Antennas for Wireless Communications*, Prentice Hall, 1999.

[3] S. Bellofiore, J. Foutz, R. Govindarajula, I. Bahceci, C. A. Balanis, A. S. Spanias, J.M. Capone, and T. M. Duman, "Smart antenna system analysis, integration, and performance for mobile ad-hoc networks (manets)," *IEEE Transaction on Antennas and Propagation*, vol. 50, no. 5, pp. 571 – 581, May 2002.

[4] G. Marvin Sanchez, "Multiple access protocols with smart antennas n multihop ad hoc rural-area networks," M.S. thesis, Royal Institute of Technology, Sweeden, Radio Communication Systems Laboratory, Department of Signals, Sensors and Sytems, June 2002.

[5] Hend Koubaa, "Reflections on smart antennas for mac protocols in multihop ad hoc networks," in *European Wireless'02*, 25 – 28 Feb 2002.

[6] Lichun Bao and J.J. Garcia-Luna-Aceves, "Transmission scheduling in ad hoc networks with directional antennas," in *ACM/SIGMOBILE MobiCom 2002*, 23 – 28 Sep 2002.

[7] Ram Ramanathan, "On the performance of ad hoc networks with beamforming antennas," in *ACM/SIGMOBILE MobiHoc 2001*, Oct 2001.

[8] J. You A. Nasipuri, S. Ye and R. Hiromoto, "A mac protocol for mobile ad hoc networks using directional antennas," in *IEEE WCNC*, 2000.

[9] V. Shankarkumar Y.B. Ko and N.H. Vaidya, "Medium access control protocols using directional antennas in ad hoc networks," in *IEEE INFOCOM 2000*, March 2000.

[10] Zhuochuan Huang and Chien-Chung Shen, "A comparison study of omnidirectional and directional mac protocols for ad hoc networks," in *IEEE Globecom 2002*, 2002.

[11] Chavalit Srisathapornphat Zhuochuan Huang, Chien-Chung Shen, "A mac protocol based on directional antenna and busy-tone for ad hoc networks," in *IEEE MILCOM 2002*, 2002.

[12] S. Horisawa S. Bandyopadhyay, K. Hausike and S. Tawara, "An adaptive mac and directional routing protocol for ad hoc wireless networks using espar antenna," in *ACM/SIGMOBILE MobiHoc 2001*, Oct 2001.

[13] S. Roy, D. Saha, S. Bandyopadhyay, T. Ueda, and S. Tanaka, "A network-aware mac and routing protocol for effective load balancing in ad hoc wireless networks with directional antenna," in *ACM Mobihoc'03*, 1 – 3 June 2003.

[14] Karthikeyan Sundaresan and Raghupathy Sivakumar, "On the medium access control problem in ad-hoc networks with smart antenna," in *ACM Mobihoc'03*, 1 – 3 June 2003.

[15] Romit Roy Choudhury, Xue Yang, Ram Ramanathan, and Nitin H. Vaidya, "Using directional antennas for medium access control in ad hoc networks," in *ACM/SIGMOBILE MobiCom 2002*, 23 – 28 Sep 2002.

[16] N. Fahmy, T. D. Todd, and V. Kezys, "Ad hoc networks with smart antennas using 802.11-based protocols," in *IEEE ICC'02*.

[17] Rajiv Bagrodia Mineo Takai, Jay Martin and Aifeng Ren, "Directional virtual carrier sensing for directional antennas in mobile ad hoc networks," in *ACM/SIGMOBILE MobiHoc 2002*, Oct 2002.

[18] S. Choi, "Ieee 802.11e mac-level fec performance evaluation and enhancement," in *IEEE Globecom'02*.