Open access • Journal Article • DOI:10.1109/JSAC.2003.814897

TRACE: time reservation using adaptive control for energy efficiency — Source link

Bulent Tavli, Wendi Heinzelman

Institutions: University of Rochester

Published on: 20 Mar 2003 - IEEE Journal on Selected Areas in Communications (IEEE)

Topics: Media access control, Network packet, Quality of service, Throughput and Broadcasting (networking)

TRACE: Time Reservation Using Adaptive Control for Energy Efficiency

Bulent Tavli, *Student Member, IEEE,* and Wendi B. Heinzelman, *Member, IEEE*

***Abstract*—Time reservation using adaptive control for energy efficiency (TRACE) is a time frame based media access control (MAC) protocol designed primarily for energy-efficient reliable real-time voice packet broadcasting in a peer-to-peer, single-hop infrastructureless radio network. Such networks have many application areas for various scenarios that obey a strongly connected group mobility model, such as interactive group trips, small military or security units, and mobile groups of hearing impaired people. TRACE is a centralized MAC protocol that separates contention and data transmission, providing high throughput, bounded delay, and stability under a wide range of data traffic. Furthermore, TRACE uses dynamic scheduling of data transmissions and data summarization prior to data transmission to achieve energy efficiency, which is crucial for battery operated lightweight radios. In addition, energy dissipation is evenly distributed among the nodes by switching network controllers when the energy from the current controller is lower than other nodes in the network, and reliability is achieved through automatic controller backup features. TRACE can support multiple levels of quality-of-service, and minimum bandwidth and maximum delay for voice packets are guaranteed to be within certain bounds. In this paper, we describe TRACE in detail and evaluate its performance through computer simulations and theoretical analysis.**

***Index Terms*—Carrier sense multiple access, energy efficiency, multiaccess communication, network reliability, packet reservation multiple access, protocols, quality-of-service (QoS), real-time systems, speech communication.**

I. INTRODUCTION

MANY common applications require a peer-to-peer single-hop infrastructureless reliable radio network architecture that enables real-time communication. Application areas of such networks include all kinds of group communications within a collection of mobile nodes that move according to a group mobility model, like the reference point group mobility model [1], without loosing full connectivity. For these types of single-hop wireless networks, the media access control (MAC) layer is the most important design block in the network, as it controls access to the shared medium. The objective of controlled access is to avoid simultaneous transmission attempts (that will result in collisions) while maintaining maximum throughput, minimum energy dissipation, and bounded packet delay of the whole network [2]–[5].

MAC protocols can be classified into two main categories: centralized and distributed. In a distributed MAC protocol, radios communicate without a central controller or base station. In other words, every radio should create its own access to the medium through a predetermined set of rules (e.g., IEEE 802.11[1] [6]). A centralized MAC protocol, on the other hand, has a controller node or a base station that is the maestro of the network (e.g., Bluetooth [7]). All the nodes in the network access the medium through some kind of schedule determined by the controller.

Centralized MAC protocols are generally more deterministic than distributed MAC protocols, which is a desirable feature for real-time traffic with delay constraints. As a result, it is advantageous to use a centralized MAC protocol in a single-hop network that supports real-time traffic delivery. For example, a distributed MAC protocol such as IEEE 802.11 cannot guarantee bandwidth or delay constraints or fair medium access. In fact, all of these parameters are functions of the data traffic, and they become unpredictable and often unacceptable at high data rates [8]. However, some centralized algorithms can guarantee some of the above requirements within certain ranges by making use of coordination via scheduling [9]. Furthermore, when using a distributed MAC protocol such as IEEE 802.11, all nodes should be active all the time, because they do not know when the next transmission is going to take place [10]. However, using a centralized MAC protocol such as Bluetooth, nodes can enter sleep mode frequently due to the explicit polling of the slave nodes by the master node, which is an effective method to save power.

In a centralized MAC protocol, the two most important issues are the controller assignment and the data transmission schedule, which correspond to the coordinator and the coordination, respectively. The coordinator could be a fixed predetermined radio, which is the sole controller for the entire network lifetime. The main drawback of this approach is that whenever the controller dies, the whole network also dies. The controller dissipates more energy than other nodes because of its additional processes and transmissions/receptions. Because of this higher energy dissipation, most possibly the controller will run out of energy before all the other nodes, leaving the entire network inoperable for the rest of the network lifetime, even though many other remaining nodes have enough energy to carry on transmissions/receptions. The data transmission schedule could also be fixed, but this does not allow the system to adapt to dynamic environments such as nodes entering the network. The alternative approach to a fixed controller and schedule is dynamic controller switching and schedule updating, which is a remedy for the problems described above. However, this approach comes with its own problems: overhead in controller handover and increased overhead in the schedule updates.

Manuscript received September 2, 2002; revised April 9, 2003.

The authors are with the Department of Electrical and Computer Engineering, University of Rochester, Rochester, NY 14627 USA (e-mail: tavli@ece.rochester.edu; wheinzel@ece.rochester.edu).

Digital Object Identifier 10.1109/JSAC.2003.814897

[1] Throughout this paper, “IEEE 802.11” is used for IEEE 802.11 in infrastructureless mode.

The information content in a single-hop broadcast medium may be higher than the usable range of a single node, in which case the nodes should select to receive only certain data packets. For example, if the number of simultaneous conversations in a group of people, communicating through a single-hop broadcast network, exceeds a certain level, then each user should select a subset of the voice packets based on some discrimination criteria like proximity, and discard the rest of the packets. The straightforward approach, which is listening to all data transmissions, keeping the ones desired, and discarding the others, is a highly inefficient way of discriminating data. An energy efficient method is information summarization prior to data transmission [11].

In this paper, we propose time reservation using adaptive control for energy efficiency (TRACE), a new MAC protocol that combines different features of centralized and distributed MAC protocols to achieve high performance for peer-to-peer single-hop infrastructureless wireless networks. TRACE uses dynamic controller switching and schedule updating to adapt to a changing environment and reduce energy dissipation in the nodes. Other features of TRACE, such as information summarization, data stream continuation monitoring, multi-level controller backup, priority-based channel access, and contention for channel access reinforce the energy-efficiency, reliability, bounded delay, and maximized throughput of the network. Although TRACE can be categorized as a MAC protocol, due to its cross layer design it performs some of the functionalities of the other layers, such as data discrimination through information summarization.

TRACE has been designed to be a very energy efficient, reliable protocol to support real-time broadcasting. Thus, TRACE is well suited to fulfill the tactical communication requirements of a small to medium size military group (i.e., a squad) or a law enforcement group (i.e., police officers pursuing a criminal or airport security personnel searching a group of passengers), where the members of the network may want to communicate simultaneously with each other. A group of researchers, students or tourists having a field trip may also benefit from TRACE-based networks. An interesting application that fits very well to a TRACE-based network is communication among a group of hearing disabled people who communicate with sign language. Since vision is the only possible means of communication for such a group, without direct vision (i.e., you cannot see simultaneously a person at your left and another at your right), it is not possible to have group communication in all situations. If each person has a PDA with a small camera and a low-resolution monitor large enough to display the signs, possibly with several panels, and an MPEG coder [12], which enables high compression, then it is possible to create a communication network for hearing disabled people.

The remainder of this paper is organized as follows. Section II describes the TRACE protocol in detail. Section III provides analysis of the performance of TRACE and simulations to compare TRACE with other MAC protocols. Section IV gives some discussion of the features of TRACE, and Section V gives an overview of related work. Section VI concludes the paper.

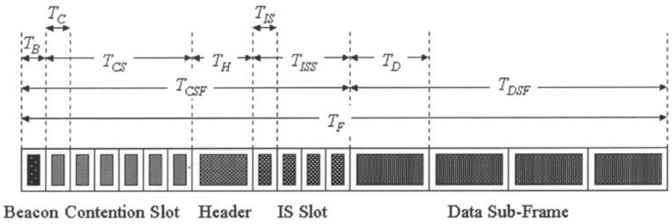


Fig. 1. Symbolic representation of TRACE frame format.

II. TRACE

A. Overview

TRACE is an energy-efficient dynamic time-division multiple-access (TDMA) protocol designed for real-time data broadcasting. In TRACE, data transmission takes place according to a dynamically updated transmission schedule. Initial access to data slots are through contention, but once a node reserves a data slot, its reservation for a data slot in the subsequent frames continues automatically as long as the node continues to broadcast a packet in each frame. Thus, nodes only need to contend for data slots at the beginning of data bursts.

A controller in the network is responsible for creating the TDMA schedule based on which nodes have continued reservations from previous frames and which have successfully contended for data slots in the current frame. The controller transmits this schedule to the rest of the nodes in the network at the beginning of the data subframe. Whenever the energy of the controller drops below the energy level of the other nodes in the network by more than a set amount, it assigns another radio with higher energy than itself as the next controller. Controller handover takes place during the TDMA schedule transmission by specifying the ID of the new controller.

Finally, if the number of transmissions in a frame exceeds a predetermined threshold, each node listens only to data from certain nodes. Each node determines which transmitters to listen to based on information obtained from all the nodes during the information summarization (IS) slot.

The following sections describe these ideas in more detail.

B. Basic Operation

TRACE is organized around time frames with duration matched to the periodic rate of voice packets. The frame format is presented in Fig. 1. Each frame consists of two subframes: a control subframe and a data subframe. The control subframe consists of a beacon message, a contention slot, a header message, and an IS slot.

At the beginning of every frame, the controller node transmits a beacon message. This is used to synchronize all the nodes and to signal the start of a new frame. The contention slot, which immediately follows the beacon message, consists of N_c subslots. Upon hearing the beacon, nodes that have data to send but did not reserve data slots in the previous frame, randomly choose subslots to transmit their requests. If the contention is successful (i.e., no collisions), the controller grants a data slot to the contending node. The controller then sends the header, which includes the data transmission schedule of the current frame. The transmission schedule is a list of nodes that have been granted data slots in the current frame along with their data slot numbers. A contending node that does not hear its ID in the schedule understands that its contention was unsuccessful (i.e., a collision occurred or all the data slots are already in use) and contends again in the following frame. If the waiting time for a voice packet during contention for channel access exceeds the threshold $T_{\rm drop}$, it is dropped. The header also includes the ID of the controller for the next frame, which is determined by the current controller according to the node energy levels.

The IS slot begins just after the header slot and consists of N_D subslots. Nodes that are scheduled to transmit in the data subframe transmit a short IS message exactly in the same order as specified by the data transmission schedule. An IS message includes the energy level of the transmitting node, enabling the controller node to monitor the energy level of the entire network, and an end-of-stream bit, which is set to one if the node has no data to send. Each receiving node records the received power level of the transmitting node and inserts this information into its IS table. The information in the IS table is used as a proximity metric for the nodes (i.e., the higher the received power the shorter the distance between transmitter and receiver nodes). Using the receive signal strength to estimate the relative distance of the transmitter to the receiver is a method employed in previous studies [13], [14]. If the number of transmissions in a particular frame is higher than a predetermined number of transmissions, $N_{\max}$, each node schedules itself to wake up for the top $N_{\max}$ transmissions that are the closest transmitters to the node.[2] Hence, the network is softly partitioned into many virtual clusters based on the receivers; this is fundamentally different from transmitter based network partitioning.

The data subframe is broken into constant length data slots. Nodes listed in the schedule in the header transmit their data packets at their reserved data slots. Each node listens to at most $N_{\max}$ data transmissions in a single frame, therefore, each node is on for at most $N_{\max}$ data slots. All nodes are in the sleep mode after the last reserved data slot until the beginning of the next frame.

If the power level of the controller node is lower than any other node by a predetermined threshold, then in the next frame controller handover takes place. The controller node assigns another node (any other node in the network with energy level higher than that of the controller) as the controller, effective with the reception of the header packet. Upon receiving the header packet, the node assigned to be the controller assumes the controller duties.

A node keeps a data slot once it is scheduled for transmission as long as it has data to send. A node that sets its end-of-stream bit to one because it has no more data to send will not be granted channel access in the next frame (i.e., it should contend to get a data slot once it has new data to send). Automatic renewal of data slot reservation enables real-time data streams to be uninterrupted [15].

C. Initial Startup

At the initial startup stage, a node listens to the medium to detect any ongoing transmissions for one frame time T_F, because it is possible that there might already be an operational network. If no transmission is detected, then the node picks a random time, smaller than the contention slot duration T_{CS}, at which to transmit its own beacon signal, and the node listens to the channel until its contention timer expires. If a beacon is heard in this period, then the node stops its timer and starts normal operation. Otherwise, when the timer expires, the node sends a beacon and assumes the controller position. In case there is a beacon collision, none of the colliding nodes will know it, but the other nodes hear the collision, so the initial setup continues. All the previously collided nodes, and the nodes that could not detect the collision(s) because of capture, will learn of the collisions with the first successful beacon transmission.

D. Prioritization

TRACE supports an optional prioritized operation mode. In this mode, the nodes have three preassigned priority levels, of which priority level-1 ($PL1$) is the highest priority and $PL3$ is the lowest priority. The highest level has the highest quality-of-service (QoS) and the lowest level has the lowest QoS. Prioritization is incorporated into the basic protocol operation at three points: contention, scheduling, and receiver based soft clustering.

In the contention stage, $PL1$, $PL2$, and $PL3$ nodes have N_{C1}, N_{C2}, and N_{C3} number of nonoverlapping contention slots, respectively. The number of contention slots per node is higher for the higher priority levels, which results in less contention for higher priority nodes.

In scheduling, $PL1$ and $PL2$ nodes are always given channel access, even if all the data slots are reserved. If all the data slots are reserved, then reservations of $PL3$ nodes are canceled starting from the latest reservation and granted to the higher priority nodes.

All the nodes should listen to data from $PL1$ nodes, whether or not they are close to the nodes. Prioritization does not affect the general protocol operation, because we assume that the number of $PL1$ and $PL2$ nodes is much less than the number of $PL3$ nodes.

E. Receiver-Based Soft Cluster Creation

Each node creates its receiver-based listening cluster, which has a maximum of $N_{\max}$ members, by choosing the closest nodes based on the proximity information obtained from the received power from the transmissions in the IS slot. Priority has precedence over proximity; therefore, transmissions by $PL1$ nodes are always included in the listening cluster by removing the furthest node in the cluster.

F. Reliability

In case the controller node fails, the rest of the network should be able to compensate for this situation and should be able to

---

[2] Note that other methods of deciding which nodes to listen to can be used within the TRACE framework by changing what data nodes send in the IS slot.

continue normal operation as fast as possible. Failure of the controller manifests itself at two possible points within a frame: beacon transmission and header transmission. A backup controller, assigned by the controller, could listen for the beacon and header and become the controller whenever the controller fails. However, if both the backup controller and the controller die simultaneously, then the network is left dead. Instead of assigning a backup controller, there is a more natural and complete way of backing up the network: the transmission schedule is a perfect list of backup controllers in a hierarchical manner. The first node in the schedule is the first backup controller, the second node is the second controller, and the Nth node is the Nth backup controller.

The backup nodes listen to the beacon, which is a part of normal network operation. If the first backup controller does not hear the beacon for interframe space (IFS) time, then the controller is assumed dead and the first node transmits the beacon. If the beacon is not transmitted for two IFS time, then the second backup controller understands that both the controller and the first backup controller are dead, and transmits the beacon. The backup procedure works in the same way for all the nodes listed in the transmission schedule in the previous frame. If after $(N+1)$ IFS time no beacon is transmitted, then the rest of the nodes understand that the controller and all the backup nodes are dead, and they restart the network. Restartup is the same as the initial network startup, but in this case nodes do not listen for an existing controller for T_F; instead they start right away, because they know the controller is dead and there is no need for waiting.

The response of the network to the controller failure in header transmission is very similar to that of beacon failure. The succeeding backup node transmits the transmission schedule of the previous frame by updating it with the information in the IS slot of the previous frame denoting nodes with reservations that no longer have data to transmit. However, none of the nodes, including the backup nodes, listen to the contention slot, so the transmission schedule cannot be updated for the contending nodes. This is not much of an issue in voice transmission, because packet loss due to delayed channel access causes the early packets to be dropped, which is preferable over packet loss in the middle of a conversation [15]. Since controller node failure is not a frequent event, it is better not to dissipate extra energy on controller backup. If all the backup nodes die simultaneously during header transmission, then the rest of the nodes begin restartup. Also, if there were no transmissions in the previous frame, then in case of a controller failure, nodes just enter restartup (i.e., there are no backup nodes).

III. SIMULATIONS AND ANALYSIS

To test the performance of TRACE, we conducted simulations using the ns software package [16]. We simulated conversational voice coded at 32 kb/s. The channel rate is chosen as 1 Mb/s. We used a perfect channel without any loss or error models. Each node listens to a maximum of five nodes. The transport agent used in the simulations is very similar to UDP, which is a best effort service. All the simulations, unless otherwise stated, are run for 100 s and averaged for three independent runs. Acronyms, descriptions, and values of the parameters used in the simulations are presented in Table I.

TABLE I
PARAMETERS USED IN THE SIMULATIONS

Acronym	Description	Value
T_F	Frame duration	25.0 ms
T_{CSF}	Contention sub-frame duration	3.8 ms
T_{DSF}	Data sub-frame duration	21.2 ms
T_B	Beacon duration	40.0 μs
T_{CS}	Contention slot duration	2.32 ms
T_C	Contention sub-slot duration	40.0 μs
T_H	Header duration (max)	0.44 ms
T_{ISS}	IS slot duration	1.0 ms
T_{IS}	IS sub-slot duration	40 μs
T_D	Data slot duration	0.848 ms
IFS	Inter-frame space	16.0 μs
T_{drop}	Packet drop threshold	50.0 ms
N_D	Number of data slots	25
N_C	Number of contention sub-slots	58
N_{Ci}	Number of contention sub-slots in priority i	3, 5, 50
N_{max}	Maximum listening cluster size	5
P_T	Transmit power	0.6 W
P_{TE}	Transmit electronics power	0.318 W
P_{PA}	Power amplifier power	0.282 W
P_R	Receive power	0.3 W
P_I	Idle power	0.1 W
P_S	Sleep power	0.0 W
m_s	Average spurt duration	1.0 s
m_g	Average gap duration	1.35 s

A. Frame Structure and Packet Sizes

Frame time T_F is chosen to be 25 ms, which is the periodic rate of voice packet generation; of this 25 ms, 21.2 ms is for the data subframe (DSF) and 3.8 ms is for the control subframe (CSF). There are 58 40-μs duration contention subslots, 25 40-μs duration IS subslots, and 25 848-μs duration data slots. The number of contention slots is approximately equal to e times the number of data slots, because the optimal throughput of a slotted ALOHA system is $1/e$. Beacon, contention, and IS packets are all 3 B. The header packet has a variable length of 3–53 B, consisting of 3 B of packet header and 2 B of data for each node to be scheduled. The data packet is 104-B long, consisting of 4 B of packet header and 100 B of data. Variations in the packet sizes are due to the differences in the information content of each packet. Each slot or subslot includes 16 μs of guard band (IFS) to account for switching time and round-trip time.

B. Voice Source Model

In voice source modeling, we assume each node has a voice activity detector, which classifies speech into "spurts" and "gaps" (i.e., gaps are the silent moments during a conversation) [15], [17], [18]. During gaps no data packets are generated, and during spurts data packets are generated in the rate of the speech coder, which is 32 kb/s in our simulations. Both spurts and gaps are exponentially distributed statistically independent random variables, with means m_s and m_g, respectively. In our simulations and analysis, we used the experimentally verified values of m_s and m_g, which are 1.0 and 1.35 s, respectively [18].

C. Energy Models

We used the energy model described in [14], where transmit power consists of a constant transmit electronics part P_{TE} and a variable power amplifier part P_{PA}. Hence, the transmit power P_T can be expressed as the sum of two terms

$$P_T = P_{TE} + P_{PA}. \tag{1}$$

P_{PA} should be adjusted to compensate for the path loss in wave propagation. The maximum distance between the nodes is 250 m in the scenarios we employed, and P_{PA} is set to ensure that maximally separated nodes could hear each other's transmissions. Receive power P_R is dissipated entirely on receiver electronics. Idle power P_l is the power needed to run the electronic circuitry without any actual packet reception. In sleep mode, the radio is shut down so sleep mode power P_S is very low.

D. Throughput

A maximum of 25 nodes can transmit data simultaneously; therefore, the maximum achievable total throughput is 800 kb/s. However, it is not possible to reach this upper bound while ensuring that QoS is met. QoS in the context of voice traffic corresponds to the packet drop ratio R_{PD} due to the packet delay exceeding a certain maximum delay, $T_{\text{drop}} (T_{\text{drop}} = 50 \text{ ms})$. R_{PD} is the ratio of the average number of dropped voice packets per frame and the average number of voice packets generated per frame. Since the voice signals are composed of spurts and gaps, it is possible to support more than 25 users by multiplexing more than 25 conversational speech sources into 25 data slots.

Fig. 2 shows a plot of the average number of data packets generated per frame as a function of the number of nodes in the network. The theoretical value of the average number of data packets generated per frame N_G in a network of N_N nodes is obtained as

$$N_G = \frac{m_s}{m_s + m_g} N_N. \tag{2}$$

Both theoretical and simulation curves increase linearly almost with constant slope with N_N. All the simulation data points are within 3.0% error range of the theoretical curve, with a maximum difference of 0.85 packets per frame at $N_N = 60$. Fig. 2 shows that the average number of voice packets generated per frame is 43% of the number of voice sources.

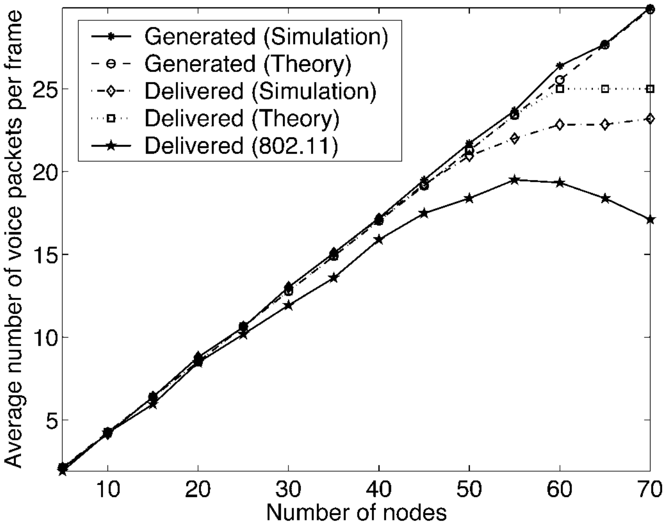


Fig. 2. Average number of voice packets per frame versus total number of nodes with active voice sources.

It is possible to achieve a normalized capacity η[3] of 2.35 conversations per channel with perfect multiplexing of the voice sources over time. This means that TRACE can theoretically support a maximum of 58 nodes with no packet drop. However, the voice sources are independent (i.e., they are not coordinated, as the input pattern is not a design parameter) and it would be too optimistic to expect perfect statistical multiplexing. Therefore, we expect packet drops to occur with fewer than 58 nodes.

The theoretical average number of packets delivered per frame N_A is obtained as

$$N_A = \min\left[\frac{m_s}{m_s + m_g} N_N, N_{DS}\right] \tag{3}$$

where N_{DS} is the total number of data slots in a frame (25 in our simulations). Curves showing the average number of delivered packets per frame obtained from the simulations and theory are in good agreement for $N_N < 50$ (see Fig. 2). However, for $N_N \geq 50$ the difference between the curves is large (i.e., at $N_N = 60$ the difference is 2.1 packets per frame). In theory, we did not consider any packet drops, and we assumed data packets are distributed evenly in all frames. In simulations, both of these assumptions are violated for $N_N > 50$. For $N_N > 58$, the average number of packets per frame exceeds the number of data slots; because of this, in our theoretical model $N_A = 25$, but we cannot achieve this upper bound in the simulations. This is because of the fact that in some frames the number of voice packets are smaller than 25 and in some others much higher than 25. Thus, due to the independent statistical behavior of the voice sources, it is not possible to achieve the upper bound without sacrificing QoS (i.e., R_{PD}).

Fig. 2 also shows the number of data packets delivered per T_F time for IEEE 802.11, which is lower than that of TRACE for all N_N. The maximum difference between TRACE and IEEE 802.11 is 6.1 packets per T_F time at $N_N = 70$, which corresponds to a 26.2% decrease in throughput. For broadcast traffic, IEEE 802.11 does not use the standard four-way handshake

---

[3] The normalized capacity is defined in [18] as the ratio of the maximum number of nodes (i.e., conversations) that can be supported without exceeding the packet drop ratio of 0.01 and the number of channels (data slots).

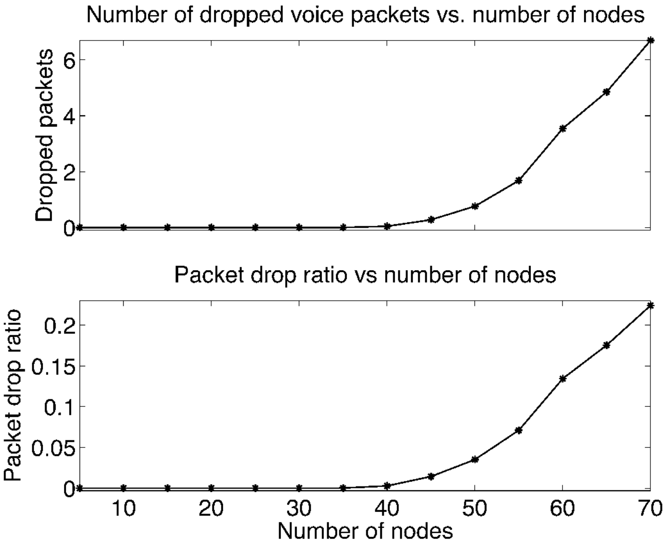


Fig. 3. Upper panel displays the average number of dropped packets per frame as a function of N_N and the lower panel displays the average value of packet drop ratio R_{PD} as a function of N_N.

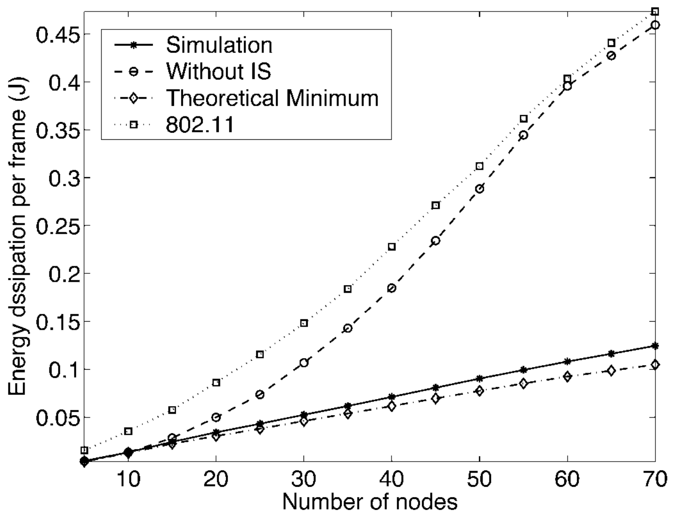


Fig. 4. Average network energy dissipation per frame versus number of nodes.

mechanism; instead only the data packet is transmitted, since no feedback can be obtained from the other nodes, and binary exponential backoff (BEB) is not employed for broadcast traffic [19]. Thus, IEEE 802.11 becomes carrier sense multiple access (CSMA) for broadcast traffic [20]. The throughput of IEEE 802.11 is lower than TRACE due to collisions, which arise because of the lack of coordination among the nodes (i.e., simultaneous transmissions result in collisions and none of the transmitting nodes are aware of the situation).

Fig. 3 shows the average number of dropped packets per frame and R_{PD} as functions of N_N in the upper and lower panels, respectively. R_{PD} increases exponentially for $N_N \geq 40$. In this range, the actual number of nodes that simultaneously have voice packets to send frequently exceeds the number of data slots, so voice packets are dropped since it is not possible to grant permission to all nodes simultaneously.

The normalized capacity η of TRACE reaches 1.76 at $N_N = 44$ $(R_{PD} = 0.01)$, whereas the η of packet reservation multiple access (PRMA) is reported as 1.16 [18]. It is also reported in [18] that at an optimal operating point the η of PRMA reaches 1.64. However, the problem of keeping the network in the optimal operating point is not addressed in [18]. So the η at the optimal case can be thought of as the upper bound for PRMA. There are several factors contributing to the difference between the η's of PRMA and TRACE. The main factor in this difference is that the contention for channel access results in collisions and data slots cannot be used by either of the contenders in PRMA. In TRACE, since contention is not in the data slots, there is no loss of data slots due to contention. In addition, the number of contention slots is higher than the number of data slots, which further reduces the collisions. Another factor is that the T_{drop} of PRMA is 20% lower than that of TRACE.

Channel bit rate used in [15] and [18] for PRMA evaluation is 720 kb/s, which is entirely used by the nodes for uplink communications. The bandwidth used by the controller for downlink communications is not mentioned in [15] and [18]. We used a channel bit rate of 1 Mb/s, which includes both uplink and downlink bandwidth and all the control packets. The bandwidth exclusively used for data transmissions and receptions is 848 kb/s.

E. Energy Dissipation

Fig. 4 shows a plot of the total network energy dissipation per frame for different values of N_N. The theoretical minimum energy is the energy needed to transmit and receive data only. We assume an omniscient network controller takes care of network coordination and informs the nodes without dissipating any energy. The maximum difference between the theoretical minimum and simulation is 19.6 mJ (15.8%) at $N_N = 70$. All the energy above the theoretical minimum energy is spent for control packets and network monitoring.

Energy dissipation without the IS slot is much higher than energy dissipation when the IS slots are used to create listening clusters, because all the nodes should be listening to all data transmissions, forwarding the desired packets to the upper layer and discarding the rest, which results in extra power dissipation for unnecessary but also inevitable information reception in the absence of the IS slot. The maximum difference between the case without the IS slot and with the IS slot is 335 mJ, which corresponds to a 269% increase in energy dissipation. Thus using data summarization slots (IS slots) are very helpful in reducing energy dissipation.

IEEE 802.11 has 52 B of packet header in broadcast packets in standard operation, whereas TRACE has only 4 B of data packet header. In order to compare these two protocols on a fair basis, we reduced the header size for IEEE 802.11 to 4 B, so the data packet size is 104 B for both TRACE and IEEE 802.11 in our simulations. Fig. 4 shows that energy dissipation for IEEE 802.11 is higher than all the other cases for all N_N, because in standard IEEE 802.11 operation all the nodes in the network are always on and all the broadcast packets are received without any discrimination. Maximum difference between TRACE and IEEE 802.11 energy dissipation curves is 349 mJ (281% increase in energy dissipation) at $N_N = 70$. Energy dissipation for IEEE 802.11 is higher than that of TRACE without IS slots because in IEEE 802.11, none of the nodes goes to sleep mode,

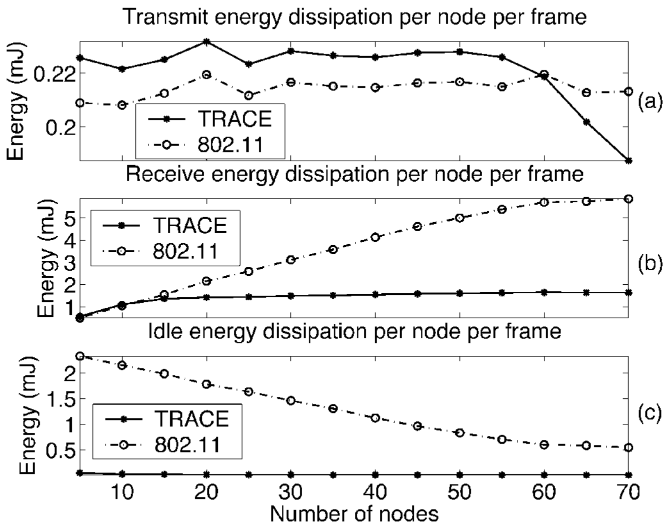


Fig. 5. (a) Transmit energy dissipation per node per frame for TRACE and 802.11. (b) Receive energy dissipation per node per frame for TRACE and 802.11. (c) Idle energy dissipation per node per frame for TRACE and 802.11.

whereas in TRACE without IS slots, nodes go to sleep mode if the network is idle.

Fig. 5(a)–(c) shows the energy dissipation per node per frame in transmit, receive, and idle modes for TRACE and 802.11, respectively. 802.11 has almost constant transmit energy dissipation at all node densities, because all the packets are transmitted in 802.11 without being dropped. Transmit energy of TRACE is almost constant and higher than that of 802.11 for $N_N < 60$, due to additional control packet transmissions. However, for $N_N \geq 60$, due to the dropped packets, transmit energy dissipation of TRACE is lower than that of 802.11. Receive energy dissipation of TRACE is constant for $N_N \geq 15$, after which the average number of transmissions exceeds the maximum listening cluster size. 802.11 receive energy increases linearly with node density until $N_N = 60$, and stays constant for $N_N \geq 60$. Idle energy dissipation of TRACE is almost zero for all node densities. 802.11 idle energy dissipation decreases with increasing node density, because idle time is decreasing with increasing node density.

Total energy dissipation per node per frame for TRACE and 802.11 at $N_N = 5$ are 0.83 and 3.19 mJ, respectively. The ratios of transmit, receive, and idle energy dissipation at $N_N = 5$ for TRACE and 802.11 are 1.0/2.46/0.22 and 1.0/2.39/11.17, respectively. Energy dissipation of TRACE and 802.11 for packet transmission and reception are almost the same, because the listening cluster ($N_{\max} = 5$) does not save any energy at this node density for TRACE. Most of the extra energy dissipation for 802.11 when compared with TRACE is due to the idle mode energy dissipation, which constitutes 73% of the total energy dissipation. At $N_N = 70$, the per node per frame energy dissipation for TRACE and 802.11 are 1.83 and 6.96 mJ, respectively. The ratios of transmit, receive, and idle energy dissipation at $N_N = 70$ for TRACE and 802.11 are 1.0/8.7052/0.0335 and 1.0/27.5166/2.5537, respectively. The difference between TRACE and 802.11 is mostly due to the listening cluster based power saving mechanism of TRACE, because most of the energy dissipation of 802.11 (i.e., 85% of total energy dissipation) is due to the packet receptions at this node density.

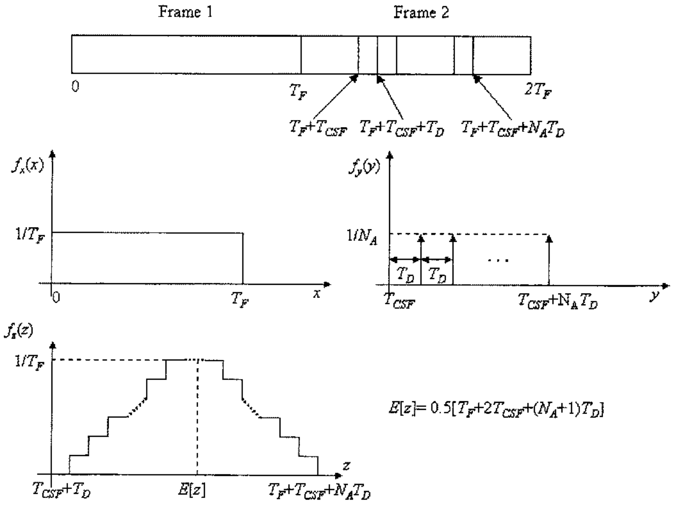


Fig. 6. Packet delay calculations. The top row displays the frame structure used for packet delay analysis. The pdf's of x, y, and z are plotted in middle and bottom rows.

Energy dissipation is a function of data traffic, which is directly proportional to the number of nodes. For lower node densities, the dominant factor in energy dissipation for 802.11 is idle listening. Thus, if the idle power and sleep power are very close in an energy model, then the energy dissipation for TRACE and 802.11 will be very close in a low density network. If the node density is high, then the dominant term in energy dissipation for 802.11 is the receive power and the contribution of idle mode energy dissipation becomes marginal.

F. Packet Delay

The arrival time of a voice packet is uniformly distributed to one frame time. It is not possible for a packet to arrive and be delivered in the same frame; the earliest delivery can be in the next frame. The delivery time is a uniform discrete random variable, because packets can be delivered only at the end of each data slot, and no data slot has precedence over others.

Random variables x and y, which are shown in Fig. 6, represent the packet arrival time and the packet delivery time, respectively. The probability density function (pdf) of x the packet arrival time is given as

$$f_x(x) = \begin{cases} \frac{1}{T_F}, & 0 < x \leq T_F \\ 0, & \text{otherwise} \end{cases}. \tag{4}$$

The pdf of the delivery time y is

$$f_y(y) = \frac{1}{N_A} \sum_{k=1}^{N_A} \delta(y - T_{\text{CSF}} + kT_D) \tag{5}$$

where T_{CSF} is the control subframe duration, and $\delta(.)$ is the Dirac-delta function.

We can find the delay by subtracting x from y, but we must add an offset of T_F to y in order to define both variables ac-

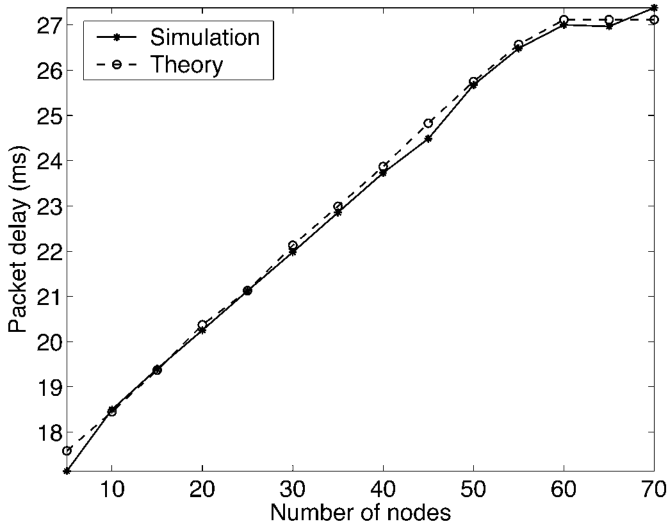


Fig. 7. Packet delay versus number of nodes.

cording to beginning of frame 1 (i.e., $y = 0$ corresponds to $y = T_F$). The delay is given by

$$z = T_F + y - x. \tag{6}$$

Since x is a uniform random variable between 0 and T_F, $T_F - x$ is equivalent to x, so

$$z = y + x. \tag{7}$$

The pdf of z is obtained by convolving the pdfs of x and y

$$f_z(z) = f_x(x) \otimes f_y(y) \tag{8}$$

$$\begin{aligned} f_z(z) = {} & \frac{1}{N_A T_F} \sum_{k=1}^{N_A} \\ & \times \{u\left(z - (T_{\text{CSF}} + kT_D)\right) \\ & \quad u\left(z - (T_F + T_{\text{CSF}} + (N_A + 1 - k)T_D)\right)\} \end{aligned} \tag{9}$$

where $u(.)$ denotes the unit step function. The expected value of z is obtained as

$$E[z] = 0.5\left(T_F + 2T_{\text{CSF}} + (N_A + 1)T_D\right). \tag{10}$$

Fig. 7 shows a plot of the average packet delay versus the number of nodes. The maximum difference between the simulation data and theory is 0.26 ms at $N_N = 70$, which corresponds to a 1.0% difference.

G. Node Failure

To test the automatic controller backup scheme, we designed a random controller failure simulation. In the simulation the controller can fail with a probability p at each frame. This corresponds to an exponentially decreasing nonfailure probability in time, which is shown to be a valid model for wireless radios [21]. Let u be the random variable that represents the nonfailure for the controller at the kth beacon transmission and define $q = 1 - p$ to be the probability of nonfailure. The pdf of u is

$$f_u(k) = \left(\frac{1-q}{q}\right) q^k. \tag{11}$$

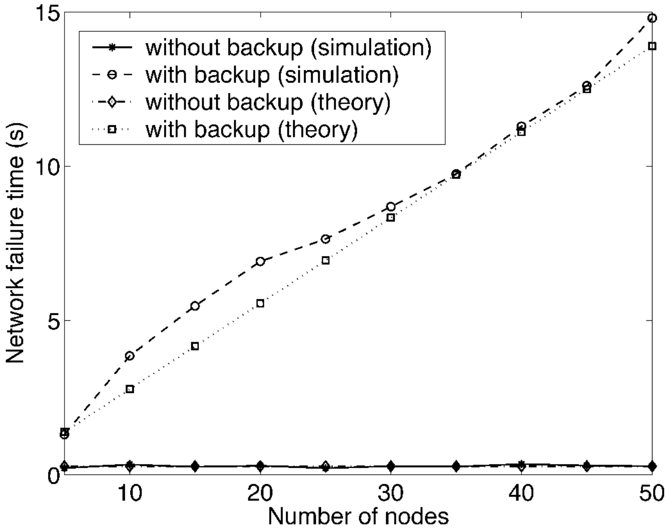


Fig. 8. Network failure time versus number of nodes.

The first term is the normalization term to make the area of the pdf unity; the second term states that the probability of nonfailure decreases exponentially. The expected value of u is

$$\mu = \left(\frac{1-q}{1}\right) \sum_{k=0}^{\infty} kq^k. \tag{12}$$

This gives the average lifetime (i.e., failure time) of a network without any backup mechanism and with a controller nonfailure probability of q. The expected lifetime of a network having a backup mechanism with N nodes μ_N is given by

$$\mu_N = N\mu. \tag{13}$$

Network lifetime curves obtained from simulations and theory with $\mathrm{p} = 0.1$ are plotted in Fig. 8. Simulations are averaged over ten statistically independent simulation runs. The average network lifetime without backup is 0.2824 and 0.2778 s for the simulation and theory, respectively. The average network lifetime with backup elongates the network failure time directly proportional with the number of nodes in the network. Network lifetime increases 50 times for a 50-node network theoretically. The increase in network lifetime in the simulations is 52.4, on the average for a 50-node network.

IV. DISCUSSION

In the simulations, we assumed that all the nodes in the network are active voice sources and independent of each other to demonstrate the worst-case performance of TRACE; however, it is unlikely in a realistic scenario that everybody is speaking without listening to others. Therefore, it is possible to support a higher number of nodes with the same packet drop rate in a realistic scenario. Energy dissipation per node will also be lower if not all the nodes are active. There will not be any change in packet delay characteristics, because silent nodes are just passive participants in the network.

We consider the possibility of saving more energy by using a multihop approach, but it turns out that since the dominant term in our radio model is the energy dissipation on radio electronics, we cannot save any power by a multihop approach with the radio model and coverage area we are using.

Capture is a factor that affects the fairness of PRMA and all other ALOHA family protocols. Indeed, a strong capture mechanism increases the throughput of PRMA, because of the fact that most of the contention attempts result in favor of the node close to the base station. Instead of loosing both packets and wasting the whole data slot, only one of the nodes looses the contention and the other captures the channel, which increases the total throughput and degrades the fairness among the nodes in an uncontrolled manner (i.e., unlike the prioritization in TRACE, which is a controllable design parameter). The effects of capture in TRACE are only marginal.

The IS slot contributes significantly to the energy efficiency of TRACE. The end-of-stream information is included in the IS slot, because it is the most appropriate point in the frame structure for this information. A node does not know whether it has a voice packet or not in the next frame during its data transmission because the packet generation rate is matched to the frame rate, so end-of-stream information cannot be sent in the data slot. The earliest point where a node knows it is out of packets is during the control subframe. If the end-of-stream information is not sent in the IS slot but in the data slot (i.e., no data is sent to indicate the end-of-stream like in PRMA), then the controller should be listening to all the data slots to monitor for the continued use of data slots, which results in waste of considerable energy.

In our current implementation, the information for data discrimination is proximity; however, the information in the IS slot can be modified for different applications. For example, the IS slot can be used to send metadata describing the data that will be transmitted in the corresponding data slot. The nodes can choose which transmitters to listen to based on this metadata. An efficient way of using metadata prior to data transmission in a multihop sensor network application is presented in [11].

Priority levels of TRACE might be used to support various requirements of the applications using TRACE as the MAC layer. For example, in a military operation, it is necessary that the commander has priority over other soldiers and everybody listens to the commander's speech ($PL1$), and the leaders of each subsquad should also have a priority lower than that of the commander ($PL1$) but higher than the others ($PL3$). In a multimedia application $PL1$ and $PL2$ could be thought of as constant bit rate (CBR) sources and PL3 as a variable bit rate (VBR) or available bit rate (ABR) source. In a field trip, the tour guide can be a $PL1$ node and the rest can be $PL3$ nodes.

TRACE does not have a global synchronization requirement. Each node updates the frame start time by listening to the beacon sent by the controller, and all the transmissions and receptions are defined with respect to this time, which is updated at each frame by the controller.

V. RELATED WORK

Continuation of data slot reservation for an uninterrupted sequence of voice packets is the key feature that makes TRACE a real-time communication protocol that guarantees bounded delay for voice packets. Reservation ALOHA (R-ALOHA), originally proposed for satellite communications, was the first protocol that employed the idea of slot reservation [2], [22], [23]. R-ALOHA is a combination of slotted ALOHA (S-ALOHA) and TDMA. In R-ALOHA, time is organized into frames, and frames are divided into slots. The frame structure of R-ALOHA is inherited from TDMA. Successful data transmission in a slot automatically reserves the corresponding slot for the transmitting node in the next frame. By repeated use of that slot position, a node can transmit a long stream of data. Any unreserved slot is available for the next frame; nodes may contend for that slot using S-ALOHA. Thus, in R-ALOHA, contention is on data slots and collisions corrupt (possibly long) data packets. All the nodes in the network should be on all the time in order to monitor the status of each slot. If there is a packet transmission, all the nodes receive it and discard it if it is not destined for them. Inherently it is not possible to save power with R-ALOHA. Fairness and prioritization are also not addressed by R-ALOHA.

Voice activity detection improves the throughput of TRACE substantially. Voice activity detection in multiple access was first used in PRMA [15], [18]. The main goal of PRMA, which is closely related with R-ALOHA, is to support real-time voice traffic and use the remaining bandwidth for asynchronous data transmissions. PRMA is distinguished from R-ALOHA by its response to network congestion and use of voice activity detection. In PRMA, information packets from periodic sources, such as speech, are discarded if they remain in the node beyond a certain time limit. Voice activity detection increases the capacity of the radio channel significantly due to the discontinuous nature of speech (i.e., no packets are generated when there is no voice signal). PRMA is designed to operate in a star topology, where the base station is in the center and the wireless nodes are dispersed around it. No direct communication is supported; even if the nodes are in the same cluster, they should be communicating via the base station (i.e., the same operation principle as Bluetooth). Energy efficiency and support for broadcast was also not among the design considerations of PRMA.

Stability is an important issue, which determines the system performance for R-ALOHA and PRMA [24]. If the number of nodes contending for the same slot is too high, then none of the contending nodes can capture the data slot because of collisions. Therefore, both throughput and delay suffer severely. In order to sustain the system stability, the number of contending nodes and available data slots should be estimated and system parameters should be updated accordingly [18], [24]. TRACE is virtually immune to stability problems because the contention is not in the data slots but in contention subslots. The natural isolation between the contention-free data subframe and the contention subslots makes TRACE highly stable and robust.

A comparison of an early version of TRACE, PBP (an enhanced version of IEEE 802.11 for single-hop networks) and ASP (an energy efficient polling protocol for Bluetooth) in a sensor network application for a many-to-one data transmission model is given in [25]. It is shown that the energy dissipation of TRACE is much less than PBP for the same number of data transmissions. PBP is shown to be not very energy efficient when compared with TRACE because of the lack of central coordination and high overhead.

Bluetooth networks are not capable of supporting large numbers of nodes due to the limited piconet size, which is eight [17]. Although scatternet creation, which is an option to extend Bluetooth networks, allows larger networks, it is not clear how to create an efficient scatternet. Bluetooth's operation principle is based on information conveyed through the piconet controller, which eliminates the possibility of direct peer-to-peer communication. Therefore, Bluetooth is not a good choice for the application scenarios we targeted for TRACE.

VI. CONCLUSION

In this paper, we presented TRACE, a TDMA-based MAC protocol for energy efficient real-time packetized voice broadcasting in a single-hop radio network. Two features of TRACE make it an energy efficient protocol: 1) scheduling and 2) receiver- based listening cluster creation via information summarization slots. Network lifetime is maximized in TRACE using dynamic controller switching and automatic backup mechanisms. Separation of the contention and data transmission is the determining factor in high throughput, low delay and stability under a very wide range of data traffic. Different QoS levels are also supported in TRACE via priority levels.

All of the above features are quantified through simulations and analytical models. It is shown that TRACE has better energy saving and throughput performance than PRMA and IEEE 802.11.

Our future research will concentrate on extending TRACE to multihop networks and for heterogeneous traffic, such as data along with voice, which might necessitate reservation of more than one data slot per node per frame.

ACKNOWLEDGMENT

The authors would like to thank D. Stephenson and S. Elvy of the RF Communications Division, Harris Corporation, Rochester, NY, for their feedback on this work. The authors would like to thank the reviewers for their helpful comments.

REFERENCES

[1] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *J. Wireless Commun. Mobile Comp.*, vol. 2, pp. 483–502, 2002.

[2] K. Pahlavan and A. H. Levesque, *Wireless Information Networks*. New York: Wiley, 1995.

[3] A. S. Tanenbaum, *Computer Networks*. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[4] T. S. Rappaport, *Wireless Communications*. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[5] L. L. Peterson and B. S. Davie, *Computer Networks*. New York: Academic, 2000.

[6] *The IEEE 802.11 Handbook: A Designer's Companion*, IEEE Press, NJ, 1999.

[7] Bluetooth Project (1999). *http://www.bluetooth.com* [Online]

[8] V. Kanodia, C. Li, A. Sabharwal, B. Sadaghi, and E. Knightly, "Distributed multi-hop scheduling and medim access with delay and throughput constraints," in *Proc. ACM/IEEE MOBICOM*, 2001, pp. 200–209.

[9] J.-C. Chen, K. M. Sivalingam, P. Agrawal, and S. Kishore, "A comparison of MAC protocols for wireless local networks based on battery power consumption," in *Proc. IEEE INFOCOM*, vol. 1, 1998, pp. 150–157.

[10] S. Singh and C. S. Raghavendra, "PAMAS: power aware multi-access protocol with signaling for ad hoc networks," *ACM Comput. Commun. Rev.*, vol. 28, pp. 5–26, 1998.

[11] W. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *Proc. ACM/IEEE MOBICOM*, 1999, pp. 174–185.

[12] M. Budagavi, W. R. Heinzelman, J. Webb, and R. Talluri, "Wireless MPEG-4 video communication on DSP chips," *IEEE Signal Processing Mag.*, vol. 17, pp. 36–53, 2000.

[13] S. Ni, Y. Tseng, and J. Shen, "The broadcast storm problem in a mobile ad hoc network," in *Proc. ACM/IEEE MOBICOM*, 1999, pp. 151–162.

[14] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An application specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wireless Commun.*, vol. 1, pp. 660–670, Oct. 2002.

[15] D. Goodman, R. Valenzuela, K. Gayliard, and B. Ramamurthi, "Packet reservation multiple access for local wireless communications," *IEEE Trans. Commun.*, vol. 37, pp. 885–890, Aug. 1989.

[16] Network Simulator (ns) [Online]. Available: http://www.isi.edu/nsnam/ns

[17] J.-F. Frigon, V. C. M. Leung, and H. C. B. Chan, "Dynamic reservation TDMA protocol for wireless ATM networks," *IEEE J. Select. Areas Commun.*, vol. 19, pp. 370–383, Feb. 2001.

[18] D. J. Goodman and S. W. Wei, "Efficiency of packet reservation multiple access," *IEEE Trans. Veh. Technol.*, vol. 40, pp. 170–176, Feb. 1991.

[19] B. Williams and T. Camp, "Comparison of broadcasting techniques for mobile ad hoc networks," in *Proc. ACM MOBIHOC*, 2002, pp. 194–205.

[20] K. Tang and M. Gerla, "Mac layer broadcast support in 802.11 wireless networks," in *Proc. IEEE MILCOM*, 2000, pp. 544–548.

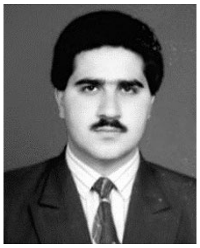
[21] G. Hoblos, M. Staroswlecki, and A. Aitouche, "Optimal design of fault tolerant sensor networks," in *Proc. IEEE Int. Conf. Control Applications*, 2000, pp. 467–472.

[22] W. R. Crowther *et al.*, "A system for broadcast communications: reservation ALOHA," presented at the *6th Hawaii Int. System Sciences Conf.*, 1973.

[23] S. S. Lam, "Packet broadcast networks—a performance analysis of the R-ALOHA protocol," *IEEE Trans. Commun.*, vol. 29, pp. 596–603, 1980.

[24] S. Tasaka, "Stability and performance of the R-ALOHA packet broadcast system," *IEEE Trans. Comput.*, vol. 32, pp. 717–726, 1983.

[25] Z. Cheng, M. Perillo, B. Tavli, W. Heinzelman, S. Tilak, and N. B. Abu-Ghazaleh, "Protocols for local data delivery in wireless microsensor networks," in *Proc. 45th IEEE Midwest Symp. Circuits Systems*, vol. 1, 2002, pp. 623–626.



Bulent Tavli (S'97) received the B.S. degree in electrical and electronics engineering from the Middle East Technical University, Ankara, Turkey, in 1996, and the M.S. degree in electrical and computer engineering from the University of Rochester, Rochester, NY, in 2001, where he is currently working toward the Ph.D. degree.

Currently, he is a Research Assistant in the Electrical and Computer Engineering Department, University of Rochester. His current research interests lie in the area of low-power ad hoc wireless protocol architectures.

Wendi B. Heinzelman (S'95–M'01) received the B.S. degree in electrical engineering from Cornell University, Ithaca, NY, in 1995, and the M.S. and Ph.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology (MIT), Cambridge, in 1997 and 2000, respectively.

She is an Assistant Professor in the Department of Electrical and Computer Engineering, University of Rochester, Rochester, NY. Her current research interests lie in the area of wireless communications and networking, mobile computing, and multimedia communication.

Dr. Heinzelman is an Elected Member of the Design and Implementation of Signal Processing Systems (DISPS) Technical Committee of the Signal Processing Society and a Member of Sigma Xi and the Association for Computing Machinery (ACM).