Resource Scheduling in Wireless Networks Using Directional Antennas

Imad Jawhar, *Member*, *IEEE*, Jie Wu, *Fellow*, *IEEE*, and Dharma P. Agrawal, *Fellow*, *IEEE*

Abstract—Due to a continued increase in the speed and capacities of computing devices, combined with our society's growing need for mobile communication capabilities, multihop wireless networks (MWNs), such as wireless mesh networks (WMNs), have gained a lot of interest from the research community. Quality of service (QoS) provisioning in these networks is an essential component that is needed to support multimedia and real-time applications. On the other hand, directional antenna technology provides the capability for considerable increases in spatial reuse, which is essential in the wireless medium. In this paper, a bandwidth reservation protocol for QoS routing in TDMA-based MWNs using directional antennas is presented. The routing algorithm allows a source node to reserve a path to a particular destination with the needed bandwidth, which is represented by the number of slots in the data phase of the TDMA frame. Further optimizations to improve the efficiency and resource utilization of the network are also provided.

Index Terms—Directional antennas, multihop wireless networks (MWNs), quality of service (QoS), routing, time division multiple access (TDMA).

1 Introduction

In multihop wireless networks (MWNs), a device transmits data using an omnidirectional antenna that radiates its power equally in all directions. This prevents other nodes located in the transmission range from using the medium simultaneously. On the other hand, directional antennas allow a transmitting node to concentrate its energy in a particular direction. Similarly, a receiving node can focus its antenna in a particular direction, which leads to increased sensitivity in that direction, significantly reduces multipath effects and cochannel interference (CCI), and enables spatial reuse of a given channel. This allows directional antennas to accomplish several objectives:

1. Power saving: a smaller amount of power can be used to cover the same desired range.
2. Spatial reuse: since transmission is focused in a particular direction, the surrounding area in the other directions can still be used by other nodes to communicate using the same channel.
3. Shorter routes (in number of hops): this is due to a longer range achieved by using the same transmission power as omnidirectional antennas.
4. Smaller end-to-end delay: this is due to shorter routes [1], [2], [3], [4], [5], [6], [7]. These factors provide a network whose nodes use directional antennas with the ability to reduce unintentional interference, increase the network efficiency, and enhance the communication capabilities.

Different models have been presented in literature for directional antennas [8]. An antenna-array-based directional scheme generally provides an increased antenna gain against multipath fading. A constant signal gain can be maintained in an intended direction, and the sidelobes can be adjusted to avoid the source of interference so as to reject the CCI. Consequently, the communication capacity, coverage, and quality of the wireless system can be considerably increased. In this paper, a MultiBeam Adaptive Array (MBAA) system is used [1] and is capable of forming multiple beams for simultaneous transmissions or receptions of multiple data messages in different directions.

As is the case of omnidirectional antennas, Medium Access Control (MAC) protocols for directional antennas can be classified into two categories: Contention based and contention free. The most common approach in the first category is Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). In the second category, the Time Division Multiple Access (TDMA) scheme is the most prevalent one. In [9], Ramanathan presented an analysis of the performance of multihop wireless networks with beamforming antennas. The author discusses the issues of deafness, directional exposed, and hidden terminal problems, along with the challenges that are specific to a directional antenna environment. The CSMA/CA contention-based mechanism is used by several directional MAC protocols. In [10], Choudhury et al. present an analysis of medium access control protocols using directional antennas in multihop wireless networks. In [11], Ko et al. present a MAC protocol where each node has multiple directional antennas with a single transceiver. The protocol uses an omnidirectional/directional RTS/CTS mechanism for minimizing the collisions and increasing the spatial reuse. While a CTS is heard only on one omnidirectional antenna, an RTS needs to be sent out on all antenna sectors to set up the

- *I. Jawhar is with the College of Information Technology, UAE University, PO Box 17551, Alain, UAE. E-mail: ijawhar@uaeu.ac.ae.*
- *J. Wu is with the Department of Computer and Information Sciences, Temple University, 1805 N. Broad Street, Philadelphia, PA 19122. E-mail: jiewu@temple.edu.*
- *D.P. Agrawal is with the Computer Science Department, University of Cincinnati, Box 7911, PO Box 210030, Cincinnati, OH 45221-0030. E-mail: dpa@cs.uc.edu.*

Manuscript received 17 Mar. 2009; revised 15 Nov. 2009; accepted 24 Nov. 2009; published online 24 Dec. 2009.
Recommended for acceptance by M. Ould-Khaoua.
For information on obtaining reprints of this article, please send e-mail to: tpds@computer.org, and reference IEEECS Log Number TPDS-2009-03-0125.
Digital Object Identifier no. 10.1109/TPDS.2009.171.

1045-9219/10/$26.00 © 2010 IEEE Published by the IEEE Computer Society

network allocation vector (NAV) for all devices in the communication range. The paper also introduces two schemes that use directional DATA/ACK and omnidirectional CTS, with sending the RTS signal omnidirectionally or directionally. The relative performance of the two schemes is totally topology dependent. In [12], each node is assumed to have a switched-beam antenna using a beamforming matrix. In [13], Fahmy et al. present a scheme using omnidirectional RTS/CTS with steered beam antennas. In [10] and [14], a Directional Network Allocation Vector (DNAV) is used, which augments the NAV with a direction field. If a node receives an RTS or CTS from a certain direction, then it defers transmissions only in that direction. In [15] and [16], the angles are determined using the signal strength information along with the position information.

Another MAC protocol for multihop wireless networks using directional antennas is a TDMA-based scheme. In this approach, time is divided into frames, which are in turn, divided into slots that can be used for data transmission. Slot scheduling is done to prevent exposed and hidden terminal problems among 1-hop and 2-hop neighbors. Defining an optimal schedule is NP-complete [17], [18], [19]. A heuristic approach for a multiple access channel assignment, named UxDMA, is presented in [20] by specifying time, frequency, or code division. However, the need for a complete network topology, and the distribution of the schedule limits the scalability of this approach. In [21], Dyberg et al. analyze the performance of multihop wireless networks using TDMA MAC with beam steering and adaptive beamforming antennas. The authors use a centralized scheme, which is not well suited for this type of network. A distributed protocol is presented in [1], which uses the 2-hop neighborhood to derive a slot allocation schedule.

Some researchers have also presented protocols for directional antennas in multihop wireless networks at the network layer. In [22], Choudhury and Vaidya present the Directional Dynamic Source Routing (DDSR) protocol that operates on top of the Directional Medium Access Control (DiMAC) protocol, which is an extension of 802.11 for directional antennas. The authors use a single switched directional antenna model with two modes of operation: omnidirectional (referred to also as simply *omni* in this paper) and directional. In omni mode, after a signal is detected, the antenna determines the beam in which the received signal power is maximum. The rest of the dialog is carried out using this beam. In the directional mode, a node can select only one of its beams, and is formed with specified directional gain. To cover the complete 360 degree, considerable sweeping delays are incurred by the protocol due to sequential transmission of the RTS/CTS packets over different beams. The authors evaluate the impact of directional antennas and identify challenges in the use of this technology. Their work suggests that directional antennas have decreased impact in the dense or linear networks. Significant performance gains are realized in sparse and random topologies. Our protocol differs from DDSR [22] in that it is intended to be used for the TDMA environment as opposed to the contention-based access in 802.11. In addition, ours is a QoS routing protocol, which not only reserves a path between the source and destination, but also ensures that the discovered path meets the minimum bandwidth requirements specified by the upper layers and the application in the form of the number of TDMA slots reserved at each of the intermediate links. In [23], Saha and Johnson present improved routing techniques for WMNs using directional antennas. They used directional antennas to bridge the network partitions by transmitting selected packets over longer distances. In addition, the authors use the directional antennas to repair any route break due to node movement, and reduce delivery latency in dynamic source routing (DSR) by avoiding dropped packets and associated additional routing overhead. In [24], Jasani and Yen propose an improvement to DSR using directional antennas by preventive route maintenance and extending the life of a link using directional antennas. Preventive warnings are transmitted to the previous nodes in the path to create a directional antenna pattern when received packet power goes below a certain minimal threshold. The authors call this process of switching from omnidirectional transmission to directional transmission *orientated handoff*, and performance improvement is realized at the network layer. The protocol in this paper is also DSR based, but differs from the above protocols in the fact that it is designed for use in the TDMA environment, and for the MBAA directional antenna model.

Bao and Garcia-Luna-Aceves [1] propose a Receiver-Oriented Multiple Access (ROMA) protocol for networks using MBAA-antennas in a TDMA environment by utilizing a Neighbor-aware Contention Resolution algorithm (NCR) in [25]. Transmission and reception are performed using directional antennas. In ROMA, nodes contend for shared resources (transmission slots in this case) and contention is resolved by using the context number (slot number in this case) and node identifier. Nodes with the highest priorities among their contenders are elected to access the resource, or transmission slot, without conflict. The neighbor protocol in ROMA is used for topology maintenance, which includes 2-hop topology information for each node, and detection of the neighbors. This is accomplished by employing short signals that use the omnidirectional mode of the antenna. ROMA is a distributed algorithm that allows the nodes to calculate their channel access schedules based on their 2-hop topology information. The protocol evenly splits nodes in the network into transmitters and receivers, which are paired together to establish a useful communication. In addition, ROMA operates at the link layer level of the networking stack and is mainly concerned with providing a collision-free medium access mechanism between the neighboring nodes. The protocol presented in this paper operates at a higher level than ROMA. In the proposed protocol, a multihop route between source and destination nodes is discovered and reserved using a source routing mechanism. After route discovery, the nodes can use the reserved path to transmit their data using the reserved slots at each of the intermediate nodes.

In [26], Jawhar and Wu present a race-free routing protocol for QoS support in TDMA-based multihop wireless networks. The protocol allows a source node to find and reserve a QoS path with a certain required bandwidth (which is translated into number of data slots) to a desired

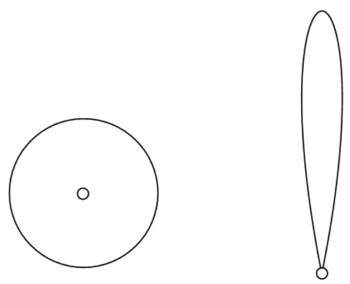

Fig. 1. (a) Transmission pattern of an omnidirectional antenna. (b) Transmission pattern of a directional antenna.

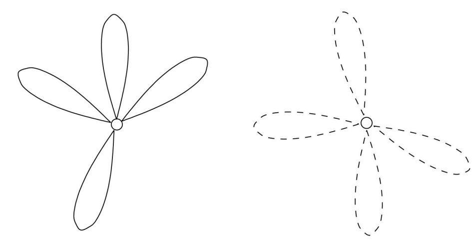

Fig. 2. Transmission pattern of an MBAA antenna system with $k = 4$ beams. Each of the k beams can be oriented in a different desired direction. (a) Beams in transmission mode. (b) Beams in reception mode.

destination node. In this paper, the protocol is extended to do path reservation in TDMA-based multihop wireless networks, where the nodes are equipped with directional MBAA-antennas. Consequently, the protocol is for the directional TDMA environment. It is different from the above protocols in that it is both on-demand, based on the DSR protocol strategy, and is distributed. Each node only needs to keep track of the slot allocation status of its 1-hop and 2-hop neighbors, as opposed to having information about the entire topology and slot allocation states of all of the nodes in the network. This characteristic enhances the efficiency and the scalability of the proposed protocol. In addition, this QoS routing protocol ensures that each link along the discovered path meets the minimum bandwidth requirement specified by the application layer. This constraint is needed by many multimedia and real-time applications that impose certain QoS requirements on the network in order to operate properly.

The remainder of the paper is organized as follows: Section 2 presents the assumptions and definitions used in the protocol. Section 3 provides the protocol, along with the data structures, algorithms, and some examples. Section 4 includes simulation results that demonstrate the effectiveness of the protocol. Section 5 concludes the paper.

2 Directional Antenna System Assumptions and Definitions

In this paper, it is assumed that each node is equipped with an MBAA system with each antenna capable of transmitting or receiving using any one of k beams that can be directed toward the desirable node [1], [27], [28]. In order for node x to transmit to a node y, node x directs one of its k antennas to transmit in the direction of node y, and node y, in turn directs one of its k antennas to receive from the direction of node x.

Radio signals transmitted by omnidirectional antennas propagate equally in all the directions. On the other hand, directional antennas possess multiple antenna elements so that individual omnidirectional RF radiations interfere with each other in a constructive or destructive manner. This causes the signal strength to increase in one direction, and is known as the main lobe. The increase of the signal strength in a desired direction and the lack of it in other directions is modeled as a lobe. The angle of the directions, with respect to the center of the antenna pattern, where the radiated power drops to one-half the maximum value is defined as the antenna beamwidth, denoted by β [1]. With the advancement of silicon and DSP technologies, several antenna patters can be formed in different desired directions (for transmission or reception) simultaneously. Fig. 1a shows the transmission patterns of an omnidirectional antenna, while Fig. 1b gives the transmission pattern of a directional antenna.

In the multihop MWN environment considered in this paper, each node is equipped with an MBAA antenna that is capable of receiving and transmitting one or more packets simultaneously, by pointing the antenna beams toward desired nodes, while annulling in all other directions. The antenna system can either transmit or receive data, but cannot do both simultaneously.

It is also assumed that an MBAA antenna is capable of broadcast by adjusting the beam width or by using the omnidirectional mode. This capability is useful for communicating control information, as well as finding the neighborhood. Preston et al. [29] presented operation modes for the directional antennas for finding the coarse, as well as the precise angular location of a single or multiple radiating sources in 100 or 200 microseconds. Fig. 2 shows a node equipped with an MBAA antenna array with $k = 4$ beams. Fig. 2a shows the antenna array in the transmission mode, and Fig. 2b shows the antenna array in the reception mode.

This paper uses the neighbor protocol of [1] to acquire and maintain the two-hop neighbor information needed by the scheduling mechanism. Exchanging neighborhood information is not done using the directional antenna, as this requires a priori knowledge of the neighborhood. Therefore, neighborhood information is sent over a common channel on a best effort basis using the omnidirectional mode in a separate time slot. The collected control information is used to propagate route reservation control messages during the route discovery process [1].

Two nodes, x and y, are considered 1-hop neighbors if they are within each other's directional range. In order for a node x to successfully transmit data to one of its 1-hop neighbor nodes y, x must orient one of its transmitting beams in the direction of y, and y must orient one of its reception beams in the direction of x. Fig. 3 shows a group of nodes communicating using MBAA directional antennas, with node d transmitting to both b and e simultaneously, using two different directional antenna beams. Also, node b is receiving from a and d simultaneously. Node g is transmitting to f. Note that even though node g's transmission to f

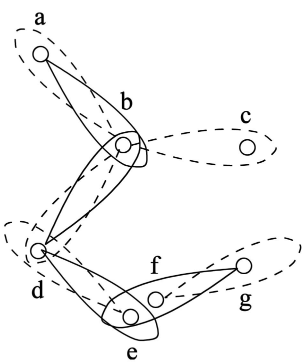

Fig. 3. An example showing node communication using directional antennas.

covers e, e does not have one of its receiving beams oriented toward g, and subsequently, is deaf [30] to that transmission.

Each node x maintains information about the angular location (direction) of each of its 1-hop and 2-hop neighbors [1]. For simplicity, the nodes are assumed to be placed on a flat plane. As illustrated in Fig. 4, the horizon of each node is divided into $360^\circ/(\beta/2) = 720^\circ/\beta$ segments, and every two continuous segments define one group. A segment corresponds to the minimum angular separation of two neighbors in order to receive two separate antenna beams without interference. Therefore, $720^\circ/\beta$ groups exist. Each 1-hop neighbor y of x belongs to two groups that overlap with y. A_x^y denotes the set of angular groups belonging to 1-hop neighbor y of x. Two nodes y and z are considered in the same angular direction with respect to a third node x if and only if $A_x^y \cap A_x^z \neq \phi$. As an example, consider the nodes in Fig. 4, where the horizon with respect to a node a is shown. According to the earlier definition, the set of angular groups for links (a,b), (a,c), (a,d), and (a,e) are $A_a^b = \{13, 14\}$, $A_a^c = \{14, 15\}$, $A_a^d = \{15, 16\}$, and $A_a^e = \{1, 2\}$. Therefore, nodes b and c are considered in the same angular direction with respect to node a because $A_a^b \cap A_a^c = \{14\} \neq \phi$. Similarly, nodes c and d are considered in the same angular direction. However, nodes b and d are not in the same angular direction since $A_a^b \cap A_a^d = \phi$.

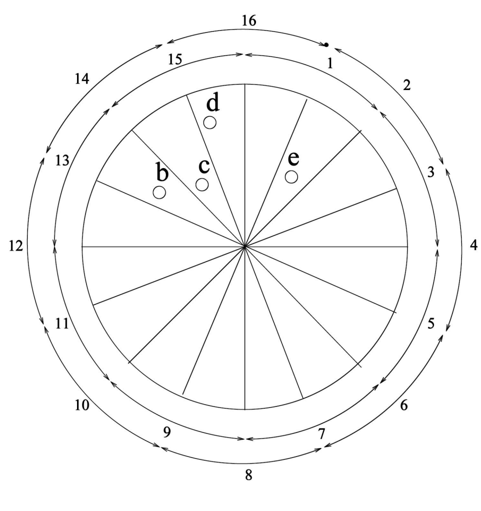

Fig. 4. The horizon as seen by a node. The figure includes 16 segments and 16 angular groups.

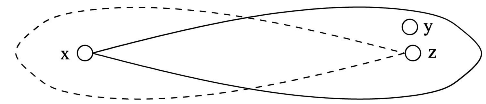

Fig. 5. Illustration of allocation rule 2.

3 Our Proposed Protocol

The TDMA is the networking environment assumed in this paper, where a single channel is used to communicate between nodes. The TDMA frame is composed of a control phase and a data phase [31], [32]. Each node has a designated control time slot. However, different nodes must compete for using the data time slots.

Liao et al. [33] show the complexity of transmitting and receiving in a TDMA single channel omnidirectional antenna environment. In this section, we present slot allocation rules for the TDMA directional antenna environment. The model used in this protocol is similar to that used in [26] and [33], but includes modifications to support directional antenna systems. Each node keeps track of the slot status information of its 1-hop and 2-hop neighbors. This is necessary in order to allocate slots in a way that does not violate the slot allocation conditions imposed by the nature of the wireless medium, and to take the hidden and exposed terminal problems into consideration. Below are the slot allocation conditions:

3.1 Slot Allocation Conditions for Directional Antennas

A time slot t is considered free to be allocated to send data from a node x to a node y if the following conditions are satisfied:

1. Slot t is not scheduled to receive in node x or scheduled to send in node y, by any of the antennas of either node (i.e., antennas of x must not be scheduled to receive and antennas of y must not be scheduled to transmit, in slot t).
2. Slot t is not scheduled for receiving in any node z, that is a 1-hop neighbor of x, from node x where y and z are not in the same angular direction with respect to x (i.e., $A_x^y \cap A_x^z \neq \phi$).
3. Slot t is not scheduled for receiving in node y from any node z, that is a 1-hop neighbor of x, where x and z are in the same angular direction with respect to y (i.e., $A_y^x \cap A_y^z \neq \phi$).
4. Slot t is not scheduled for communication (receiving or transmitting) between two nodes z and w that are 1-hop neighbors of x, if w and y are in the same angular direction with respect to z (i.e., $A_z^w \cap A_z^y \neq \phi$), and x and z are in the same angular direction with respect to w (i.e., $A_w^x \cap A_w^z \neq \phi$).

Fig. 5 illustrates allocation rule 2. Node x cannot transmit to node y using slot t because it is already using slot t to transmit to node z, which is in the same angular direction as node y. Fig. 6 illustrates allocation rule 3. Node x cannot allocate slot t for sending to node y because slot t is already scheduled for sending from node z. Node z is a 1-hop neighbor of x, and $A_y^x \cap A_y^z \neq \phi$. Fig. 7 illustrates allocation rule 4. Slot t cannot be allocated to send from x to y because

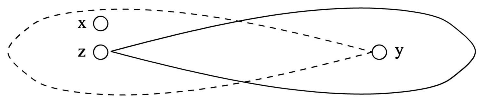

Fig. 6. Illustration of allocation rule 3.

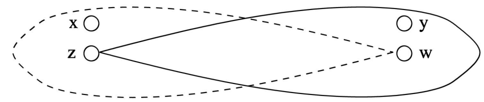

Fig. 7. Illustration of allocation rule 4.

it is already scheduled to communicate between two nodes z and w, that are 1-hop neighbors of x, where $A_z^x \cap A_z^w \neq \phi$ and $A_x^y \cap A_x^z \neq \phi$.

3.2 The Data Structures

The proposed protocol is on-demand, source based, and is similar to DSR [34]. On-demand nature increases its efficiency since traffic overhead control is needed only when data communication is desired between nodes.

Each node maintains and updates three tables; ST, RT, and H. Considering a network with n nodes, and s data slots in the TDMA frame, in a node x, the tables are denoted by ST_x, RT_x, and H_x. The tables contain the following information:

- $ST_x[1..n, 1..s]$: This is the send table, which contains slot status information for the 1-hop and 2-hop neighbors. For a neighbor i and slot j, $ST_x[i, j]$ is a structure that has two fields: 1) *The state field*: It can have one of the following values representing three different states: 0—free, 1—allocated to send, 2—reserved to send. 2) *The angular groups field*: It contains k sets of angular groups (one for each antenna). The entry $A[a]_i^j$ denotes the set of angular groups to which the ath sending antenna is pointed. $A[a]_i^j = \phi$ is used to indicate that the ath antenna for neighbor i is not used during slot j.
- $RT_x[1..n, 1..s]$: This is the receive table, which contains slot status information for the 1-hop and 2-hop neighbors. For a neighbor i and slot j, $RT_x[i, j]$ is a structure that has two fields: 1) *The state field*: It can have one of the following values representing three different states: 0—free, 1—allocated to receive, 2—reserved to receive. 2) *The angular groups field*: It contains k sets of angular groups. The entry $A[a]_i^j$ denotes the set of angular groups to which the ath receiving antenna is pointed. Also, $A[a]_i^j = \phi$ is used to indicate that the ath antenna for neighbor i is not used during slot j.
- $H_x[1..n, 1..n]$: This table contains information about node x's 1-hop and 2-hop neighborhood. Each entry $Hx[i, j]$ is a structure, which has two fields: 1) *The neighbor field*: It contains a 1 if node i, which is a 1-hop neighbor of node x, has node j as a neighbor, and contains a 0 otherwise. 2) *The angular group field*: A_i^j, which contains the set of angular groups to which node j belongs.

It is important to note that the word "slot" implies a "slot in a particular direction using an associated antenna." For example, each of the slot timers defined later in this paper are associated with a particular slot/antenna pair.

3.3 The QoS Path Reservation Algorithm

When a node S wants to send data to a node D, with a bandwidth requirement of b slots, it initiates the QoS path discovery process. Node S determines if enough slots are available to send from itself to at least one of its 1-hop neighbors. If so, it broadcasts a $QREQ(S, D, id, b, x, PATH, NH)$ to all of its neighbors. The message contains the following fields:

1. S: ID of the source node.
2. D: ID of the destination node.
3. id: Message ID. The (S, D, id) triple is, therefore, unique for every QREQ message and is used to prevent looping.
4. b: Number of slots required in the QoS path from S to D.
5. x: The node ID of the host that is forwarding this QREQ message.
6. $PATH$: A list of the form $((h_1, l_1), (h_2, l_2), \ldots, (h_k, l_k))$. It contains the accumulated list of hosts and time slots, which have been allocated by this QREQ message so far. h_i is the ith host in the path, and l_i is the list of slots used by h_i to send to h_{i+1}. Each of the elements of l_i contains the slot number that would be used, along with the corresponding set of angular groups A_i^{i+1}, which represents the direction in which the sending antenna of host i must be pointed to send data to host $i+1$ during that slot.
7. NH: A list of the form $((h'_1, l'_1), (h'_2, l'_2), \ldots, (h'_k, l'_k))$. It contains the next-hop information. If node x is forwarding this QREQ message, then NH contains a list of the next-hop host candidates. The couple ($h_i^{'}$, l'_i) is the ID of the host, that can be a next-hop in the path, along with a list of the slots which can be used to send data from x to h'_i. l'_i is a list of the slots to be used to send from host i to host $i+1$ along with the angular group for each slot. $l_i^{'}$ has the same format as l_i in PATH.
8. $Max\_QREQ\_node\_wait\_time$, $Max\_QREQ\_tot\_wait\_time$, and $Max\_QREQ\_QREP\_tot\_wait\_time$: QREQ message wait timing constraints that are specified by the application. Each timer is discussed later in more detail.

When an intermediate node receives the QREQ message, it composes the NH list, which includes the neighbors with which it has a link that contains at least b free slots in the corresponding direction. The message is then forwarded to these neighbors. If a QREQ message reaches the destination node D, it means that a QoS path from S to D has been discovered, and that there are at least b free slots available to send data from each node to each subsequent node along this path. These slots are now marked as *allocated* in the ST and RT tables of the corresponding nodes. Subsequently, node D unicasts a reply message, $QREP(S, D, id, b, PATH, NH)$, to node S. This message is propagated along the nodes indicated in $PATH$. As the QREP message travels back to the source node, all of the intermediate nodes along

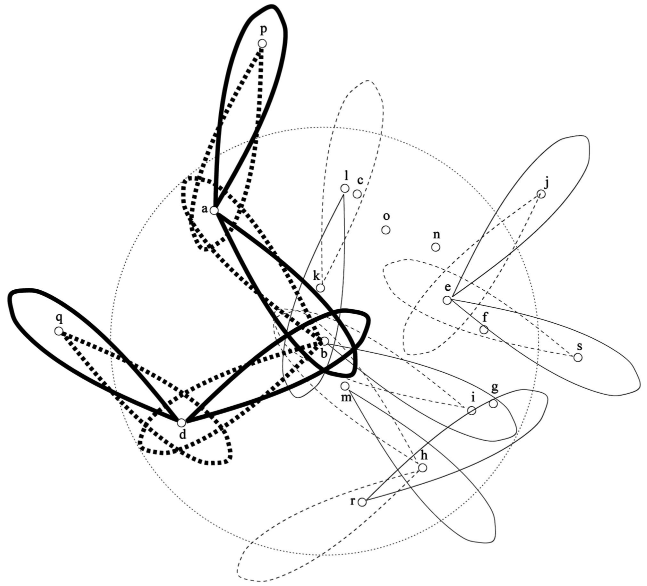

Fig. 8. A detailed example showing the allocation of slots 1 and 2 at node b. Bold cones show transmissions/receptions in slot 1, and plain cones show transmissions/receptions in slot 2. The circle indicates the directional range of node b.

the allocated path must confirm the reservation of the corresponding allocated slots (i.e., change their status from *allocated* to *reserved*). The timing and propagation of the QREQ and QREP messages are controlled by timers, a queuing process, and synchronous and asynchronous slot status broadcasts, which are discussed later in details.

3.4 A Detailed Example of Slot Allocation at an Intermediate Node

Fig. 8, and the corresponding entries in Fig. 9, provide an example of the slot allocation considerations at an intermediate node b. In the example, node b receives a QREQ message from node a, and is determining to see which of its 1-hop neighbors it can extend the QREQ message. The example illustrates considerations for slots 1 and 2 of the TDMA frame. The portion of the allocation for slots 1 and 2

Node	a		b		c		d		e	
Slot	S1	S2	S1	S2	S1	S2	S1	S2	S1	S2
S/R	S		R	S			S			S
A1	{5, 6}		{14, 15}	{5, 6}			{2, 3}			{1, 2}
A2	{1, 16}		{10, 11}				{13, 14}			{4, 5}

Node	f		g		h		i		j	
Slot	S1	S2	S1	S2	S1	S2	S1	S2	S1	S2
S/R						R		R		R
A1						{13, 14}		{12, 13}		{9, 10}
A2						{10, 11}				

Node	k		l		m		n		o	
Slot	S1	S2	S1	S2	S1	S2	S1	S2	S1	S2
S/R		R		S		S				
A1		{1, 16}		{8, 9}		{5, 6}				
A2										

Node	p		q		r		s			
Slot	S1	S2	S1	S2	S1	S2	S1	S2		
S/R	R		R			S		R		
A1	{8, 9}		{5, 6}			{2, 3}		{12, 13}		
A2										

Fig. 9. The allocation table, which corresponds to the detailed example showing the allocation of slots 1 and 2 at node b.

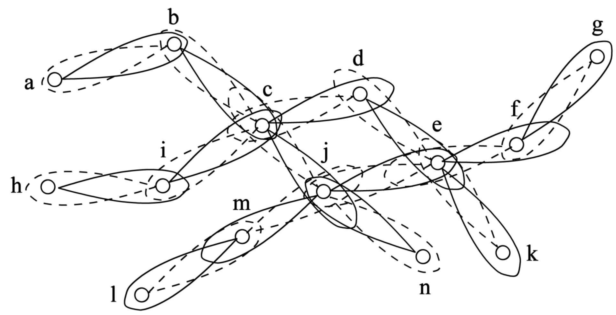

Fig. 10. An example showing three QoS paths: $abcdefg$, $hicjek$, and $njml$.

at node b is shown in Fig. 9. In this example, slot 1 is reserved to send from node a to node b and node p, simultaneously, for different QoS paths. Slot 1 is also reserved to send from node d to node q and node b. These reservations of the same slot to send from the same node to multiple nodes, for different QoS paths, is not possible in an omnidirectional antenna system. This demonstrates the significant spatial reuse that can be achieved in the directional antenna environment. According to rule 1 of the slot allocation, slot 1 cannot be allocated by node b to send to any of its neighbors on any of its antennas because this slot is already scheduled to receive by node b (from nodes a and d).

Let's consider the possibility of allocating slot 2 to send from node b to each of its neighbors. According to rule 1, slot 2 cannot be allocated to send from b to i because it is already scheduled to send for another QoS path. Also, according to the same rule, slot 2 cannot be scheduled to send from node b to e because slot 2 is already scheduled to send by node e. It is scheduled to send from node e to nodes j and s for different QoS paths, which is another illustration of spatial reuse that is feasible due to a directional antenna system.

According to rule 2 of the slot allocation conditions, node b cannot allocate slot 2 to send to node g because slot 2 is already scheduled to send from node b to node i, where $A_b^i \cap A_b^g \neq \phi$ (i.e., node i is in the same angular direction as node g with respect to node b). Also, according to rule 3, slot 2 cannot be allocated by node b to send to node h. This is because this slot is already scheduled to receive in node h from node m, where $A_h^m \cap A_h^b \neq \phi$. Note that slot 2 is also scheduled by node h to receive from r on a different antenna without preventing the use of the same slot to send from node m to node h.

According to rule 4, slot 2 cannot be scheduled to send from node b to c because it is already scheduled to send from l to k, where ($A_k^l \cap A_k^c \neq \phi$) and ($A_l^b \cap A_l^k \neq \phi$). According to the same rule, and because of the same reason, slot 2 cannot be scheduled to send from b to k or from b to l.

As a result, node b is able to allocate slot 2 to send to nodes f, n, and o.

3.5 An Example of Multiple QoS Paths Passing through Common Nodes

Fig. 10 shows an example with three different reserved QoS paths: $abcdefg$ (path 1), $hicjek$ (path 2), and $njml$ (path 3). The three paths share several common nodes. Namely, nodes c and e are common between paths 1 and 2, and node j is common between paths 2 and 3. Due to the use of directional antennas, these common nodes are able to receive different data belonging to different paths from multiple directions in the same data slot without interference. Similarly, they can also transmit to different nodes belonging to different paths in different directions in the same data slot. This scenario is not possible in the omnidirectional antenna environment and illustrates the potential increase of network throughput in MWNs using directional antennas.

3.6 Wait Timers

We define the following timers, which control the allowable delay of the propagation of the QREQ and QREP messages through the system. These timers can be initialized to a tunable value, which can vary according to the requirements of the application being used. It is also possible to disable some of these timers if the application does not have such delay constraints.

TTL_allocated_slot_time. Each slot t in ST and RT tables has a Time to Live (TTL_t) count down timer associated with it. This TTL_t timer is only needed when the slot is set free to be allocated. As soon as a slot is converted from free to allocated, its TTL timer is set to a certain "time to live" parameter. This is a tunable parameter, which can be determined according to the application needs. The TTL_t timer is set to 0 upon initialization and when the slot becomes free. When the status of a slot t is changed from free to allocated due to a QREQ, the TTL_t timer is initialized to a predetermined $TTL\_allocated\_slot\_time$, which should be at least equal to the Round Trip Time (RTT) for a QREQ to come back as a QREP. This time is a tunable parameter, which can be fixed according to the application requirements and/or the network size and/or density. It can be increased with a larger number of nodes in the network. A reasonable value could be $2 * RTT$, while it could be set to a smaller or larger value depending on the size and propagation delay characteristics of the network involved.

A large value for this TTL_t timer corresponds to a conservative strategy. If it is too large, a slot would have to wait too long to automatically convert back to free. That lengthens the path acquisition time for a QREQ, which might not be desirable in certain applications. On the other hand, if the TTL time is too small, then a node will be too anxious to return allocated slots to the "free" status before the reservation is confirmed with a QREP message. This creates a risk of converting a slot back to free status too soon. After a short amount of time, the corresponding QREP message of the QREQ message that initially allocated this slot comes back. However, the slot, which is changed to free, can now be allocated for another path. This way, double allocation of the same slot exists for two different paths, which could lead to a racing condition that the protocol strives to avoid.

Explicit Deallocation Message from the Destination. In addition to the above soft allocation timer strategy, further performance improvements can be achieved by having an explicit deallocation message flooding from the destination to the source. This message is initiated by the destination when it receives as soon as a QoS path is discovered. The reception of the deallocation message by the nodes in the network will cause an immediate deallocation of the slots, which were not used in the final path(s). This increases the utilization and efficiency of the network. We incorporated both the soft deallocation timer, as well as the explicit deallocation message in the protocol.

TTL_reserved_slot_time. When a slot is reserved (i.e., its allocation is confirmed at a $reserved$ status) for a particular QoS path, it must be used for actual data transmission within a certain time-out period, which we define as the $TTL\_reserved\_slot\_time$. This time is a parameter, which can be set according to the application and the network environment involved. If a slot is not used for data transmission at any time, it must be returned to a free status, which is achieved in the following manner: The associated timer is refreshed each time the slot is used for data transmission, and is counted down constantly. If this timer reaches zero at any time, then the slot is returned back to $free$ status. This timing is also useful for a situation where the QREP message used to confirm a slot reservation, is successful in its propagation from the destination to the source. In this case, the nodes, which already confirmed the reservation, will still be able to return these slots back to the "free" status after this time-out period.

Max_QREQ_node_wait_time. The QREQ can wait at an intermediate node for a maximum amount of time $Max\_QREQ\_node\_wait\_time$. This is a parameter that is tuned to the application and the network requirements. A reasonable value can be equal to $2 * RTT$. Its effect is similar to what was described earlier in the $TTL\_allocated\_slot\_time$ section. This time affects the QoS path acquisition latency and may be limited by the application.

Max_QREQ_tot_wait_time. Another related delay type is the QREQ total wait time, which is the maximum allowable cumulative wait delay for the QREQ as it propagates through the network. This delay is controlled by the timer $max\_QREQ\_tot\_wait\_time$, is decremented at each node as per the time the QREQ had to wait at that node, and it is forwarded along with the QREQ to the next node.

Max_QREQ_QREP_tot_wait_time. Another timer can be defined as $Max\_QREQ\_QREP\_tot\_wait\_time$, and is the total time for path acquisition ($QREQ\ propagation + QREP\ propagation$). This time is also decremented by each node accordingly and forwarded along with the corresponding QREQ and QREP messages. Whenever a node is forwarding a QREQ or a QREP message, it checks this time. If it is zero, then this means the QoS path reservation process has taken longer than the maximum allowable time and the corresponding QREQ or QREP message should now be dropped. Furthermore, the protocol can also take one of the following actions: 1) Send a notification message to all of the nodes along the reserved path (the nodes which forwarded the QREP message from the destination to this node) to return the corresponding slots, which have been allocated and/or reserved by this path to free status, or 2) let those already-reserved-slots time out to free status as described by the $TTL\_reserved\_slot\_time$, defined earlier.

The $Max\_QREQ\_node\_wait\_time$, $Max\_QREQ\_tot\_wait\_time$, and $MAX\_QREQ\_QREP\_tot\_wait\_time$ timers are optional and can be set to different values according to their importance and/or criticality in the application that is being used. Similar timing techniques can be employed for the transmission of data packets, as well for different applications, such as multimedia, voice, and video. This is due to the fact that the packet can hold voice or video frames that must be delivered within a certain amount of time beyond which they are useless and must be simply discarded.

3.7 Status Broadcasting and Updating

There are two types of node status broadcasts: synchronous (periodic) and asynchronous.

Synchronous Periodic Status Updates. Each node broadcasts its slot allocation status (the ST and RT table information updates) to its 1-hop and 2-hop neighbors (i.e., with a 2-hop TTL). This broadcast is done periodically (synchronously) according to a predetermined periodic slot status update frequency. We define this as $periodic\_status\_update\_time$. These periodic updates enable the nodes to maintain updated neighborhood information as nodes come within or go out of their range. Furthermore, these updates inform the node of its neighbor's slot status information on a periodic basis.

When a node does not receive any synchronous (periodic) or asynchronous (due to changes in slot status) updates from a neighbor after a time-out period, which we call Status_update_tot, it will assume that this node is no longer one of its 1-hop or 2-hop neighbors, and will delete that neighbor from its ST and RT tables.

Asynchronous Status Updates. The status update is done asynchronously as the status of slots is changed from free to allocated, or from allocated to reserved. There is no need to inform the neighbors of the change from allocated to free state due to TTL timer expiration. The neighbors will count down the time of the allocated slots as well, and will change them to free status (i.e., will assume that the corresponding neighbor node will have done that) if no reservation change is indicated from the corresponding neighbor node. Note that the status updates are done with a 2-hop TTL flood to the 1-hop and 2-hop neighbors.

The asynchronous updates of receive and send slot status with the three state information, which includes the *allocated* status, solves the parallel reservation problem stated earlier in the paper, and eliminates the associated race condition, which has not been done in any previous research. When a 1-hop neighbor receives a separate and different QREQ, it becomes aware of the $free/allocated/reserved$ status of its neighbors' slots, rather than just their $free/reserved$ status. This way, it will consider only slots which are totally free and will prevent the related race condition from occurring. This consideration is addressed in the $select\_slot()$ function, which is described later in this paper.

3.8 The Main Algorithm at an Intermediate Node

The protocol uses three states per slot to avoid any race conditions when multiple routes passing through common nodes are being reserved. The possible race conditions and their remedies, which are similar for omnidirectional and directional antenna environments are described in detail in [26].

When a node y receives a broadcasting message $QREQ(S, D, id, b, x, PATH, NH)$ initiated by a neighboring host x, it checks to determine whether it has received this request (uniquely identified by (S, D, id)) previously. If not, y performs the following steps: If y is not a host listed in NH, then it exits this procedure. Otherwise, it calculates the values of the variables $NUyz$, $ANUyz$, and Fyz, which are defined as follows:

- $NUyz$: The number of slots that are not usable for sending from y to z. This means that at least one confirmed reservation exists at y or one of its neighbors, which does not allow slot t to be used from y to send to z. This is due to any violation of any of the three slot allocation conditions.
- $ANUyz$: The number of slots that are allocated not usable for sending data from y to z. A slot is called allocated not usable (ANU) if there exist totally allocated reservations at y or its neighbors, which do not allow slot t to be used from y to send to z. This could be due to any violation of any of the three slot allocation conditions. However, violations of any of the lemma conditions are only (and totally) due to pure allocations (not confirmed reservations) at y and/or its neighbors.
- Fyz: The number of slots that are free at a node y to send to a node z, respectively. This means that this slot is currently completely available to be used for sending from node y to node z, and therefore satisfies all three of the slot selection conditions.

Therefore, at node y, it is necessary to determine a separate set of $NUyz$, $ANUyz$, and Fyz for each neighbor z of y. When a node y receives a QREQ message from a node x, it uses algorithm 1, which is shown below to forward the message, or to insert it in the $QREQ\_pending\_queue$, or to drop it.

Algorithm 1 works in the following manner: When a QREQ message arrives at a node y from a node x, it does the following: The algorithm first updates the ST and RT tables with the information in PATH. Then, it uses three routines to calculate $NUyz$, $ANUyz$, and Fyz from ST and RT tables while taking all of the slot allocation rules into account.

Algorithm 1. The main algorithm at an intermediate node
When a node y receives a QREQ message
  Update the ST and RT tables with the information in PATH
  $NH\_temp = \phi$
  for each 1-hop neighbor node z of y **do**
    $NUyz = calcR(z, ST, RT)$
    $ANUyz = calcA(z, ST, RT)$
    $Fyz = calcF(z, ST, RT)$
    if $Fyz \geq b$ **then**
      $L = select\_slot(y, z, b, ST, RT)$
      if $L \neq empty$ **then**
        $NH\_temp = NH\_temp(z, L) \mid (z, L)$
      else
        Error: cannot have $Fyz \geq b$ and $L = empty$
      end if
    end if
  end for
  if $NH\_temp \neq \phi$ **then**

Let (h_i', l_i') be the entry in NH such that $h_i' = $ y
let $PATH\_temp = PATH \mid (x, l_i')$
broadcast $QREQ(S, D, id, b, x, PATH\_temp,$
$NH\_temp)$ message
else
for each 1-hop neighbor node z of y **do**
if $(Fyz + ANUyz) \geq b$ **then**
let $t_{mas} =$ *maximum time left for required allocated slots to become free (or reserved)*
if $max\_QREQ\_tot\_wait\_time \geq t_{mas}$ **then**
insert QREQ message in $QREQ\_pending\_queue$
end if
end if
end for
end if

Then, the algorithm initializes the next-hop list NH_temp to empty, and then attempts to build it by adding to this list each 1-hop neighbor z of y, which has b slots free to send from y to z. The algorithm uses the select_slot function, which allocates slots and the information in the updated ST and RT tables. There are three possible conditions that can take place.

If at least one neighbor z of y has b slots free to send from y to z, then this is called *condition 1*. The NH_temp list will no longer remain empty and the node y will broadcast (i.e., forward) the QREQ message after incorporating the node x and the list li' (i.e., the list of slots used to send from x to y) PATH (using $PATH\_temp = PATH \mid (x, li')$). Here, | means concatenation.

Otherwise, if the NH_temp list is empty after checking all of the neighbors, then that means that there are no neighbors z of y with b free slots according to the slot selection conditions. At this point, the algorithm tries to determine if there is any "hope," i.e., if there is at least one 1-hop neighbor z of y, which satisfies the condition $(Fyz + ANUyz) \geq b$. This would be *condition 2* as the algorithm checks if the maximum time left for the required allocated slots to become free (or reserved) does not exceed the maximum total wait time left for this QREQ message ($Max\_QREQ\_tot\_wait\_time$). Consequently, this QREQ message is placed in the $QREQ\_pending\_queue$, and will be scanned each time a slot becomes free. This process will be discussed in more detail later in this paper. If, on the other hand, no 1-hop neighbor z of y has a condition of $(Fyz + ANUyz) \geq b$, then there is "no hope" at the current time and the QREQ message is dropped.

3.9 The *select_slot* Function

The $select\_slot(y, z, b, ST, RT)$ function returns a list of slots that are available to send from node y to z. As described earlier, it does this according to the slot allocation rules and the slot status information from the updated ST and RT tables. $select\_slot()$ returns an empty list if b slots are not available rom node y to z.

3.10 The *QREQ_pending_queue*

The QREQs that are waiting for slots to become free are placed in a $QREQ\_pending\_queue$. While waiting for the status of different slots in the table, some slots will be freed, and others will be confirmed. Every time a change in slot status occurs (due to timer expiration, or confirming a reservation), the queue is scanned.

Scanning the $QREQ\_pending\_queue$. Every time the queue is scanned, all QREQ messages having expired wait timers are deleted from the queue. These timers are: $Max\_QREQ\_node\_wait\_time$, $Max\_QREQ\_QREP\_tot\_wait\_time$, and $Max\_QREQ\_tot\_wait\_time$. Also, for each QREQ in the queue, the new values for Fyz, $ANUyz$, and $NUyz$ are calculated, and it is determined under which conditions the new QREQ status falls. There are three possibilities:

- Changed to condition 1 (i.e., now $Fyz \geq b$): In this case, forward the pending QREQ and delete the QREQ from the $QREQ\_pending\_queue$.
- Changed to condition 2 (i.e., now $(Fyz + ANUyz) \geq b$): In this case, leave the corresponding QREQ in the $QREQ\_pending\_queue$.
- Changed to condition 3 (i.e., $(Fyz + ANUyz) < b$): In this case, delete the corresponding QREQ from the $QREQ\_pending\_queue$ (i.e., drop this QREQ message). Here, another policy can be adopted, which would be to send a reject message back to the source of the QREQ to inform it of the rejection if the protocol requires informing the source nodes of the failing QREQ.

If the TTL for an allocated slot expires, this means that the slot has been allocated for *too long* and not confirmed (i.e., reserved) by a QREP message. In this case, the corresponding slot status in ST and RT tables is set to $free$.

If the status of a QREQ message in the queue changes into condition 1, then the algorithm calls the $select\_slot()$ function for all nodes that are 1-hop neighbors of y. It then builds the next-hop list, which includes every neighbor node z for which there are b slots available to send from y to z, and the list of these slots. This is done using Algorithm 2.

Algorithm 2. Forwarding the QREQ message from the $QREQ\_pending\_queue$
$NH\_temp = \phi$
for every 1-hop neighbor z of y **do**
$L = select\_slot(y, z, b, ST, RT)$
if $L \neq \phi$ **then**
$NH\_temp = NH\_temp \mid (z, L)$
end if
end for
if $NH\_temp \neq \phi$ **then**
let (h_i', l_i') be the entry in NH such that $h_i' = y$
let $PATH\_temp = PATH \mid (x, l_i')$
broadcast $QREQ(S, D, id, b, y, PATH_temp,$
$NH\_temp)$
delete QREQ message from the $QREQ\_pending\_queue$
end if

The QREQ message is forwarded by the intermediate nodes that are able to allocate b slots to send data, and can therefore be a part of the QoS path that is being discovered and reserved.

As the QREQ message propagates from the source to the destination, the slot reservation information is not updated in the ST and RT tables. This unconfirmed reservation information is only maintained and updated in the QREQ message as it propagates through the nodes. The status of the corresponding slots in the ST and RT tables in the nodes continues to be $free$. This can lead to multiple reservations of the same slots by different QREQ messages due to a race condition, which is explained later in this paper. If and when the QREQ message arrives at the destination node D, then a QoS path from S to D, with b slots in each hop, has been discovered. In this case, the destination D replies by unicasting a $QREP(S, D, id, b, PATH, NH)$ back to the source, which confirms the path that is allocated by the corresponding QREQ message. The QREP message propagates from D to S through all of the intermediate nodes that are specified in PATH, which contains a list of the nodes along the discovered path and allocated slots. As the QREP message propagates through the intermediate nodes, each node updates its ST and RT tables and changes the status of the corresponding slots to $reserved$. This represents the confirmation of the reservation of the slots for the discovered path.

4 Performance Analysis

In order to verify and analyze the performance of our protocol, simulation experiments have been conducted.

4.1 Simulation

Basically, the simulator starts by generating an area with certain dimensions, then randomly places a predetermined number of nodes in the area. The nodes have a given transmission range. From the placement of the nodes and their ranges, a graph is generated. Then, the simulator generates a number of data messages with a certain length for each message (different distributions can be used). Each message has a random source and destination pair. In the simulation, different message interarrival time distributions can be considered depending on the application and environment that is assumed. The randomly generated length of each message is large enough so that each message consists of tens or hundreds of packets, and simulates an application generated event that requires the transmission of multiple packets between a particular source/destination pair. In our case, we consider the arrival times of the messages to be according to an exponential distribution (i.e., Poisson arrival process), with a certain mean interarrival time. When the data message is processed by the source, it generates a QREQ message to discover a QoS path to the corresponding destination. The QREQ message is propagated through the nodes as per our algorithm. Each node has a routing table, as well as all of the tables needed for the algorithm (ST, RT, H, routing table, all of the required slot data structures, etc.). When the source receives the QREP message, it starts data transmission. The simulations are done for three different cases: 1) one antenna representing the omnidirectional antenna case ($dir = 1,\ \text{angle of coverage} = 360^\circ$), 2) two antennas ($dir = 2,\ \text{angle of coverage} = 180^\circ$ per antenna), 3) four directional antennas ($dir = 4,\ \text{angle of coverage} = 90^\circ$ per antenna).

4.2 Simulation Parameters

A set of simulation experiments have been performed. Table 1 shows a sample of the simulation parameters used in the experiments. The results for two sets of experiments are shown. Figs. 11 and 12 contain the results for the first set of experiments, and Figs. 13 and 14 contain the results for the second set of experiments. The number of nodes (n) is 30 in an area of $300 \times 300\ \text{m}^2$. The total number of data slots in the frame (dsn) is 30. The number of slots required for each session is a random number with a uniform distribution in the range from 1 to 4 slots (1 to max_b). The range of each node is 115 m. The session (or data message) arrival is a Poisson process with a mean, which is varied from 1 to 10 messages/sec. The message length is randomly selected according to a uniform distribution, with a range from 0 to 10 Mbytes for the first set of experiments, and from 0 to 100 Mbytes for the second set of experiments.

TABLE 1
Parameters for the Directional Antenna Protocol Simulation

Parameter	Value
Network Area	$300 \times 300\ m^2$
Number of Nodes	30
Transmission Range	115 m
Bandwidth	2 Mb/s
Data Packet Size	512 bytes
Number of Data Slots	30
Number of Sessions	20
Average Message Length	100 MB
MAX_SLOT_RES_TIME	10980 ms
MAX_SLOT_ALLOC_TIME	1350 ms
MAX_B	4 slots

4.3 Simulation Results and Analysis

Several performance measures have been computed as the traffic rate (messages/second) is varied. The measured parameters are the overall percentage of packets received successfully, the average number of requests per successful acquisition of QoS path, the average number of requests per session, and the average QoS path acquisition time.

In both sets of experiments, it can be observed that the average overall percentage of successfully received packets drops as the traffic rate is increased. Also, as expected by the theoretical analysis, this percentage is consistently the smallest in the omnidirectional case ($dir = 1$). For example, in the first set of experiments summarized in Fig. 11, the overall percentage of successfully received packets ranges from 45.64 for a mean traffic rate of 1 messages/sec, down to 27.70 for a mean traffic rate of 10 messages/sec. This percentage is higher with the two-antenna case, and ranges from 53.69, down to 44.53. The highest percentage is obtained in the four-antenna case which ranges from 90.59, down to 81.80. Also, simulation shows that the average number of requests per successful acquisition of a QoS path, the total number of requests per session (i.e., including sessions that were not able to acquire a path), and the average QoS path acquisition time are consistently higher for the omnidirectional case, followed by the two-antenna and the four-antenna cases. This is due to the fact that it is increasingly easier for the network to acquire a QoS path as the number of antennas increases. The second set of experiments, which have been done for a longer data message length of 100 Mbytes, shows a decrease in the overall percentage of successfully received packets due to an increase in the total traffic. The overall percentage of successfully received packets, and path acquisition time measurements, follow the same trends as the first set of experiments, showing a considerable advantage with the

Overall % of Successful Pakets											
Data message rate	*1.00*	*2.00*	*3.00*	*4.00*	*5.00*	*6.00*	*7.00*	*8.00*	*9.00*	*10.00*	*Average*
1 dir	45.64	36.96	31.42	26.88	25.04	22.30	22.16	24.07	23.27	19.26	*27.70*
2 dir	53.64	49.74	43.84	44.28	43.02	49.13	41.40	42.63	38.16	39.51	*44.53*
4 dir	90.59	88.98	86.64	82.48	82.01	76.19	75.27	77.86	78.67	79.36	*81.80*

Average Number of Requests Per Successful Aquisition of QoS Path											
Data message rate	*1.00*	*2.00*	*3.00*	*4.00*	*5.00*	*6.00*	*7.00*	*8.00*	*9.00*	*10.00*	*Average*
1 dir	1.51	1.65	1.73	1.73	1.69	1.65	1.58	1.71	1.64	1.65	*1.65*
2 dir	1.34	1.41	1.48	1.43	1.46	1.45	1.47	1.42	1.49	1.45	*1.44*
4 dir	1.28	1.33	1.39	1.41	1.35	1.39	1.37	1.46	1.34	1.39	*1.37*

Average Number of Requests per Session											
Data message rate	*1.00*	*2.00*	*3.00*	*4.00*	*5.00*	*6.00*	*7.00*	*8.00*	*9.00*	*10.00*	*Average*
1 dir	2.33	2.51	2.60	2.64	2.66	2.70	2.69	2.70	2.70	2.73	*2.62*
2 dir	2.11	2.20	2.32	2.32	2.35	2.26	2.36	2.35	2.42	2.38	*2.31*
4 dir	1.44	1.51	1.60	1.69	1.65	1.76	1.77	1.82	1.70	1.72	*1.67*

Avgerage QoS Path Acquisition Time											
Data message rate	*1.00*	*2.00*	*3.00*	*4.00*	*5.00*	*6.00*	*7.00*	*8.00*	*9.00*	*10.00*	*Average*
1 dir	11.39	14.56	16.12	16.08	15.30	14.34	13.01	15.80	14.28	14.34	*14.52*
2 dir	7.91	9.28	10.90	9.80	10.42	10.31	10.58	9.63	11.03	10.12	*10.00*
4 dir	6.65	7.69	8.83	9.30	8.10	8.94	8.55	10.50	7.83	8.94	*8.53*

Fig. 11. Simulation results table. Data message length: 10 MB.

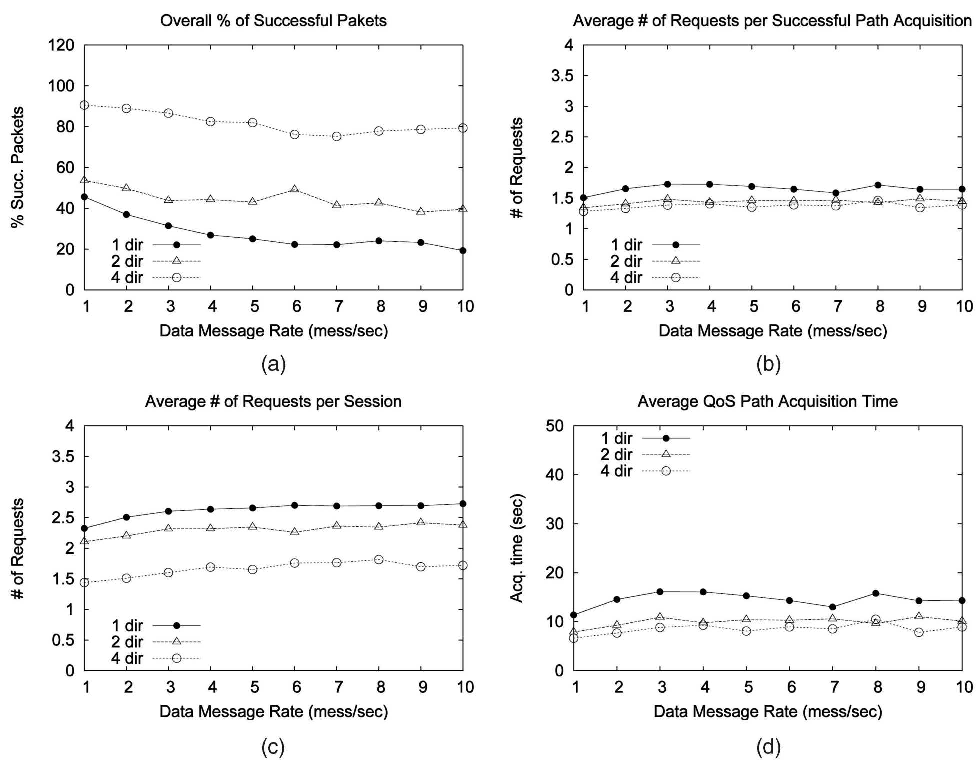

Fig. 12. Simulation results. Data message length: 10 MB.

increase in number of antennas. This confirms the analysis and our prior conclusions.

These simulation results clearly demonstrate the increased efficiency and performance of the network as the number of directional antennas increases. As indicated earlier, this increased performance is due to a considerable increase in the spatial reuse and the ability for each node to simultaneously send or receive data in different directions. This functionally increases the effective number of data slots by a multiple of the number of antennas (or directions) used. This effect significantly improves performance. As the data show, the increase in performance or speedup factor, when the number of antennas is increased by a factor of 2 (i.e., doubled from 1 to 2, and then from 2 to 4) is significant (speedup factor >1). However, it is still below the theoretical speedup factor of 2. For the first set of experiments, for example, the data show that ratio of the overall average percentage (average for all data traffic rates) of successful packets in the two-antenna case to the one-antenna case is 1.61, which is >1 and <2. The ratio of the

Overall % of Successful Packets										
Data message rate	***1.00***	***2.00***	***3.00***	***4.00***	***5.00***	***6.00***	***7.00***	***8.00***	***9.00***	***10.00***	***Average***
1 dir	42.56	39.36	29.20	27.99	25.96	25.72	25.60	21.69	18.70	22.10	*27.89*
2 dir	51.00	53.48	46.79	42.68	41.79	39.05	43.70	35.92	34.09	41.67	*43.02*
4 dir	86.31	85.05	78.75	82.60	81.15	79.50	81.85	78.78	80.61	74.63	*80.92*

Average Number of Requests Per Successful Aquisition of QoS Path											
Data message rate	***1.00***	***2.00***	***3.00***	***4.00***	***5.00***	***6.00***	***7.00***	***8.00***	***9.00***	***10.00***	***Average***
1 dir	1.61	1.69	1.64	1.71	1.78	1.65	1.64	1.60	1.64	1.64	*1.66*
2 dir	1.33	1.41	1.50	1.45	1.52	1.49	1.43	1.43	1.40	1.38	*1.43*
4 dir	1.29	1.38	1.48	1.41	1.42	1.36	1.40	1.41	1.36	1.40	*1.39*

Average Number of Requests per Session											
Data message rate	***1.00***	***2.00***	***3.00***	***4.00***	***5.00***	***6.00***	***7.00***	***8.00***	***9.00***	***10.00***	***Average***
1 dir	2.40	2.48	2.60	2.65	2.68	2.66	2.66	2.69	2.72	2.69	*2.62*
2 dir	2.15	2.18	2.30	2.36	2.39	2.41	2.31	2.45	2.43	2.30	*2.33*
4 dir	1.53	1.62	1.79	1.68	1.73	1.69	1.69	1.73	1.66	1.80	*1.69*

Avgerage QoS Path Acquisition Time											
Data message rate	***1.00***	***2.00***	***3.00***	***4.00***	***5.00***	***6.00***	***7.00***	***8.00***	***9.00***	***10.00***	***Average***
1 dir	13.54	15.30	14.26	15.81	17.29	14.50	14.29	13.32	14.25	14.12	*14.67*
2 dir	7.71	9.45	11.24	10.28	11.62	11.11	9.81	9.77	9.03	8.79	*9.88*
4 dir	6.73	8.71	10.92	9.41	9.69	8.38	9.17	9.39	8.24	9.12	*8.97*

Fig. 13. Simulation results table. Data message length: 100 MB.

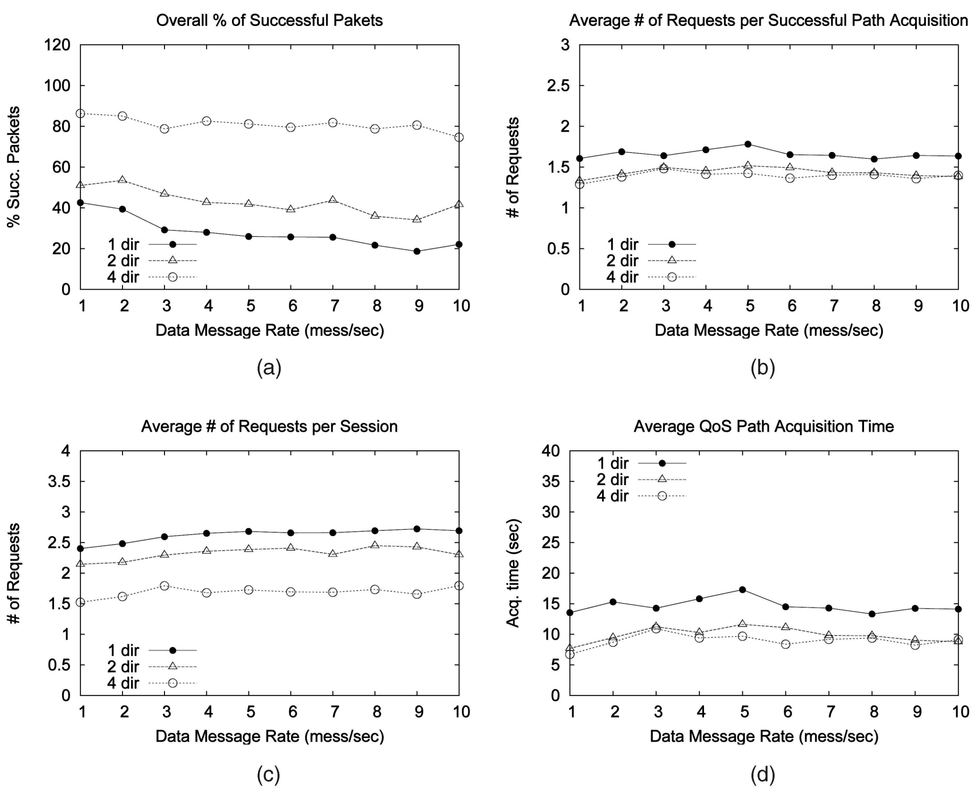

Fig. 14. Simulation results. Data message length: 100 MB.

four-antenna case to the two-antenna case is 1.84, which is also >1 and <2, and the ratio for the four-antenna case to the one-antenna case is 2.95, which is <4.

4.4 Additional Directional Antenna Trade-Offs and Future Research

It is worth noting that in highly mobile MWNs, increased overhead is due to discovering, exchanging, and maintaining topology information between nodes. In the directional antenna environment, this overhead would be higher than that of the omnidirectional case. In addition to keeping track of the slot allocation status of the one, and two-hop neighbors, each node needs to maintain updated information about the direction where each neighbor is located. This can be accomplished by using the techniques discussed in Section 1 of this paper. This overhead would increase in a highly mobile environment due to the necessity of more frequent updates of node locations, and the added exchange of location-related control messages. However, the gain in overhead due to the significant increase in spatial reuse, power savings, reduced path hop count, reduced end-to-end delay, and higher throughput, offsets this increase. The effects of mobility on this protocol, as well as optimization techniques employed to reduce this overhead, is a rich area of future research in this field, and is currently being investigated. Additionally, this protocol is a heuristic approach to a directional antenna version of the scheduling problem in the TDMA environment, which has been proven to be NP-complete [18], [35], [36], [37]. Theoretical bounds and comparison with optimal path assignment is another possible area of future research.

5 Conclusions

In this paper, a protocol for TDMA-based bandwidth reservation for QoS routing in multihop wireless networks using directional antennas has been presented. The protocol takes advantage of the significant increase in spatial reuse provided by the directional antenna environment, which drastically increases the efficiency of communication in this type of networks. This is due to the reduction in signal interference and the amount of power necessary to establish and maintain communication sessions. The simulation results clearly show a significant gain in the performance with an increase in the number of successfully received packets, as well as a decrease in the QoS path acquisition time. However, this gain in the performance is still below the theoretical speedup factor. In the future, we intend to improve this protocol through additional optimization techniques. In addition, we intend to perform more simulations in order to further study, analyze, and improve the performance of the protocol under different network environments and traffic conditions.

Acknowledgments

This work was supported in part by the US National Science Foundation (NSF) grants CCR 0329741, CCR 9900646, ANI 0073736, and EMI 0130806 and UAE Research grant 03-03-9-11/08.

References

[1] L. Bao and J.J. Garcia-Luna-Aceves, "Transmission Scheduling in Ad Hoc Networks with Directional Antennas," *Proc. Eighth Ann. Int'l Conf. Mobile Computing and Networking,* pp. 48-58, Sept. 2002.

[2] Q. Dai and J. Wu, "Construction of Power Efficient Routing Tree for Ad Hoc Wireless Networks Using Directional Antenna," *Proc. 24th Int'l Conf. Distributed Computing Systems Workshops,* pp. 718-722, Mar. 2004.

[3] I. Jawhar and J. Wu, "Resource Allocation in Wireless Networks Using Directional Antennas," *Proc. Fourth Int'l Conf. Pervasive Computing and Comm. (PerCom '06),* pp. 318-327, Mar. 2006.

[4] I. Martinez and J. Altuna, "A Cross-Layer Design for Ad Hoc Wireless Networks with Smart Antennas and QoS Support," *Proc. 15th IEEE Int'l Symp. Personal, Indoor and Mobile Radio Comm. (PIMRC '04),* vol. 1, pp. 589-593, Sept. 2004.

[5] R. Ramanathan, J. Redi, C. Santivanez, D. Wiggins, and S. Polit, "Ad Hoc Networking with Directional Antennas: A Complete System Solution," *Proc. IEEE Wireless Comm. and Networking Conf. (WCNC),* vol. 1, pp. 21-25, Mar. 2004.

[6] D. Saha, S. Roy, S. Bandyopadhyay, T. Ueda, and S. Tanaka, "A Distributed Feedback Control Mechanism for Priority-Based Flow-Rate Control to Support QoS Provisioning in Ad Hoc Wireless Networks with Directional Antenna," *Proc. IEEE Int'l Conf. Comm.,* vol. 7, pp. 4172-4176, June 2004.

[7] D. Lal, V. Jain, Q.A. Zeng, and D.P. Agrawal, "Performance Evaluation of Medium Access Control for Muliple-Beam Antenna Nodes in a Wireless Lan," *IEEE Trans. Parallel and Distributed Systems,* vol. 15, no. 12, pp. 1117-1129, Dec. 2004.

[8] R. Ramanathan, "Antenna Beamforming and Power Control for Ad Hoc Networks," *Mobile Ad Hoc Networking,* S. Basagni, M. Conti, S. Giordano, and I. Stojmenovic, eds., pp. 139-173, IEEE, Inc., 2004.

[9] R. Ramanathan, "On the Performance of Ad Hoc Networks with Beamforming Antennas," *Proc. ACM MobiHoc,* Oct. 2001.

[10] R.R. Choudhury, X. Yang, R. Ramanathan, and N. Vidya, "Using Directional Antennas in Ad Hoc Networks," *Proc. ACM MOBICOM,* Sept. 2002.

[11] Y.-B. Ko, V. Shankarkumar, and N.H. Vaidya, "Medium Access Control Protocols Using Directional Antennas in Ad Hoc Networks," *Proc. IEEE INFOCOM '00, 19th Ann. Joint Conf. IEEE Computer and Comm. Soc.,* vol. 1, nos. 26-30, pp. 13-21, Mar. 2000.

[12] M. Sanchez, T. Giles, and J. Zander, "CSMA/CA with Beam Forming Antennas in Multi-Hop Packet Radio," *Proc. Swedish Workshop Wireless Ad Hoc Networks,* Mar. 2001.

[13] N.S. Fahmy, T.D. Todd, and V. Kezys, "Ad Hoc Networks with Smart Antennas Using IEEE 802.11-Based Protocols," *Proc. IEEE Int'l Conf. Comm. (ICC),* 2002.

[14] M. Takai, J. Martin, A. Ren, and R. Bagrodia, "Directional Virtual Carrier Sensing for Directional Antennas in Mobile Ad Hoc Networks," *Proc. ACM MOBIHOC,* June 2002.

[15] S. Bandyopadhyay, K. Hasuike, S. Horisawa, and S. Taware, "An Adaptive MAC and Directional Routing Protocol for Ad Hoc and Wireless Network Using ESPAR Antenna," *Proc. ACM MOBIHOC,* Oct. 2001.

[16] A. Nasipuri, S. Ye, J. You, and R.E. Hiromoto, "A MAC Protocol for Mobile Ad Hoc Networks Using Directional Antennas," *Proc. IEEE Wireless Comm. and Networking Conf. (WCNC '00),* vol. 3, nos. 23-28, pp. 1214-1219, Sept. 2000.

[17] A. Ephremides and T.V. Truong, "Scheduling Broadcasts in Multihop Radio Networks," *IEEE Trans. Comm.,* vol. 38, no. 4, pp. 456-460, Apr. 1990.

[18] S. Even, O. Goldreich, S. Moran, and P. Tong, "On the NP-Completeness of Certain Network Testing Problems," *Networks,* vol. 14, pp. 1-24, 1984.

[19] R. Ramaswami and K.K. Parhi, "Distributed Scheduling of Broadcasts in a Radio Network," *Proc. IEEE INFOCOM '89, Eighth Ann. Joint Conf. IEEE Computer and Comm. Soc., Technology: Emerging or Converging,* vol. 2, pp. 497-504, Apr. 1989.

[20] R. Ramanathan, "A Unified Framework and Algorithm for Channel Assignment in Wireless Networks," *Wireless Networks,* vol. 5, no. 2, pp. 81-94, 1999.

[21] K. Dyberg, L. Farman, F. Eklof, J. Gronkvist, U. Sterner, and J. Rantakokko, "On the Performance of Antenna Arrays in Spatial Reuse TDMA Ad Hoc Networks," *Proc. IEEE Military Comm. Conf. (MILCOM),* Oct. 2002.

[22] R.R. Choudhury and N. Vaidya, "Performance of Ad Hoc Routing Using Directional Antennas," *Ad Hoc Networking for Pervasive Systems,* vol. 3, pp. 157-173, Mar. 2005.

[23] A.K. Saha and D.B. Johnson, "Routing Improvement Using Directional Antennas in Mobile Ad Hoc Networks," *Proc. IEEE GLOBECOM,* vol. 5, no. 29, pp. 2902-2908, Nov. 2004.

[24] H. Jasani and K. Yen, "Performance Improvement Using Directional Antennas in Ad Hoc Networks," *Int'l J. Computer Science and Network Security,* vol. 6, no. 6, pp. 180-188, June 2006.

[25] L. Bao and J.J. Garcia-Luna-Aceves, "A New Approach to Channel Access Scheduling for Ad Hoc Networks," *Proc. ACM Seventh Ann. Int'l Conf. Mobile Computing and Networking,* July 2001.

[26] I. Jawhar and J. Wu, "Race-Free Resource Allocation for QoS Support in Wireless Networks," *Ad Hoc and Sensor Wireless Networks: An Int'l J.,* vol. 1, no. 3, pp. 179-206, May 2005.

[27] H. Singh and S. Singh, "Doa-Aloha: Slotted Aloha for Ad Hoc Networking Using Smart Antennas," *Proc. IEEE Vehicular Technology Conf. (VTC),* Oct. 2003.

[28] H. Singh and S. Singh, "Smart-Aloha for Multi-Hop Wireless Networks," *J. Mobile Networks and Applications,* vol. 10, no. 5, pp. 651-662, Sept. 2005.

[29] S.L. Preston, D.V. Thiel, T.A. Smith, S.G. O'Keefe, and W.L. Jun, "Base-Station Tracking in Mobile Communications Using a Switched Parasitic Antenna Array," *IEEE Trans. Antennas and Propagation,* vol. 46, no. 6, pp. 841-844, June 1998.

[30] C. Cordeiro and D.P. Agrawal, *Ad Hoc and Sensor Networks.* World Scientific, 2006.

[31] S. Chen and K. Nahrstedt, "Distributed Quality-of-Service Routing in Ad Hoc Networks," *IEEE J. on Selected Areas in Comm.,* vol. 17, no. 8, pp. 1488-1505, Aug. 1999.

[32] C.R. Lin and J.-S. Liu, "QoS Routing in Ad Hoc Wireless Networks," *IEEE J. on Selected Areas in Comm.,* vol. 17, no. 8, pp. 1426-1438, Aug. 1999.

[33] W.-H. Liao, Y.-C. Tseng, and K.-P. Shih, "A TDMA-Based Bandwidth Reservation Protocol for QoS Routing in a Wireless Mobile Ad Hoc Network," *Proc. IEEE Int'l Conf. Comm. (ICC '02),* vol. 5, pp. 3186-3190, 2002.

[34] C.E. Perkins, *Ad Hoc Networking.* Addison-Wesley, 2001.

[35] I. Jawhar and J. Wu, "Qos Support in TDMA-Based Mobile Ad Hoc Networks," *J. Computer Science and Technology,* vol. 20, no. 6, pp. 797-810, Nov. 2005.

[36] E.L. Lloyd, "Broadcast Scheduling for TDMA in Wireless Multihop Networks," *Handbook of Wireless Networks and Mobile Computing,* I. Stojmenovic, ed., pp. 347-370, John Wiley and Sons, Inc., 2002.

[37] S.T. McCormick, "Optimal Approximation of Sparse Hessians and Its Equivalence to a Graph Coloring Problem," *Math. Programming,* vol. 2, no. 26, pp. 153-171, 1983.

Imad Jawhar received the BS and MS degrees in electrical engineering from the University of North Carolina at Charlotte, and the MS degree in computer science and the PhD degree in computer engineering from Florida Atlantic University, where he also served as a faculty member for several years. He is an assistant professor at the College of Information Technology at UAE University. He has published numerous papers in international journals, conference proceedings, and book chapters. He worked at Motorola, USA, as an engineering task leader. He was involved in the design and development of software programs that control the world's leading portable radios, and cutting-edge communication products and systems, providing maximum flexibility and customization. He was also the president and the owner of the Atlantic Computer Training and Consulting, a company based on South Florida, which trained professionals and conducted numerous classes in the latest computer system applications. Its customers included small businesses as well as large corporations such as GE, Federal Express, and International Paper. His current research focuses in the areas of wireless networks, mobile computing, sensor networks, routing protocols, and distributed and multimedia systems. He served on numerous international conference committees and reviewed publications for many international journals, conferences, and other research organizations such as the United States National Science Foundation. He is a member of the IEEE and the ACM.

Jie Wu is a chair and a professor in the Department of Computer and Information Sciences, Temple University. Prior to joining Temple University, he was a program director at the National Science Foundation. His research interests include wireless networks and mobile computing, routing protocols, fault-tolerant computing, and interconnection networks. He has published more than 500 papers in various journals and conference proceedings. He serves in the editorial board of the *IEEE Transactions on Computers*, the *Journal of Parallel and Distributed Computing*, and the *IEEE Transactions on Mobile Computing*. He is a program cochair for IEEE INFOCOM 2011. He was also a general cochair for IEEE MASS 2006, IEEE IPDPS 2008, and DCOSS 2009. He has served as an IEEE Computer Society distinguished visitor and is the chairman of the IEEE Technical Committee on Distributed Processing (TCDP). He is a fellow of the IEEE.

Dharma P. Agrawal (M'74, F'87) is the Ohio Board of Regents distinguished professor of computer science and the founding director for the Center for Distributed and Mobile Computing in the Department of ECECS, University of Cincinnati, Ohio. He was a visiting professor of ECE at the Carnegie Mellon University, on sabbatical leave during the Autumn 2006 and Winter 2007 Quarters. He has been a faculty member at the N.C. State University, Raleigh (1982-1998), and the Wayne State University, Detroit (1977-1982). His recent research interests include resource allocation and security in mesh networks, efficient query processing and security in sensor networks, and heterogeneous wireless networks. His coauthored introductory textbook on *Wireless and Mobile Computing* has been widely accepted throughout the world and has been reprinted both in China and India and translated into Korean and Chinese languages. His second coauthored book on *Ad Hoc and Sensor Networks* was published in Spring of 2006 and coedited book entitled *Encyclopedia on Ad Hoc and Ubiquitous Computing* has just been published. He has given tutorials and extensive training courses in various conferences in USA, and numerous institutions in Taiwan, Korea, Jordan, Malaysia, Saudi Arabia, and India, in the areas of ad hoc and sensor networks and mesh networks. He is an editor of the *Journal of Parallel and Distributed Systems*, the *International Journal on Distributed Sensor Networks*, the *International Journal of Ad Hoc and Ubiquitous Computing (IJAHUC)*, and the *International Journal of Ad Hoc and Sensor Wireless Networks*. He has served as an editor of the IEEE *Computer Magazine,* the *IEEE Transactions on Computers,* and the *International Journal of High Speed Computing.* He has been the program chair and the general chair for many international conferences and meetings. He has received numerous certificates and awards from the IEEE Computer Society and been elected as a core member. He was awarded the *"Third Millennium Medal,"* by the IEEE for his outstanding contributions. He has also delivered keynote speech for 10 international meetings. He also has five patents in wireless networking area. He has also been named as an ISI Highly Cited Researcher in computer science. He is a fellow of the IEEE, the ACM, the AAAS, and the WIF. He has been selected for the 2008 Harry Goode Memorial award from the IEEE Computer Society.