An Integrated Neighbor Discovery and MAC Protocol for Ad Hoc Networks Using Directional Antennas

Gentian Jakllari, Wenjie Luo, and Srikanth V. Krishnamurthy

***Abstract*— Many MAC sub-layer protocols for supporting the usage of directional antennas in ad hoc networks have been proposed in literature. However, there remain two open issues that are yet to be resolved completely. First, in order to fully exploit the spatial diversity gains possible due to the use of directional antennas, it is essential to shift to the *exclusive* usage of directional antennas for the transmission *and* reception of all the MAC layer frames. This would facilitate maximal spatial re-use and will efface the phenomena of *asymmetry in gain*. Second, in the presence of mobility the MAC protocol should incorporate mechanisms by which a node can efficiently discover and track its neighbors.**

In this paper we propose PMAC, a new MAC protocol that addresses both the issues in an integrated way. PMAC incorporates an efficient mechanism for neighbor discovery, and a scheduling based medium sharing that allows for exclusive directional transmissions and receptions. We perform analysis and simulations to understand the performance of our scheme. We find that each node, on average, can achieve a *per node* utilization of about 80 % in static and about 45 % in mobile scenarios. In terms of throughput, our protocol is seen to outperform both the traditional IEEE 802.11 and previously proposed MAC protocols for use with directional antennas in ad hoc networks.

***Index Terms*— Directional antennas.**

I. INTRODUCTION

CONTINUING reductions in the cost and size of array antennas will soon make their deployments in ad hoc networks possible.[1] Currently, there has been a lot of interest in terms of using directional antennas in ad hoc networks [4]-[10]. In most of the work the use of directional antennas in static networks is considered [15], [9], [4], [2]. There has been some limited work on dealing with mobility when directional antennas are deployed [15], [21], [10]. However, these schemes rely on omni-directional transmissions of control messages by nodes that try to reconnect with neighbors that move out of their angular range. Furthermore, most of the previously proposed schemes restrict themselves to either only directional transmissions or directional receptions. The inability of exclusively using directional antennas for both the transmission and reception of all MAC layer frames (control or data) results in two major problems: (a) the spatial re-use benefits are reduced due to the invocation of omni-directional communications and (b) the use of omni-directional receptions for certain packets and directional receptions for others leads to an inherent *asymmetry in range*. This phenomena can exacerbate the hidden terminal problem [4] and leads to a significant penalty in throughput.

A challenge associated with the exclusive deployment of directional antennas for all communications in mobile networks is that, due to the angular reduction in range in comparison to the omni-directional case, it is important for a node to *poll* each of its neighbors periodically to ensure that the neighbor's motion is tracked. The MAC protocols proposed thus far either completely ignore mobility or use omni-directional transmissions or receptions (thus inflicting the asymmetry in range problem) of HELLO messages to identify neighbors.

In this work, we propose a new MAC protocol for *mobile* ad hoc networks that addresses the issues mentioned above in an integrated way. We call our protocol PMAC for *Polling-based MAC protocol*. PMAC exclusively uses directional antennas for the transmission and reception of all the frames. Furthermore, the protocol facilitates the *discovery* of one-hop neighbors, and using polling, the *maintenance* of links to the discovered neighbors until they are outside the possible radial range of the node. Polling is also used to schedule the transmissions and receptions of information. At the scheduled time, the transmitter and the receiver nodes point their antenna beams towards each other and carry on the communication exclusively in directional mode. We wish to point out that the motivation for our cross-layer approach to integrate neighbour discovery with the MAC protocol stems from the close interaction and dependence between the two functionalities. On the one hand, neighbor discovery and maintenance is critical for the efficiency of the MAC protocol. On the other hand neighbor discovery and maintenance relies on the exchange of messages and thus, on the underlying MAC protocol. When omni-directional receptions are used, the neighbor discovery process becomes trivial; a simple broadcast can reach all the nodes. However the problem becomes more complex with fully directional communications. In order to ensure that mobile nodes are tracked, a neighbor maintenance strategy needs to be imbibed with the MAC layer.

Manuscript received July 4, 2005; revised May 6, 2006 and October 26, 2006; accepted October 30, 2006. The associate editor coordinating the review of this paper and approving it for publication was J. Zhang. This work is supported in part by the U. S. Army Research Office under the Multi-University Research Initiative (MURI) grant W911NF-04-1-0224, the NSF CAREER Grant 0237920, and the NSF NRT grant 0335302.

The authors are with the Department of Computer Science and Engineering, University of California, Riverside, 900 University Ave., Riverside, CA 92521 USA (e-mail: {jakllari,wluo,krish}@cs.ucr.edu).

Digital Object Identifier 10.1109/TWC.2007.xxxxx.

[1]As an example, in the 5.8 Ghz ISM (Industrial, Scientific, Medical) band, an 8-element cylindrical array will have a radius of only 3.3 cm [15]

The rest of our paper is organized as follows. In Section II, we discuss prior related work on MAC protocols for use with directional antennas in ad hoc networks and state how our work differs from prior efforts. We describe the antenna models that are considered in our work in Section III. We describe our polling based MAC protocol (PMAC) in detail in Section IV. We perform a simple analysis to understand how quickly a node can discover its neighbors using our protocol and we present this analysis in Section V. A layout of our simulation framework, the parameters used and the metrics that are of interest are provided in Section VI. We present our simulation results and discuss these results in Section VII. Our concluding remarks form Section VIII.

II. Related Work

There has been a lot of interest on the design of MAC protocols for use with directional antennas in mobile ad hoc networks [6], [10-13]. Efforts on routing and broadcasting using directional antennas in ad hoc networks are seen in [3], [17], [12]. In [19] a framework for a unified approach to MAC design with various types of smart antennas is presented.

Ko *et al* [9] propose a MAC protocol for use with directional antennas in static ad hoc networks. The scheme uses omni-directional transmissions/receptions of control messages. Furthermore, the experiments that the authors perform are based on the assumption that the sender knows the physical location of the receiver by means of GPS. Nasipuri *et al* [13] propose a MAC protocol for directional antennas based on carrier sensing. They examine the performance in simulations where speeds of up to 3 m/s are considered. However, neither do they consider higher speeds nor do they provide methods to track the mobility of users. Takai *et al* [21] propose the use of directional virtual carrier sensing to access the channel. The scheme relies on angle of arrival (AOA) caching to learn the destination's position; if no information is available in the cache, packets will be transmitted omni-directionally. If the mobility is high, the cached information becomes stale quickly (especially if the antenna beamwidth is small) and the protocol would have to frequently resort to omni-directional transmissions, the effects of which are not investigated. Bao and Garcia-Luna-Aceves [2] present a new link activation scheduling scheme by using a specified MBAA (Multi-Beam Adaptive Array) antenna pattern which can support simultaneous transmissions and receptions. Choudhury *et al* [4] propose the multi-hop RTS protocol which is based on previous work in [9]. The authors identify that the radial range possible with directional transmissions combined with directional receptions is longer than that possible with directional transmissions combined with omni-directional receptions. Since the control packets (RTS and the CTS packets as in IEEE 802.11) are received omni-directionally, a multi-hop RTS is used to establish handshakes with distant neighbors that cannot be reached if those neighbors were to receive omni-directionally. Wang and Garcia-Luna-Aceves [24] discuss the interactions between spatial reuse and collision avoidance, and point out that omni-directional transmission of control packets may nullify the spatial reuse benefits.

Korakis *et al* [10] propose the use of a circular and directional RTS (CRTS) message for locating and tracking of neighboring nodes. By doing so, they are able to use directional transmissions only and thus achieve a higher coverage range. However, the scheme deploys omni-directional receptions which, in turn, can reduce the possible increase in coverage range and cause the asymmetry in gain problem [4].

Vasudevan *et al* [23] propose algorithms for neighbor discovery in directional antenna equipped ad hoc networks. The authors compute optimal parameter settings for neighbor discovery by considering *clique* topologies. The value of this work is in its theoretical contributions and less so in practice since, it assumes that a node has an "a priori" estimate of the number of its one hop neighbors that in practice is not available or is as difficult to acquire. Furthermore, this work does not consider neighbor maintenance, essential for mobile networks, nor the required design changes to the MAC protocol. Our work fully addresses the challenges of neighbor discovery, maintenance and medium access control while keeping the design simple and easy to implement in practical settings.

In spite of these previous efforts, as mentioned earlier, there are still two significant problems that arise with the deployment of directional antennas that remain unresolved.

1. **Full exploitation of directional transmissions:** Most of the solutions proposed, do not eliminate the requirement of omni-directional transmissions and/or receptions of control packets. This has three consequences. (a) it limits the frequency re-use significantly, (b) it limits network connectivity, since the nodes are required to be within the omni-directional radius (fully directional communications can in some cases help bridge possible partitions that may arise with simply omni-directional communications) and (c) creates the problem of asymmetry in gain which in turn can decrease the network throughput [10].
2. **Locating and tracking neighbors under mobility:** In most of the previous work, the assumption that each node *knows* a neighbor's physical position so as to beamform correctly in the appropriate direction, is made. However, under mobility, the MAC protocol should offer a mechanism for a node to locate and track its neighbors. As mentioned, Korakis *et al* [10] use the CRTS approach to solve this problem. In [18] an algorithm that relies on omni-directional receptions, for neighbor discovery is proposed. However, these approaches suffer from a manifestation of the asymmetry in range problem.

To the best of our knowledge there is no MAC protocol that addresses both of these issues. The motivation to do so guides the design of the PMAC, described in Section IV.

III. Antenna Model

In our studies of PMAC, we employ an electronically steerable antenna which has a single beam and can target its boresight to any position within its range. We assume that a node can either transmit or receive directionally at any given instance in time, but not both. The antenna beamwidth can be as narrow as 5° [2]. For the purposes of our study, we vary the antenna beamwidth as a parameter and use values from 30°

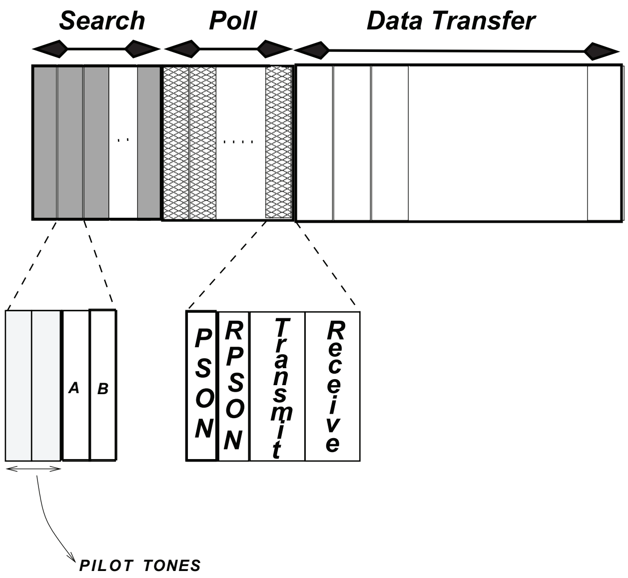


Fig. 1. Frame structure in PMAC.

to 180° in various experiments. However, all the nodes in the network, for a particular simulation experiment, use antennas with identical fixed beamwidths.

Our antenna system is assumed to function in two modes. When a node is searching for neighbors (to be discussed in detail in Section IV), the antenna functions as a switched beam antenna. It uses a fixed beamwidth θ, and it could scan one of K fixed directions where $K\theta = 2\pi$. The scanning includes both transmitting its own search signal and listening to other nodes' search signals. Once a node establishes a connection with a neighbor, it communicates with that neighbor in the polling and data transfer phases (to be discussed). In these phases, the antenna determines and tracks the neighbor's position by continuously monitoring the DOA (direction of arrival) or the AOA (angel of arrival) from the signals received from that neighbor. The node may accordingly update its antenna's weighting coefficients to point its main lobe towards the destination. We assume that the time taken by an antenna to adapt its weighting coefficients is negligible in comparison to the time duration of a slot (to be defined) used for medium access control as in previous work [11]. With advances in signal processing technology, antennas have evolved and could potentially have the ability to nullify the side-lobe interference. In our analytical study, without loss of generality, we assume that the gain of the side lobes of the antennas is negligible.

IV. The Polling-Based MAC Protocol

In this section we describe our polling based medium access control protocol (PMAC) in detail. We use a scheme in which time is divided into contiguous frames. For our scheme, therefore, it is essential that each node in the network be synchronized with its neighbors in time. This requirement is not unreasonable [2] since the nodes can synchronize during their polling slots (appropriate guard bands might be required) . Furthermore, time-synchronization methods have been proposed for ad hoc networks [16]. It is possible that the nodes have different views of time as long as they are aware of the clocks of each of the neighbors that they communicate with.

The Frame Structure: The MAC protocol will allow a node to exist in one of three states;

- *search* state in which it searches for new neighbors
- *polling* state in which it polls known neighbors
- *data transfer* state wherein information is actually transferred.

As mentioned earlier, time is divided into contiguous frames as shown in Fig. 1. Each frame is divided into three segments.

The first segment is called the search segment. In each of the time-slots in the search segment, a node points its antenna (by appropriately adjusting its antenna weighting coefficients) in a randomly chosen direction. If communication is established with a new neighbor, messages are exchanged (the mechanism will be discussed later) and the two nodes agree to communicate on a regular basis in one of the slots in the polling segment, consistently, in subsequent frames.

In the polling segment of each subsequent frame, the nodes schedule data transfers (in either direction) in the data segment. The communication in the particular polling slot takes place irrespective of whether the nodes have any data to exchange. Receiving this message from a neighbor helps a node adjust its antenna weighting coefficients for that neighbor, i.e., track the motion of the particular neighbor. The specific number of polling slots that PMAC uses is an important parameter. In prior work on topology control in wireless networks, the number of neighbors that a node should communicate with, and/or the power level to be used for such communications are computed for optimizing performance metrics such as connectivity. Details can be found in [7], [6], [20] and the references therein. Takagi and Kleinrock in their classic work [20] have computed the number of neighbors that a node needs so that the progress of the packets towards their destinations is optimized. That number was computed to be eight. In later work, Ni and Chandler [14] have shown by means of simulations that 6-8 neighbors can make a small network connected with high probability. Given these prior studies, in our implementation of PMAC we have chosen the number of polling slots to be eight.[2]

In the data transfer segment, nodes finally exchange the data packets according to the schedule set during the polling segment.

The frame duration is a system parameter that should be appropriately chosen based on the expected mobility patterns. If nodes move with high speeds then the frame size should be small since, a node should poll its neighbors frequently in order to keep track of their positions. On the other hand, if there is little mobility, the frame size could be chosen to be longer. Clearly, in a dynamic network, where nodes move with different speeds, an optimal choice for the frame duration is difficult. If a neighbor moves out of the angular range of a node under consideration within a

[2]We wish to point out that recent work by Xue and Kumar [25] has shown that no constant number is sufficient to guarantee the connectivity of a wireless networks. However, this result is for asymptotically large networks. For large networks Gupta and Kumar [5] have shown the per node throughput to converge to zero. Such large scale wireless networks are however, unlikely in practice.

frame period, the node will have to rediscover the neighbor using the search slots. The trade-offs in terms of choosing a long versus short frame size will be discussed in Section VII.

Search Slots: In the search segment, each node searches for new neighbors. In each search slot, the antenna is pointed in a randomly chosen direction. Each slot can be further divided to four sub-slots in each of which a particular sub-operation is performed. In the first sub-slot, the node would randomly choose to transmit its pilot tone (or identifier) *or* choose to receive. Both the transmissions and receptions are directional. If the node chose to receive in the first sub-slot it would transmit its pilot tone in the second sub-slot and vice versa. If there is a neighbor who has tuned his antenna in the same direction (in order to receive), when the node transmits, this neighbor will hear the pilot tone; it would correspondingly respond in second sub-slot. On the other hand, if both nodes (as above) decide to perform the same function (either a transmission or a reception), they will not be able to discover each other. Furthermore, we wish to point out that if two or more nodes decide to transmit in the same direction and create interfering transmission beams, the transmissions will collide at the receiver. In this case, the nodes involved will have to try again to discover each other.

Sub-slots 3 and 4 are labelled sub-slot A and sub-slot B in Fig. 1. In sub-slots A and B, the nodes that successfully exchanged pilot tones, exchange *a list* to specify the slots in their corresponding polling segments that are unused. The node that transmitted the pilot tone in the first sub-slot uses sub-slot A for transmitting its list; the other node of the pair transmits its list in sub-slot B. The two nodes with the help of each other's lists, then, identify a polling slot which can be used for scheduled polling. In our current implementation of PMAC, the pair of nodes pick the first common free polling slot for communicating on a periodic basis.

Polling Slots: The polling slots serve twofold: first, they allow two nodes to re-establish contact periodically so that they can track each other and ensure that the link is maintained. Second, they can be used in order to schedule data transfers in the third part of the frame. Once, the nodes agree upon a polling slot (as described earlier), they communicate in the same slot periodically frame after frame until they cannot communicate with each other due to their moving out of each other's radial range. In a given frame, in the particular chosen polling slot, the nodes steer their antennas in the direction in which they had communicated with each other in the previous frame. Upon re-establishing the connection, the antenna directions are further tuned in order to maximize the signal strength with respect to each other.

If the beamwidth is large (in which case the interference reduction capabilities of the directional antenna are diminished considerably) or in some rare scenarios in which new neighbors move into the vicinity due to mobility, a collision can occur in a polling slot. In order to ensure that the collision is detected, we use a control message exchange as in the IEEE 802.11 MAC protocol. The PSON message (for Polling Slot ON) is similar to the RTS (Request to Send) message used in the IEEE 802.11 MAC protocol and the RPSON (for Response to Polling Slot ON message) is similar to the CTS (Clear to Send) message. In the event of two successive collisions (of any combination of the PSON and the RPSON messages), the two communicating nodes attempt to choose another polling slot. The nodes piggyback a list of their free polling slots at the end of each data exchange between them (in the data transfer part of the frame). Upon experiencing a collision, the nodes attempt to use the last common free slot[3] (as per the list) in order to communicate with each other. If this were to fail they resort to a search. More sophisticated schemes for ensuring that the two nodes can quickly reconnect can be designed. However, since we expect this to happen infrequently, especially with small beamwidths, we use this simple scheme for the purposes of this work. We alternate the order of transmission of polling messages. As an example, if A transmits the PSON to C in frame n, then C would transmit the PSON to A in frame $(n+1)$. The node that sent the PSON, transmits in the first part (transmit slot) of the polling slot and receives information in the second part (receive slot).

Each node announces, during its polling slot, the next data packet that it needs to send and its length. It also indicates the available instances in the data frame when it would be able to send or receive any data packet from the corresponding neighbor. If the node were to transmit in the first part of the polling slot, it first takes into account its own data packet's transmission before accounting for the neighbor's transfer in the data transfer portion of the frame. If on the other hand, the node was the recipient in the first part of the polling slot, then it accounts for the transfer of its neighbor's data before accounting for its own. Accordingly, each node *schedules* the announced data transmission/reception in the data transfer part of the frame. The data transfer is scheduled at the earliest time possible in either the current frame or in a future frame (in the corresponding data transfer segment) depending upon previously scheduled communications of each of the communicating nodes.

Data Transfer: The scheduled data transfers take place in the data transfer portion of the frame. If a given node has a data transfer scheduled with a neighbor at a particular time (in a previous polling slot), the node points its antenna in the direction of the neighbor. Similarly, the neighbor would have pointed its antenna in the direction of the node under discussion. The scheduled packets are then transferred from the sender to the receiver.

An RTS and CTS message are included prior to the data transfer in order to detect possible rare collisions. These collisions could occur either due to scheduling conflicts that arise due to large beamwidths or due to mobility effects (a new neighbor moves into the vicinity and has a conflicting communication scheduled in the same direction). These RTS and CTS messages are transmitted directionally. Note that the data packets could be of arbitrary length (in terms of a number of slots of some basic size); thus, packet transmissions

[3]Using a deterministic strategy ensures that utilized slots are grouped together. This in turn can increase the probability of two nodes finding a common slot for rendezvous. In the case of completely random strategy, since there is no grouping of any sort, the probability of finding a common free slot may be expected to be lowered to some extent.

are asynchronous within this part of the frame. We mean asynchronous in the sense that packet transmissions can begin at any basic slot boundary and the packet sizes are variable in terms of the number of basic slots. Typically the receipt of an RTS and CTS message of a different transmssion precludes a node from performing its own previously scheduled data transfer if it detects the possibility of causing interference. The node that detects a conflict simply refrains from sending its RTS or CTS message at the pre-scheduled instant. The neighbor with whom the communication was scheduled infers that there has been a conflict in scheduling. The pair will then attempt to reschedule the data transfer in conflict at a later time, using the polling segment in the following frame. Note that the occurrence of such events are rare if the antenna beamwidth is sufficiently small.

Benefits of PMAC: Our schemes either completely eliminate or alleviate the problems that are present in other previously proposed schemes.

Eliminating the problems due to range asymmetry: As mentioned earlier, since our protocol uses only directional transmissions it avoids the problem of asymmetry in gain. Furthermore the use of fully directional communications provides an increase in directional range (identified in [4]) that can benefit routing [3] in terms of computing shorter paths and bridging potential partitions that may exist when only omni-directional communications are used.

Handling Mobility: Previous schemes either completely ignore mobility or use omni-directional transmissions in order to detect neighbors that move out of angular range. In our scheme since we poll neighbors periodically, we ensure that each node is continuously aware of its neighbors' positions. Even in the presence of bursty traffic wherein a node may not exchange data with a neighbor for extended periods in time, the polling of the neighbor helps the node track the neighbor.

Reducing the effects of deafness: Many of the previously proposed schemes suffer from the problem of deafness [4]. When two nodes exchange control messages (RTS and CTS) directionally, a different neighbor of one of these communicating nodes may not hear the directional exchange. Later, during the data exchange between the nodes, this neighbor, being unaware of the data exchange might attempt to initiate communications with one of these nodes. However, clearly, it would not receive a response. This effect is referred to as *deafness*. As a consequence of the effects described, the neighbor would then back-off. The problem could repeat itself and may lead to the neighbor incorrectly concluding that a link failure has occurred. Since, our protocol is based on scheduled communications as opposed to asyncrhonous random access based communications, deafness does not occur.

V. An Analytical Model to Compute Neighbor Discovery Time

In this section, we develop a simple analysis to find an expression for the probability that, during the initialization of the network, a node takes J frames to find and connect with a particular neighbor. We represent this probability by P_J. We also compute an expression for the probability that all of the node's neighbors are found within J frames. We make a set of assumptions and define certain parameters and metrics.

Assumptions:

- The network remains static for the duration of this preliminary search. This is reasonable since we expect the network initialization time to be fairly small as compared to the time it takes for the topology of the network to change drastically.
- The interference experienced by a receiver is limited to those interfering transceivers that are in its directional range and whose transmit beams are pointed toward that receiver. In such a case, where a receiver is the target for multiple such directional transmissions, we assume that a collision is experienced by the receiver.
- We begin counting frames from frame 1 upon initialization. Thus, the frame i is the i^{th} frame from network initialization.
- The notation *"finding a neighbor"* corresponds to first instance when the node discovers the neighbor.[4]

Parameters:

- The beamwidth of a directional antenna for either transmissions or receptions is fixed and is $\frac{2\pi}{K}$, where K is a system parameter. In other words, a node is capable of pointing its antenna in one of K fixed directions.
- The node density, i.e., the number of nodes per unit area, $= \sigma$.
- The range of the antenna beam is r units (in distance).
- The number of nodes that are within a transmit or receive beam is then $m = \frac{\pi r^2}{K}\sigma$. We assume for the ease of analysis, that for any node the number of neighbors within the node's transmit or receive beam is fixed and equal to m.
- The number of search slots per frame (a design parameter): η.

Metrics:

- s: the probability that a node finds a particular given neighbor in a particular slot.
- f: the probability that a node finds a particular given neighbor in a particular frame.
- F_i: the probability that a node finds a particular neighbor in exactly the i^{th} frame.
- P_J: the probability that a node discovers a particular neighbor in at most J frames.
- $P_{m,J}$: the probability that a node discovers all m neighbors within an angular sector in at most J frames.
- $P_{k,J}$: the probability that k neighbors are discovered within an angular sector in at most J frames, where $k \leq m$.

In order for a particular node (say node A) to discover node C in a particular slot, its antenna should be pointed towards C. The probability of this event is $\frac{1}{K}$. Similarly node C should point its antenna towards node A. The probability of this event is $\frac{1}{K}$ as well.[5] Furthermore, it is necessary that one of them

[4] Note that in subsequent search slots, the two nodes may synchronize their transmit and receive antennas; however, since they already have found each other, this is irrelevant.

[5] In this work we tradeoff performance for simplicity by choosing a constant probability p. Adaptively choosing p to optimize the search would be difficult and complex since one would need to estimate node density.

should be in the transmit mode and the other should be in the receive mode. As mentioned earlier, a node could choose either the transmit or the receive mode with probability $\frac{1}{2}$. None of the other $(m-1)$ nodes that can cause interference to the nodes' communication should be transmitting at the same time. Thus,

$$s = 2 \times \frac{1}{2K}\frac{1}{2K}(1-\frac{1}{2K})^{m-1} = \frac{1}{2K^2}(1-\frac{1}{2K})^{m-1}. \quad (1)$$

Correspondingly, the probability that A finds C in a particular frame that consists of η search slots is: $f = 1-(1-s)^{\eta}$. Accordingly, the probability that A finds C exactly in frame i is given by:

$$F_i = f(1-f)^{i-1}. \quad (2)$$

Thus, the probability of A finding the node C in one of the first J frames (since the event that C is found in frame i is mutually exclusive[6] from the event that C is found in frame k for $i \leq J$ and $k \leq J$ and $i \neq k$)

$$P_J = \sum_{i=1}^{J} F_i = \sum_{i=1}^{J} f(1-f)^{i-1}. \quad (3)$$

Simplifying this expression, we get

$$P_J = 1-(1-\frac{1}{2K^2}(1-\frac{1}{2K})^{m-1})^{\eta J}. \quad (4)$$

Since the event of finding a particular neighbor is independent of the event of finding another particular neighbor,[7] we compute the probability of finding k neighbors within the particular angular sector in J frames to be:

$$P_{k,J} = \begin{pmatrix} m \\ k \end{pmatrix} P_J^k(1-P_J)^{m-k}. \quad (5)$$

Then, the possibility of finding all m neighbors within the angular sector in J frames is given by:

$$P_{m,J} = (1-(1-\frac{1}{2K^2}(1-\frac{1}{2K})^{m-1})^{\eta J})^m. \quad (6)$$

We plot the probability of a node finding all of its neighbors in a particular sector versus the antenna beamwidth for an example topology in Fig. 2. In this topology, we assume 12 nodes in total, placed uniformly around the node that we're interested in. Thus, for different antenna beamwidths, m varies. For example, given a 60° antenna beam, $m = 2$. For this experiment, we fix the number of search slots in the frame (SSL) to be 20.

From Fig. 2, we see that for the assumed simple topology, the larger the antenna beam, the higher the probability that a particular neighbor will be found within a specified number of frames (J). This result is expected since the larger beam corresponds to fewer angular segments, which leads to a greater likelihood of neighbors aligning their antennas towards each other. Note that a higher antenna beamwidth could result in a greater likelihood of collisions; however, we find that this effect is less prominent.

[6]Without loss of generality, let us assume that $i < k$. If a node A finds node C in frame i, it means (a) node A had not discovered node C earlier (Equation 2) and (b) that it will no longer look for node C in frame $(i+1)$ and so on. Thus, the event that node A will find node C in frame k cannot occur. If on the other hand, node A finds node C in frame k, it would mean (from Equation 2) that node C was not discovered in a prior frame (including node i). Thus, the two events (node A discovers node C in frame k and node A discovers node C in frame i) cannot occur simultaneously i.e., the events are mutually exclusive.

[7]Note that a node that is already discovered still contributes to interference effects. Thus, the interference effects experienced and the possibility of collision is the same throughout. This is the reason why the discovery of a node is independent of the discovery of any other node.

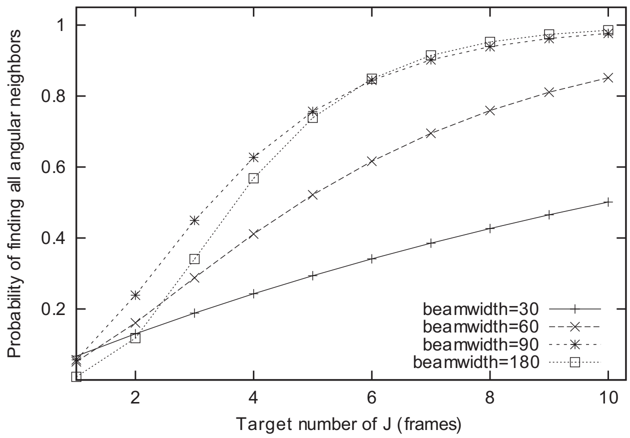


Fig. 2. Probability of finding all angular neighbors within J frames.

VI. Simulation Model and Framework

Our simulations are performed in Opnet, version 10.0 [1]. The physical layer models used assume that there is a propagation delay between a transmission and a reception; this delay is dependent on the distance (D) between the transmitter and the receiver and the wavelength (λ) of the radio waves. The waves suffer an attenuation given by [22]: $L_p = (\frac{\lambda}{4\pi D})^2$.

The received power P_r can be computed based on this propagation model to be: $P_r = P_t \times L_p \times G_t \times G_r$, where P_t represents the transmit power, G_t and G_r represent the transmit and receive antenna gains respectively, and L_p is the propagation loss. We choose a fixed antenna gain. The OPNET simulator correspondingly computes the radial range depending upon the chosen beamwidth. In addition to the main lobe with the chosen beamwidth, the antenna pattern generator of OPNET creates side lobes, which together form a bulb of appropriate size at the base of the main lobe.

The Signal to Noise Ratio (SNR) is then computed as the average power in the received information signal to the accumulated average power from all interference and noise sources. Assuming QPSK (Quadrature Phase Shift Keying) modulation, the bit error rate (BER) is then estimated based on the computed SNR, and a probabilistic insertion of errors in the packet based on the computed SNR is carried out. If the packet is error free, it is deemed to be successfully received.

We place nodes in a two dimensional flat terrain area. We either choose specific topologies (as specified) or place nodes randomly. Each node generates constant bit rate (CBR) traffic and the source-destination pairs are chosen randomly at the beginning of the simulation and stay unchanged for the duration of the simulation. The Random Waypoint is used as the mobility model. We choose a speed of 2m/s to represent a pedestrian environment and a speed of 10m/s to represent vehicular environments.

We denote the number of search slots in a frame by Search Segment Length (SSL), the number of poll slots by Poll

TABLE I
SIMULATION PARAMETERS

Search Segment Length (SSL)	20 (slots)
Poll Segment Length (PSL)	8 (slots)
Data Transfer Segment Length (DTSL)	800 (slots)
Packet size in slots 1,2 in Search Segment	10 bytes
Packet size in slots A,B in Search Segment	20 bytes
Poll packet size	20 bytes
Poll reply packet size	14 bytes
PSON packet size	20 bytes
RPSON packet size	14 bytes
RTS packet size	20 bytes
CTS packet size	14 bytes
Data packet size	512 bytes
Frame size	1.64 seconds
Data Transmission Rate	2 Mbps
Channel frequency	2.4 Ghz
Antenna gain	20 db

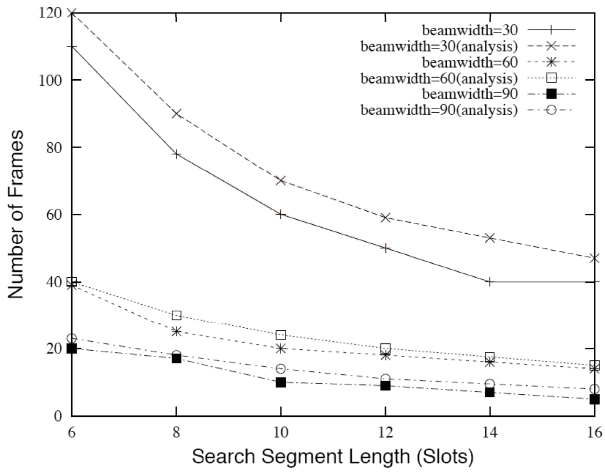


Fig. 3. Number of frames required for center node to find all possible neighbors in star topology.

Segment Length (PSL), and the number of slots in the data transfer segment (each slot is equal in size to that of a data packet) by Data Transfer Segment Length (DTSL). We use the system parameters listed in Table 1 unless specified otherwise. The value of PSL, as discussed in Section IV, is set to eight.

In each polling slot we limit the maximum channel time that can be reserved in the data transfer part to at most the time taken to transmit 400 packets (each of 512 bytes). We impose this restriction since we do not want a single node to dominate channel access. Note that reservations can be made for data transfers in subsequent frames. We also require that reservations be made for at most four frames in advance. This restriction is necessary since, in conditions of mobility, if a node is allowed to make reservations in the too distant future, it may actually move out of range by the time that the reservation is honored. We are interested in the following performance metrics:

- **Total Network Throughput**: The number of packets successfully transported at the MAC layer per unit time. Since PMAC incorporates the function of neighbor discovery upon initialization, we take care to exclude the initial time duration in our simulations to account for this phase of operations.[8]
- **Per node Channel Utilization Ratio (*CUR*)**: The fraction of time that a node either transmits or receives useful information over the total simulation time.
- **Fairness:** We define fairness as per Jain's fairness index defined in [8]. For any given set of node throughputs, $(x_1, x_2, ..., x_n)$ the following function is used as the fairness index: $f(x_1, x_2, ..., x_n) = \frac{(\sum_{i=1}^{n} x_i)^2}{n \sum_{i=1}^{n} x_i^2}$. The fairness index will lie between 0 and 1. If the protocol gives the same throughput to every user, i.e., ideal fairness, than the fairness index will be 1. If at the other extreme, one user gets all the throughput, the fairness index will be 1/n.

[8]This initial duration was observed to be approximately 2 seconds on average.

VII. SIMULATION RESULTS AND DISCUSSIONS

In this section, we evaluate the performance of PMAC in terms of the defined metrics. We also study the sensitivity of the performance to various system parameters. Each of our simulations is run for 500 seconds. In each graph, every point is an average computation over 10 simulation runs.

A. *The Neighbor Discovery Process*

In our first experiment we consider a star topology, one node is at the center and eight other nodes are uniformly distributed at distance one from the center, to study PMAC in terms of the efficiency in discovering neighbors at initialization. We had a simple analytical model in a previous section to compute certain probabilistic metrics that quantify the efficiency of the neighbor discovery process. Our simulation experiments with the realistic channel models provide an estimate of the time duration taken by the node at the center of the star topology to discover all of its neighbors. We vary the number of slots in the search segment as well as the antenna beamwidth and compute the number of frames required by the node in order to discover its neighbors. Along with the simulations results, we have also depicted the number of frames required to guarantee that at least 95% of the neighbors are discovered as per Eq. (7) of Section V.

As shown in Fig. 3, with an increase in SSL, the number of frames that is required in order for the node to discover all of its neighbors decreases. When $SSL = 20$, the node at the center takes, on average, 5 frames to find all of its neighbors by using a 90° antenna beam. If the antenna beam is 60°, the node takes about 10.5 frame durations on average. Clearly, the smaller the beamwidth, the higher the delay incurred in the initial discovery process. As an example, with a 30° antenna beam, when $SSL = 6$, the required number of frames is more than 100. Note that with a 1.64 second frame size, this translates to less than 3 minutes. This might be acceptable since this search process is only required during

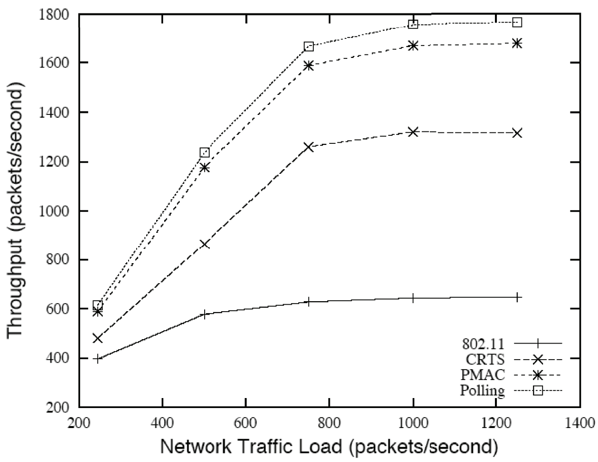


Fig. 4. Total network throughput vs. network traffic load in stationary random topology.

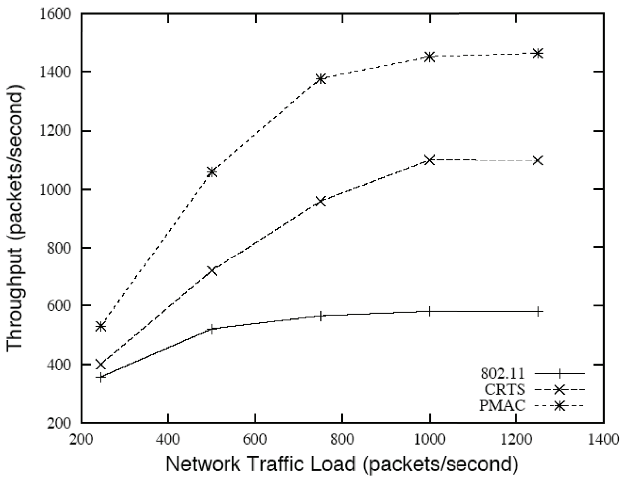


Fig. 5. Total network throughput vs. network traffic load in mobile random topology.

the initialization phase.[9] Furthermore, note that in order to ensure strong connectivity, a node does not need to discover all of its neighbors. Data transfer can in fact be performed to a neighbor that is not discovered via a multi-hop route from one of the neighbors that is already discovered.

It is of interest to point out that the results from the simulations are very close to the analysis. This demonstrates the validity of our assumptions in the analysis.

B. *Performance in Terms of Throughput*

In this subsection we present the performance of our protocol in terms of total network throughput which we defined in Section VI.

We compare the performance of our protocol, PMAC, with that of the IEEE 802.11 MAC and the CRTS scheme proposed in [10]. The reason we choose CRTS is because it is the only MAC protocol that, as PMAC, integrates the node discovery process as a part of the MAC layer. Furthermore, it has been shown in [10] that the scheme outperforms prior proposed schemes.

In addition to comparing the performance of PMAC with that of the aforementioned protocols, in order to quantify the overhead due to of neighborhood discovery and maintenance, we compare the throughput of PMAC , against a idealized "lighter" schedule-based MAC protocol; we refer to this protocol as Polling. The Polling protocol does not perform any periodic searching, instead the searching is performed only at the initialization (this overhead is not included in the performance results) and no other searching is performed – the simulations are carried out on static topologies . A node polls its neighbors only for scheduling data transmissions in the *Data Transfer* part of the frame. Dynamic scheduling through polling allows for more efficient use of the channel given the bursty nature of the traffic in most computer networks. At the same time, since every node polls only a small subset of its one-hop neighbors (eight in our implementation), the overhead from the polling is negligible (in our implementation the Poll Segment Length is 1% of the size of the Data Transfer Length).

For this set of experiments, we place 50 nodes randomly in a $500m \times 500m$ flat terrain. Each node deploys an electronically steerable antenna that creates a $45°$ beam. We have considered static topologies, wherein the nodes upon deployment maintain their position during the entire simulation period, and mobile topologies, wherein the nodes roam within the simulation terrain at a constant speed of 10 m/s.

The results of the case with stationary topology are depicted in Fig. 4 while those for the mobile topology are depicted in Fig. 5. As seen, PMAC and CRTS both provide significant improvements over the traditional IEEE 802.11 MAC protocol, in both topologies. However note that PMAC outperforms CTRS by as much as 33% in terms of throughput in mobile scenarios. We find that under heavy loads, CRTS suffers from the asymmetry in gain due to omni-reception of the RTS/CTS packets from the neighbors of the transmitter and/or the receiver. In [10] the authors use the circular RTS to address this issue. By doing so, they notify all the transmitter's neighbors of the intended communication; therefore the data packet transmission is protected. However since the CTS is transmitted only toward the transmitter, the receiver's neighbors are not notified and therefore the ACK is exposed to the asymmetry. Although the ACK duration is small, under heavy load, ACK collisions create significant problems. On the other hand, PMAC, using only scheduled directional communications does not suffer from these problems.

Another conclusion that can be drown from the results is that, both PMAC and CRTS offer a solution robust to mobility, given the small drop in performance observed when the nodes are mobile as opposed to being static. The robustness is due to the mechanism that both the protocols incorporate for discovering and tracking the neighboring nodes. Note that the performance for 802.11 does not drop with mobility simply because of the fact that it uses omnidirectional antennas; thus, neighborhood information can be easily maintained and

[9]The initialization phase is invoked when the network is deployed for the very first time. Our assumption is that network operations are not yet functional during initialization. Thus, during this time, it is assumed that the network is quasi-static i.e., mobility is low. Once the initialization is complete, the network is assumed to be operational and that the nodes can move with higher speeds.

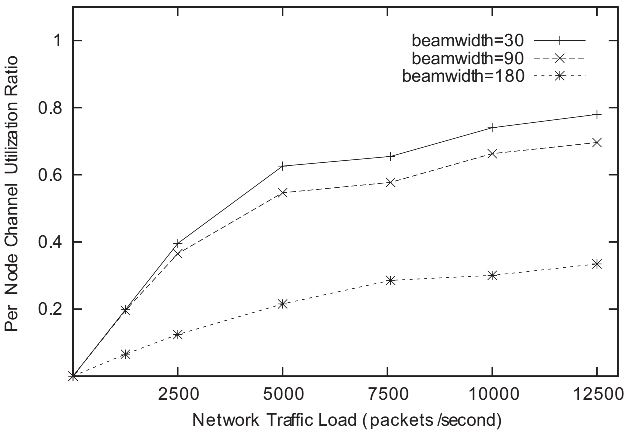


Fig. 6. Per node channel utilization ratio vs. network traffic load in stationary random topology.

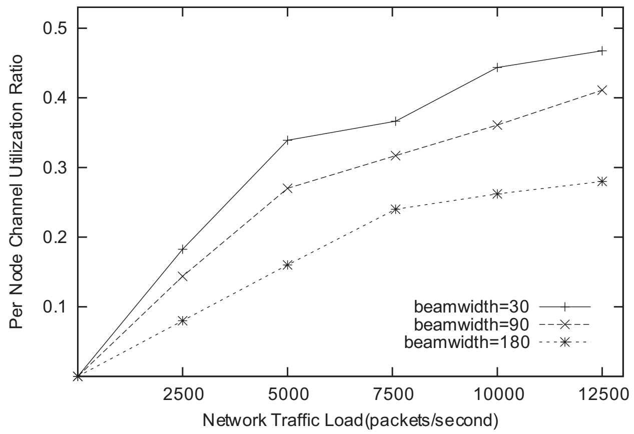


Fig. 7. Per node CUR vs. network traffic load in random topology with speed = 2m/s.

neighbors are less likely to move out of range due to mobility.

Furthermore, the results show that the idealized Polling protocol outperforms PMAC by only small percentage (4-7%). This may be expected given that, by design, the searching and polling part of PMAC are small and add little overhead.

C. Performance in Terms of CUR

In this subsection we present the performance of our protocol in terms of channel utilization ratio (CUR). We use this metric to understand where the sweet spot of operations is for PMAC. In particular, since the frame size is dictated by mobility, the metric was chosen to understand the sensitivity of the performance to the choice of frame size.

Due to the absence of frames in both the CRTS and the 802.11 MAC protocol we do not evaluate them in terms of CUR.

We place twenty five nodes, uniformly distributed in a $500m \times 500m$ flat terrain; each of these nodes generates CBR traffic with a packet size of 512 bytes. There is no mobility unless otherwise specified. Fig. 6 depicts the per node channel utilization ratio. Note that the maximum CUR increases with load and the maximum value achieved is about 78% with a 30° antenna beamwidth. Under ideal conditions, one can compute the maximum achievable CUR (an upper bound) to be:

$$CUR_{max} = \frac{data\ slot\ size\ \times\ DTSL}{frame\ size}, \tag{7}$$

where DTSL is the Data Transfer Segment Length that was defined in Section VI, while *data slot size* is the time, in terms of slots, required to transmit a data packet. This bound is computed with the assumption that the data transfer portion of the frame is completely utilized. With the parameters that we use, this turns out to be 97.56 %. There are several factors that contribute to the degradation of the actual CUR in realistic scenarios:

1) The wastage of channel bandwidth during the initialization phase (or during reconfigurations in mobility when a node loses connectivity with a few of its neighbors) wherein the node finds its neighbors.

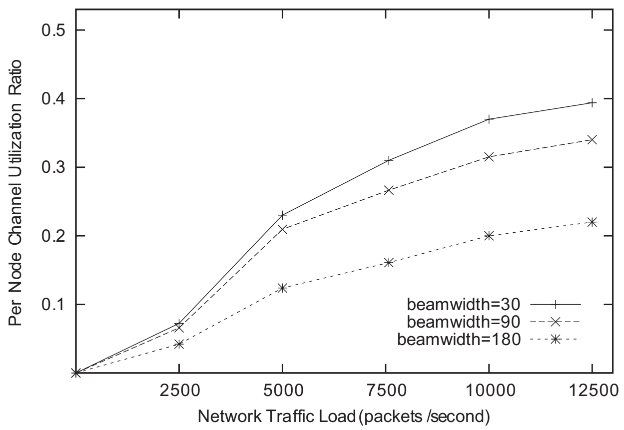


Fig. 8. Per node CUR vs. network traffic load in random topology with speed = 10m/s.

2) The scheduling is done in the order in which nodes are polled and hence, might lead to a sub-optimal utilization of the data part in the frame.[10]
3) Possible collisions of PSON and RPSON messages and to a lesser extent RTS and CTS messages.
4) Topology of the network may not facilitate complete utilization. As an example, in a star topology with 8 nodes around a central node, the CUR that is achievable by the nodes around the central node may be expected to be $\frac{1}{8} \times 100 = 12.5\%$. On the other hand, the center node may see a 100 % CUR.

The first factor is not a significant factor since in our static scenario, the initialization takes up a small fraction of time in the simulation runs. We find that most of the inefficiencies are due to topological effects and due to some inefficiency in performing scheduling. However, we point out that even with this very simple scheduling policy we achieve quite a high channel utilization efficiency for ad hoc networks.

Effects of Mobility: Fig. 7 and Fig. 8 show the CUR for the cases wherein the nodes are now mobile. In the network considered in Fig. 7 nodes move with a speed of 2 m/s

[10]More efficient scheduling policies are possible; but the study of these policies is beyond the scope of this paper.

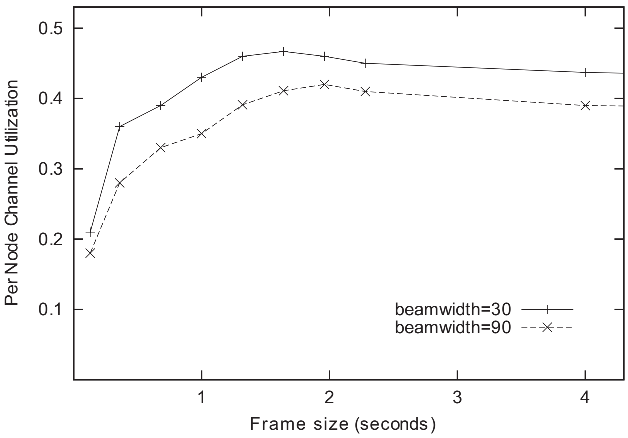


Fig. 9. Per node CUR achieved vs. frame size in random mobile topology with speed = 2m/s.

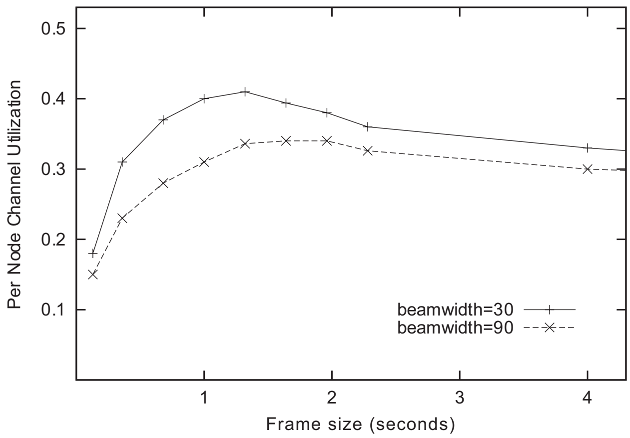


Fig. 10. Per node CUR achieved vs. frame size in random mobile topology with speed = 10m/s.

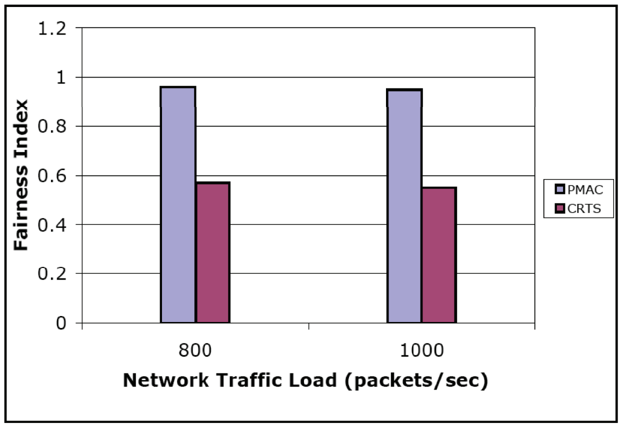


Fig. 11. Fairness index vs. network traffic load in stationary random topology.

and in the network considered in Fig. 8 the nodes move with a speed of 10 m/s. In these experiments, if a node generates packets for a neighborhood node that it is unable to communicate with any more (due to the node moving out of range), it simply discards those packets. We observe from Fig. 7 that the channel utilization degrades as compared with the stationary case by as much as 40 %. In mobile scenarios, transient topological effects (as described earlier) can cause a degradation in the CUR. An additional factor that comes into play is the possibility of an increased number of scheduling conflicts in both the polling part and the data part that cause collisions as nodes move around. The former results in a need for the rediscovery of neighbors using the search part. Reservations made for that neighbor are wasted. These factors result in an additional wastage of channel capacity. Another interesting factor to note is that mobility immediately causes a degradation in performance; the difference in the performance of the protocol with speeds of 10 m/s and 2 m/s does not seem to be significant for the chosen frame size. We wish to point out that even with this degradation, *each node* in the network (even at speeds of 10 m/s) is able to use the channel efficiently for transport of useful information about 40 % of the time. Thus, in total, all the nodes together attribute to a much higher channel utilization in the network.

As described earlier, in order to ensure that a node is tracked while it moves, the frame size is to be chosen appropriately. If the frame size is too long, nodes would frequently move out of range. If on the other hand, the frame size too small we incur a large overhead. Our goal is to examine the sensitivity of the maximum achievable CUR to variations in frame size. Towards this, we set the offered traffic load to 12,500 packets/second (which is the maximum load that the network can handle with the 2Mbps data rate.) and vary the frame size (vary the number of data slots per frame). The number of polling slots and search slots are kept fixed at 8 and 20 respectively as before.

Fig. 9 and Fig. 10 show the maximum CUR versus the frame duration with moving speeds of 2m/s and 10m/s respectively. Note that we observe the expected behavior in our experiments. For small frame sizes the CUR suffers because of excessive overhead. As we increase the frame size we see an increase in the CUR. However, for larger values the CUR in fact begins to drop. The drop is more prominent at higher speeds. With an increased beamwidth, one might expect that the polling interval can be potentially longer since nodes stay within the angular range of a node for longer periods in time. This effect is visible in Fig. 9 and Fig. 10, i.e., with a 90° beamwidth, the maximum CUR occurs at a frame size that is larger than the frame size at which the maximum CUR occurs if the beamwidth is 30°. Furthermore, note that the per node maximum channel utilization drops with increased beamwidth due to higher interference effects. Furthermore, the maximum CUR achieved drops as we increase the speed (from 2m/s to 10m/s) since, now, we would need a smaller frame size and hence incur higher overhead. We reiterate that even with this, as shown earlier, PMAC outperforms the previously proposed CRTS and the IEEE 802.11 schemes especially for small antenna beamwidths and in regular structured topologies.

D. Performance in Terms of Fairness

In Fig. 11 the results in terms of the fairness index, as a function of the load in the network are depicted. We have performed the simulations under heavy loads, when the throughput has reached the saturation point, and the same topology and traffic used for the throughput depicted in Fig. 4. Heavily loaded conditions are likely to be of interest in evaluating fairness since at lighter loads, fairness may not be an issue. We have compared PMAC to the CRTS scheme, which is the only MAC protocol that facilitates neighbor discovery for directional antennas. As one might expect, a pure TDMA approach will have a fairness index of 1 so we don't include it in the graph.

Since the CRTS is a random access scheme with back-offs, it suffers from low fairness. Due to the inherent exponetial back-off mechanism in CRTS, when a node fails to acquire the channel, it will double its backoff window. However, when finally it is able to transmit a packet successfully, it will reduce its backoff back to the minimum value. Under heavy loads, once a node is able to transmit a packet it will have much better probability of getting access to the channel again then its one hop neighbors who might have higher back-off waiting periods. On the other hand, PMAC uses a controlled polling approach to regulate the access to the channel, which, as depicted in Fig. 11, allows for very high levels of fairness.

VIII. Conclusions

In this paper, we propose a MAC protocol for use with directional antennas in mobile ad hoc networks. Our protocol overcomes the problems due to asymmetry in range when these antennas are deployed. Furthermore, it efficiently handles mobile scenarios by facilitating the discovery of new neighbors by a node and the maintenance of links to the discovered neighbors. The key idea that forms the basis for our protocol is to use a polling strategy wherein a node polls its discovered neighbors periodically; this would enable the node adjust its antenna weighting coefficients so as to continuously track its neighbors. Thus, we call our protocol PMAC for Polling-based MAC. Since PMAC uses fully directional scheduled communications, it also eliminates some of the problems that arise due to the use of asynchronous random access MAC protocols that have been proposed for use with directional antennas. Specifically, with PMAC, we no longer have the problems due to (a) asymmetry that arises as a result of intermixing omni-directional and directional transmissions/receptions and (b) deafness.

We perform analysis and extensive simulations to evaluate our protocol and we find that we achieve an extremely high *per node* channel utilization of up to 80 % in static scenarios and up to 50 % in mobile scenarios. Our protocol outperforms the IEEE 802.11 MAC and the previously proposed CRTS scheme that has been shown to outperform most other MAC protocols for use with such antennas.

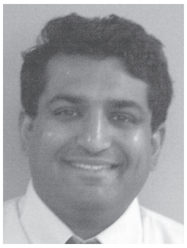
Acknowledgments: The authors wish to thank the editor Dr. Junshan Zhang and the reviewers for their constructive comments, which helped in improving the quality of our paper.

References

[1] Opnet user's documentation. [Online.] Available: http://www.opnet.com
[2] L. Bao and J. J. Garcia-Luna-Aceves, "Transmission scheduling in ad hoc networks with directional antennas," in *Proc. ACM MobiCom*, Sep. 2002, pp. 48–58.
[3] R. R. Choudhury and N. H. Vaidya, "Impact of directional antennas on ad hoc routing," in *Proc. IFIP PWC*, Sep. 2003.
[4] R. R. Choudhury, X. Yang, N. H. Vaidya, and R. Ramanathan, "Using directional antennas for medium access control in ad hoc networks," in *Proc. ACM MobiCom*, Sep. 2002, pp. 59–70.
[5] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inform. Theory*, vol. 46, no. 2, pp. 388–404, Mar. 2000.
[6] L. Hu, "Topology control for multihop packet radio networks," *IEEE Trans. Commun.*, vol. 41, no. 10, pp. 1474-1481, Oct. 1993.
[7] Z. Huang, C.-C. Shen, C. Srisathapornphat, and C. Jaikaeo, "Topology control for ad hoc networks with directional antennas," in *Proc. IEEE ICCCN*, Oct. 2002, pp. 16–21.
[8] R. Jain, *The Art of Computer Systems Performance Evaluation*. New York: Wiley, 1991.
[9] Y.-B. Ko, V. Shankarkumar, and N. H. Vaidya, "Medium access control protocols using directional antennas in ad hoc networks," in *Proc. IEEE INFOCOM*, Mar. 2000, vol. 1, pp. 13-21.
[10] T. Korakis, G. Jakllari, and L. Tassiulas, "A mac protocol for full exploitation of directional antennas in ad-hoc wireless networks," in *Proc. ACM MobiHoc*, June 2003, pp. 98–107.
[11] S. V. Krishnamurthy, A. S. Acampora, and M. Zorzi, "Polling-based media access protocols for use with smart adaptive array antennas," *IEEE/ACM Trans. Networking*, vol. 9, no. 2, pp. 148–161, Apr. 2001.
[12] A. Nasipuri, "On demand routing using directional antennas in mobile ad hoc networks," in *Proc. WCNC*, 2002.
[13] A. Nasipuri, S. Ye, J. You, and R. Hiromoto, "A mac protocol for mobile ad hoc networks using directional antennas," in *Proc. IEEE WCNC*, Sep. 2000, vol. 3, pp. 1214–1219.
[14] J. Ni and S. Chandler, "Connectivity properties of a random radio network," *IEE Proc.-Commun.*, pp. 289–296, Aug. 1994.
[15] R. Ramanathan, "On the performance of ad hoc networks with beamforming antennas," in *Proc. ACM MobiHoc*, Oct. 2001, pp. 95–105.
[16] K. Romer, "Time synchronization in ad hoc networks," in *Proc. ACM MobiHoc*, Oct. 2001, pp. 173–182.
[17] 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 *Proc. ACM MobiHoc*, June 2003, pp. 88–97.
[18] M. Steenstrup, "Topology management in mobile networks with smart antennas," in *Proc. IEEE 3rd Scandinavian Workshop ADHOC Networks*, 2003.
[19] K. Sundaresan and R. Sivakumar, "A unified mac layer framework for ad-hoc networks with smart antennas," in *Proc. ACM MobiHoc*, May 2004, pp. 244–255.
[20] H. Takagi and L. Kleinrock, "Optimal transmission ranges for randomly distributed packet radio terminals," *IEEE Trans. Commun.*, vol. 32, no. 3, pp. 246–257, Mar. 1984.
[21] M. Takai, J. Martin, R. Bagrodia, and A. Ren, "Directional virtual carrier sensing for directional antennas in mobile ad hoc networks," in *Proc. ACM MobiHoc*, June 2002, pp. 183–193.
[22] T. S. Rappaport, *Wireless Communications: Principles and Practice*. Upper Saddle River, NJ: Prentice Hall, 1996.
[23] S. Vasudevan, J. Kurose, and D. Towsley, "On neighbor discovery in wireless networks with directional antennas," in *Proc. IEEE INFOCOM*, Mar. 2005, vol. 4, pp. 2502–2512.
[24] Y. Wang and J. Garcia-Luna-Aceves, "Spatial reuse and collision avoidance in ad hoc networks with directional antennas," in *Proc. IEEE GLOBECOM*, Nov. 2002, vol. 1, pp. 112–116.
[25] F. Xue and P. R. Kumar, "The number of neighbors needed for connectivity of wireless networks," *Wireless Networks*, vol. 10, no. 2, pp. 169–181, 2004.

Gentian Jakllari is a PhD candidate at the Computer Science Dept. at the University of California, Riverside. He obtained his bachelor's degree in computer science at the University of Ioannina, Greece and his M.Sc. in computer science at the University of California, Riverside. His research interests are in the field of wireless networking with emphasis on the design, evaluation and implementation of cross-layer protocols and algorithms.

Wenjie Luo got her Masters degree in Computer Science and Engineering from the University of California, Riverside. Her interests are in the use of directional antennas in wireless networks.



Srikanth V. Krishnamurthy received his Ph.D degree in electrical and computer engineering from the University of California at San Diego in 1997. From 1998 to 2000, he was a Research Staff Scientist at the Information Sciences Laboratory, HRL Laboratories, LLC, Malibu, CA. Currently, he is an Associate Professor of Computer Science at the University of California, Riverside. His research interests span CDMA and TDMA technologies, medium access control protocols for satellite and wireless networks, routing and multicasting in wireless networks, power control, the use of smart antennas and security in wireless networks. Dr. Krishnamurthy has been a PI or a project lead on projects from various DARPA programs including the Fault Tolerant Networks program, the Next Generation Internet program and the Small Unit Operations program. He is the recipient of the NSF CAREER Award from ANI in 2003. He has also co-edited the book *Ad Hoc Networks: Technologies and Protocols* published by Springer Verlag in 2005. He has served on the program committees of INFOCOM, MOBIHOC and ICC and is the associate editor-in-chief for *ACM MC2R*.