

University of Nebraska - Lincoln

DigitalCommons@University of Nebraska - Lincoln

U.S. Navy Research

U.S. Department of Defense

9-30-2009

Energy-efficient, Flow-specific Medium Access using Preamble Sampling

T. Owens Walker III

Naval Postgraduate School, Monterey, California, USA, towalker@nps.edu

Murali Tummala

Naval Postgraduate School, Monterey, California, USA, mtummala@nps.edu

John McEachen

Naval Postgraduate School, Monterey, California, USA, mceachen@nps.edu

Follow this and additional works at: <http://digitalcommons.unl.edu/usnavyresearch>



Part of the [Operations Research, Systems Engineering and Industrial Engineering Commons](#)

Owens Walker III, T.; Tummala, Murali; and McEachen, John, "Energy-efficient, Flow-specific Medium Access using Preamble Sampling" (2009). *U.S. Navy Research*. 4.

<http://digitalcommons.unl.edu/usnavyresearch/4>

This Article is brought to you for free and open access by the U.S. Department of Defense at DigitalCommons@University of Nebraska - Lincoln. It has been accepted for inclusion in U.S. Navy Research by an authorized administrator of DigitalCommons@University of Nebraska - Lincoln.

Energy-efficient, Flow-specific Medium Access using Preamble Sampling

T. Owens Walker III, Murali Tummala, and John McEachen

Department of Electrical and Computer Engineering
Naval Postgraduate School, Monterey, California, USA
{towalker, mtummala, mceachen}@nps.edu

Abstract—Current wireless sensor network applications face challenging power constraints that demand low duty cycles. In this paper, we propose an energy-efficient, flow-specific medium access scheme that is capable of achieving low duty cycles while taking advantage of the performance improvements available through a flow-specific medium access approach. The scheme is based on an adaptive sleep and wake cycle that responds to changes in both contention and non-contention traffic loads through the use of preamble sampling. A preamble sampling probability parameter is introduced to manage the trade-off between energy efficiency and network throughput and delay performance. A performance analysis is conducted and verified through simulation.

Keywords—wireless; medium access; flow-specific; energy efficient; preamble sampling; cross-layer.

I. INTRODUCTION

Recent work has identified the delay and throughput performance advantages of a flow-specific medium access solution for wireless networks [1],[2]. The traffic-adaptive Cooperative Wireless Sensor Medium Access (CWS-MAC) protocol [1] has been proposed to provide either contention or non-contention medium access service on a per flow basis. This approach has been shown to outperform contention [3], non-contention [4] and hybrid [5] medium access solutions for both traditional [1] and large-scale wireless sensor networks [2].

Military applications of wireless sensor networks such as unattended battlefield monitoring [6] and enemy signals collection [7] tend to be extremely power-constrained because the remote and hostile environment prevents both node and battery replacement [6] and the very small form factor often prohibits the use of renewable energy sources [6]. To satisfy these demanding energy constraints, battlefield sensor networks must feature low duty cycles.

The major contribution of this paper is an energy-efficient, flow-specific medium access scheme that is capable of achieving the low duty cycles required in current power-constrained wireless sensor applications while providing the throughput and delay performance advantages of a flow-specific approach. We accomplish this by incorporating an adaptive sleep and wake cycle into the traffic-adaptive CWS-MAC protocol [1] using preamble sampling. This adaptive sleep and wake cycle responds to changes in traffic load and

we introduce a preamble sampling probability parameter that is capable of managing the trade-off between energy efficiency and throughput and delay performance.

This paper is organized as follows. We begin by discussing current work in energy-efficient wireless medium access in Section II. In Section III, we propose our energy-efficient, flow-specific medium access scheme. We provide performance analysis in Section IV and simulation results in Section V.

II. ENERGY-EFFICIENT WIRELESS MEDIUM ACCESS

Energy efficiency is a significant challenge to medium access solutions in both mobile ad hoc networks and wireless sensor networks due to the limited battery power available [8]. This is particularly exacerbated in the latter where it may be impractical to change out or recharge the sensor node batteries [8]. Researchers have identified five major sources of energy waste in wireless medium access schemes [8],[9]:

1. Packets must be discarded and retransmitted when they experience *collisions*. The retransmissions result in an increase in both power consumption and latency.
2. *Overhearing* occurs when nodes receive packets for which they are not the intended receiver.
3. *Control packet overhead* stems from the use of dedicated control packets to coordinate transmissions.
4. *Idle listening* occurs when nodes listen for packets while the channel is idle.
5. *Overemitting* occurs when a message is transmitted to a destination that is not ready to receive it.

IEEE 802.11 [10] includes a mechanism to transition a node into a reduced power “sleep state” called the power save (PS) mode. In this mode, the transceiver is powered down and a node can neither transmit nor receive. In an 802.11 infrastructure network, the PS mode requires the presence of an access point (AP) that is always on and coordinates traffic for the mobile host. The mobile host informs the AP when it is powering down and the AP then buffers packets for the host. The mobile host periodically wakes up to check if the AP has buffered packets for it. In an ad hoc network, each mobile host in the PS mode wakes up periodically during designated ad hoc traffic indication map (ATIM) windows. Within these ATIM windows, nodes transmit ATIM messages (using the

distributed coordination function (DCF) mechanism) which indicate intended receivers for buffered packets. If a host in the PS mode is not on the list of intended receivers, it will power down until the next ATIM window. IEEE 802.11 is designed for single-hop (or fully connected networks where all nodes can “hear” each other) and synchronization, neighbor discovery and network partitioning present problems when this scheme is applied to multi-hop ad hoc wireless networks [11]. The authors of [11] present several proposals to address these issues, but the control overhead and latency of their proposed solutions can be large because they do not coordinate the sleep periods of the nodes [9]. An alternate approach, the Power Aware Multi-access protocol with Signaling (PAMAS) [12] proposes a second signaling channel and takes advantage of the RTS/CTS exchange to power nodes down if they are not the intended receiver of the upcoming transmission. In all of these protocols, the contention resolution mechanism of reduces the energy waste due to collisions and the sleep mechanism limits the overhearing cost.

While the protocols discussed above reduce the energy consumption due to overhearing, they do not address the problem of idle listening. In an ideal solution to this problem, a node will only wakeup when it is the destination for the upcoming packet transmission. The challenge of low duty cycle operation has been approached through both asynchronous and synchronous techniques.

There are two common asynchronous approaches to sleep implementation. The first is a hardware solution that makes use of a secondary, low-power “wake-up radio” [13] while the second is an algorithmic solution that is referred to as either *preamble sampling* or *low power listening* [14]. The requirement for a second radio can be problematic for wireless sensor networks, so we turn our attention instead to the latter approach. In preamble sampling, a node will periodically wakeup and listen to the channel to see if it has traffic pending. Thus, a node with a packet to transmit need only transmit a beacon for the duration of the sampling cycle to wake-up the destination node. Upon waking up and hearing the beacon, a node will then remain awake to receive the subsequent transmission. The beacon can simply be a physical layer RF pulse which is easy to implement, but this has the undesirable side effect that all nodes in the reception range will remain awake and wait for the subsequent transmission to determine whether or not they are the intended destination. At higher loads, this can result in a substantially higher duty cycle. Alternately, the beacon can be a MAC layer mechanism that includes destination information, but this requires a more complex implementation. WiseMAC [15] reduces the requirement for the relatively long preamble transmissions by allowing neighboring nodes to exchange preamble sampling times. The preamble duration is then a function of the accuracy of the synchronization between the two nodes (bounded by the preamble sampling period). B-MAC [16], a commonly used reconfigurable MAC protocol that has been implemented on the Berkeley family of motes, includes preamble sampling in the suite of functions it provides.

By synchronizing sleep schedules, a family of protocols [9],[17],[18] have been proposed that further reduce the energy consumption due to idle listening. S-MAC [9] addresses the

idle listening problem and attempts to improve control overhead and latency by coordinating the sleep periods of neighboring nodes. This is accomplished by broadcasting sleep schedules among neighbors. Neighbors then form virtual clusters by aligning their sleep schedules. The result is a set of coordinated, fixed length, sleep (or duty) cycles that are comprised of alternating periods of listening/transmitting and sleeping. T-MAC [17] improves upon the energy-efficiency of this scheme by allowing the sleep cycle to be adaptive through the use of an inactivity time-out mechanism. D-MAC [18] takes advantage of the data-gathering tree structure in many wireless sensor networks to coordinate the sleep schedules and reduce the latency introduced by the sleep cycles.

Scheduled access using contention-free, TDMA approaches eliminates the energy cost due to collisions and also allows nodes to sleep when they are neither transmitting or receiving in a given time slot. To realize these latter energy savings, nodes must be able to determine if they are the intended receiver in the transmission scheduled in the given slot. This can be accomplished through the use of sender and receiving scheduling [4] or preamble sampling [5] in which nodes wakeup at the beginning of each slot and check to see if they are the intended receiver.

In this paper, we propose a single channel, energy-efficient solution that is capable of supporting the per flow medium access service present in flow-specific medium access approaches. We use preamble sampling to minimize overhearing, idle listening, and overemitting. Collisions are reduced by switching from a contention-based medium access to a scheduled approach as the per flow load increases and we minimize control packet overhead by proposing a distributed scheme in which nodes are capable of making local medium access decisions without the requirement for a centralized controller.

III. PROPOSED ENERGY-EFFICIENT, FLOW-SPECIFIC MEDIUM ACCESS SCHEME

Our proposed energy-efficient flow-specific medium access control scheme is based on the traffic-adaptive Cooperative Wireless Sensor Medium Access Control (CWS-MAC) protocol [1]. We modify the work of [1] by adding a sleep state during which sensor nodes power down their transceiver and processor to reduce energy consumption. Individual nodes participate in a sleep and wake cycle that is coordinated through the use of preamble sampling. This allows individual nodes to sleep when they are not designated receiver(s) during the current transmission. In this section, we present the frame structure and discuss the operation of the proposed scheme.

A. Frame structure

The frame structure of our proposed energy-efficient traffic-adaptive CWS-MAC protocol is shown in Figure 1. The non-contention and contention modes of this flow-specific protocol are implemented as in [1]. The underlying non-contention scheme is developed around a TDMA frame structure while the contention mode is based on a slotted ALOHA approach. The contention mode is overlaid on top of the frame structure through the use of a contention beacon of

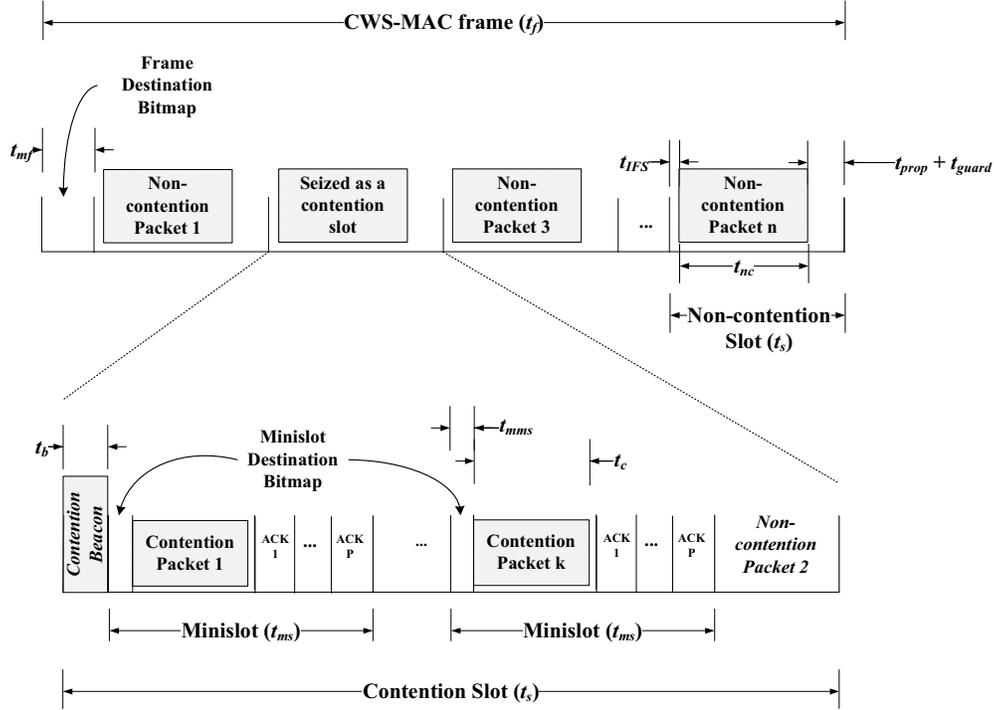


Figure 1. Frame structure of the proposed energy-efficient, traffic-adaptive CWS-MAC protocol.

duration t_b . Prior to transmitting their non-contention packets, slot owners sample for this beacon at a time t_{IFS} (relative to the slot boundary). Upon detection of the beacon, slot owners defer and the non-contention slot is effectively “seized” as a contention slot. Further details on flow-specific medium access and the operation of traffic-adaptive CWS-MAC can be found in [1]. We now turn our attention to the modifications we have made to implement the preamble sampling-based sleep and wake cycle.

A frame destination bitmap is included at the beginning of each frame as shown in Figure 2. This map is used to identify the designated receivers in each transmission slot. This destination bitmap is subdivided into slots which correspond to the slots assigned in the transmission frame. Each slot in the destination bitmap is further subdivided into another level of slots (again corresponding to the slot structure of the transmission frame) which represent the individual slot transmission maps. The first level corresponds to the transmission slot while the second level identifies the designated receiver(s) in that transmission slot. The frame destination bitmap is of duration

$$t_{mf} \geq \frac{n^2}{R} \quad (1)$$

where n is the number of slots in the transmission frame and R is the channel data rate. This is a lower bound because the size may be increased to provide both error correction to protect the destination map field and guard bands based on the fidelity of the slot synchronization scheme utilized.

A minislot destination bitmap is also included at the beginning of each minislot within the contention slot as shown

in Figure 3. This map is used to designate the receiver(s) for the subsequent contention packet transmission in that minislot. Again using a bitmap approach, it is comprised of a series of bits corresponding to each slot in the transmission frame. These bits are used to designate the intended receiver(s). The minislot destination bitmap is of duration

$$t_{mms} \geq \frac{n}{R}. \quad (2)$$

While guard bands are typically not needed in this instance, (2) remains a lower bound due to the potential use of an error correction scheme.

The destination bitmap approach takes advantage of the existing slot structure used to support the non-contention mode of the protocol. The slot assignment process can be either distributed or centralized and the results must be disseminated to all nodes identified within the transmission frame.

B. Operation

In this section, we discuss the operation of our proposed energy-efficient scheme. We begin with the frame destination bitmap and then outline operation in both the non-contention and contention modes. Nodes wake up at the beginning of each transmission frame to receive the frame destination map. This map is created by the individual nodes who broadcast their transmission slot maps at the appropriate time within the frame destination bitmap. For a given node, this individual slot map identifies the receiver(s) for the upcoming transmission in the node’s slot within the current transmission frame. A bit value of one indicates that a specific node is an intended receiver for the transmission while a bit value of zero indicates that it is not. A value of all zeroes in the slot map indicates that the owner of

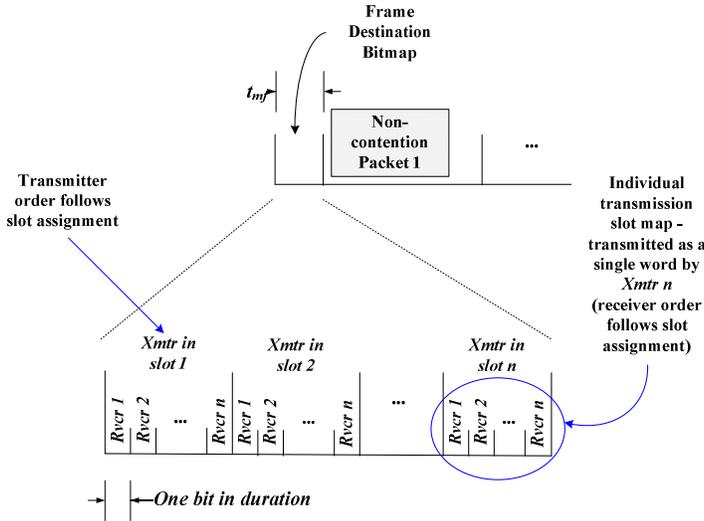


Figure 2. Frame destination bitmap.

that slot does not have any non-contention traffic to transmit. The order of the bits corresponds to the slot assignment within the transmission frame. If a node is assigned to transmit in slot j within the frame, then bit j within the slot map is used to identify it as a potential receiver in that transmission slot. Note that a non-contention packet must arrive prior to the frame to be included in the frame destination bitmap and, hence, be eligible for transmission in the current frame.

Once the frame destination map has been received and processed, nodes will know in which transmission slots they are designated as an intended receiver. In the non-contention mode, nodes enter the sleep state for the duration of transmission slots in which they are neither an intended receiver or the slot owner with traffic to transmit.

To accommodate the contention mode, nodes will wake up with some probability p_s to sample the channel in each transmission slot at time t_{IFS} (relative to the slot boundary) for the presence of a contention beacon. When a contention beacon is detected, the slot is redesignated by the node as a contention slot. A value of $p_s = 1$ represents the case where the node samples every transmission slot. A value of $p_s = 0$ indicates that a node will only detect contention slots that occur when the node is either an intended receiver of the scheduled non-contention packet transmission or it is the slot owner and has non-contention traffic to transmit.

During a contention slot, a node will wakeup at the beginning of each minislot to receive the minislot destination bitmap to determine whether or not it is an intended receiver for the subsequent contention traffic transmission in that minislot. This minislot destination bitmap is broadcast by a node that intends to attempt transmission in that minislot. A node will stay awake for the subsequent contention packet transmission if it is designated as an intended receiver. A node is allowed to enter the sleep state until the next minislot if it is not the intended receiver or if the bitmap is unreadable as a result of a collision due to multiple attempted transmissions in the minislot. Upon termination of the final minislot, a node will

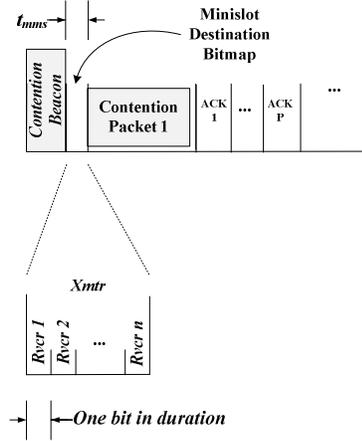


Figure 3. Minislot destination bitmap.

remain in the sleep state unless it was the designated receiver (or transmitter) of non-contention packets in the original non-contention slot that was “seized.” If this is the case, the node will wake up to receive (or transmit) non-contention packets during the non-contention period reserved at the end of the contention slot.

The use of the frame destination bitmap ensures that a node will always be awake for the transmission of a non-contention packet for which it is the intended receiver. Similarly, the use of the minislot destination bitmap ensures that a node will always be awake for the transmission of a contention packet for which it is the intended receiver provided that it has detected the contention slot. The probability that a node detects a contention slot is a function of the probability p_s that it will sample for the contention beacon. If a node fails to detect a contention slot, the node will sleep through the slot and will not receive any contention packets transmitted during that slot for which it is the intended receiver. The same acknowledgement mechanism that recovers from collisions within the contention slot can also be used to recover from packet losses due to missed contention slots.

IV. PERFORMANCE ANALYSIS

In this section, we analyze the duty cycle performance and the throughput and delay of the proposed energy-efficient, flow-specific scheme. We also examine the effect of the sampling probability p_s on the performance of the protocol and its role in managing the tradeoff between energy consumption and throughput and delay performance. For the analysis of this section, we assume packet arrivals are Poisson-distributed and, without a loss of generality, that each node is assigned a single slot in the transmission frame.

A. Duty cycle analysis

We begin our duty cycle analysis by defining the duty cycle, Π , as the ratio of the mean time a node spends in the wake state during a frame, t_{awake} , to the frame time as in

$$\Pi = \frac{t_{awake}}{t_f}. \quad (3)$$

The higher the duty cycle, the more time a node stays awake and, consequently, the greater the energy consumption.

At a minimum, a node will wake up to receive the frame destination bitmap and, hence, the minimum achievable duty cycle is

$$\Pi_{\min} = \frac{t_{mf}}{t_f}. \quad (4)$$

In general, the mean duty cycle is a function of the probability that a node will wake up for a slot, the amount of time it will be awake in the slot, t_w , and the number of slots in a transmission frame as in

$$\bar{\Pi} = \frac{t_{mf} + nt_w \times \Pr[\text{node will be awake for slot}]}{t_f}. \quad (5)$$

A slot can either be a non-contention slot or a contention slot, so we can expand (5) to

$$\begin{aligned} \bar{\Pi} = & \frac{t_{mf}}{t_f} + n \left\{ \frac{t_{wnc}}{t_f} \times \Pr[\text{non-cont slot}] \times \right. \\ & \Pr[\text{awake for non-cont slot}] \\ & \left. + \frac{t_{wc}}{t_f} \times \Pr[\text{cont slot}] \times \Pr[\text{awake for cont slot}] \right\} \end{aligned} \quad (6)$$

where t_{wnc} and t_{wc} are the amount of time a node will be awake in a non-contention and contention slot, respectively. From the analysis provided in [19],

$$\begin{aligned} \Pr[\text{non-cont slot with } p_s = 1] & \equiv p_0 = e^{-\Lambda_c t_s} \text{ and} \\ \Pr[\text{cont slot with } p_s = 1] & = 1 - p_0 = 1 - e^{-\Lambda_c t_s} \end{aligned} \quad (7)$$

where Λ_c is the aggregate contention packet arrival rate and t_s is the slot size. To account for a non-zero value for p_s , we must account for the probability that a node will miss a contention slot for which it is the designated destination of a contention packet transmission. The probability of a contention slot is then

$$\begin{aligned} \Pr[\text{cont slot}] = & \Pr[\text{cont pkt arvl in prev slot}] \\ & + \Pr[\text{cont pkt arvl 2 slots prior}] \\ & \quad \times \Pr[\text{no cont pkt arvl in prev slot}] \\ & \quad \times \Pr[\text{a node misses 1 cont slot}] \\ & + \Pr[\text{cont pkt arvl 3 slots prior}] \\ & \quad \times \Pr[\text{no cont pkt arvl in prev 2 slots}] \\ & \quad \times \Pr[\text{a node misses 2 cont slot}] \\ & + \dots \end{aligned} \quad (8)$$

The probability of a contention packet arrival in a slot is the same for all slots and is found in [19] to be p_0 . The probability of a non-contention packet arrival in a slot is also the same for all slots and is given by $(1-p_0)$. Substituting into (8) and rearranging, we have

$$\begin{aligned} \Pr[\text{cont slot}] & \equiv p_c \\ & = (1-p_0) \left(1 + \sum_{i=1}^{\infty} (p_0)^i \Pr[\text{a node misses } i \text{ cont slots}] \right) \end{aligned} \quad (9)$$

and

$$\Pr[\text{non-cont slot}] \equiv p_{nc} = (1-p_c). \quad (10)$$

We will explore the probability of missed contention packets in more detail later. Substituting (10) into (6), we have

$$\begin{aligned} \bar{\Pi} = & \frac{t_{mf}}{t_f} + n \left\{ p_{nc} \frac{t_{wnc}}{t_f} \times \Pr[\text{awake for non-cont slot}] \right. \\ & \left. + (1-p_{nc}) \frac{t_{wc}}{t_f} \times \Pr[\text{awake for cont slot}] \right\}. \end{aligned} \quad (11)$$

We now solve for the probabilities that a node will be awake for the non-contention and contention slot as well as the time a node will stay awake in each of these slots.

The probability that a node will be awake for a non-contention slot is the probability that either the node is the slot owner and it has non-contention traffic to transmit or it is the intended receiver for another node that has traffic to transmit in its slot. This is summarized as

$$\begin{aligned} \Pr[\text{awake for non-cont slot}] = & \\ & \Pr[\text{node is slot owner}] \\ & \quad \times \Pr[\text{node has non-cont packets to xmt}] \\ & + \Pr[\text{node is not slot owner}] \\ & \quad \times \Pr[\text{slot owner has non-cont packets to xmt}] \\ & \quad \times \Pr[\text{node is destination}]. \end{aligned} \quad (12)$$

If we assume that a node is assigned exactly one slot in a transmission frame, then the probability that a node is a slot owner is uniformly distributed with

$$\begin{aligned} \Pr[\text{node is slot owner}] & = \frac{1}{n} \text{ and} \\ \Pr[\text{node is not slot owner}] & = \frac{n-1}{n}. \end{aligned} \quad (13)$$

The probability that a given node has non-contention traffic to transmit is equivalent to the probability that at least one non-contention packet arrived at the node during the prior transmission frame. Assuming Poisson packet arrivals, this is

$$\Pr[\text{a node has non-cont packets to xmt}] = 1 - e^{-\lambda_{nc} t_f} \quad (14)$$

where λ_{nc} is the per node non-contention packet arrival rate. Substituting (13) and (14) into (12) and simplifying, we have

$$\Pr[\text{awake for non-cont slot}] = \frac{1}{n} \left(1 - e^{-\lambda_{nc} t_f}\right) \times \{1 + (n-1) \times \Pr[\text{node is destination}]\}. \quad (15)$$

Finally, neglecting the time required to sample for the contention beacon (which is much smaller than either the frame time or the slot time), a node will be awake in a transmission frame long enough to receive the non-contention traffic or, from Figure 1,

$$t_{wnc} = t_s - t_{IFS} - (t_{prop} + t_{guard}) \quad (16)$$

where t_{prop} is the maximum propagation distance and t_{guard} is the guard band designed to accommodate slot synchronization errors.

The probability that a node will be awake for a contention slot is the probability that it will detect the contention slot as in

$$\begin{aligned} \Pr[\text{awake for cont slot}] = & \Pr[\text{node has cont pkt to xmt}] \\ & + \Pr[\text{node has no cont pkt to xmt}] \\ & \quad \times \Pr[\text{sample for cont beacon}] \\ & + \Pr[\text{node has no cont pkt to xmt}] \\ & \quad \times \Pr[\text{not sample for cont beacon}] \\ & \quad \times \Pr[\text{awake for non-cont slot}] \end{aligned} \quad (17)$$

which is, from (15) and the analysis of (9),

$$\begin{aligned} \Pr[\text{awake for cont slot}] = & p_{xmt} + (1 - p_{xmt}) p_s \\ & + \frac{(1 - p_{xmt})(1 - p_s)}{n} \left(1 - e^{-\lambda_{nc} t_f}\right) \\ & \quad \times \{1 + (n-1) \times \Pr[\text{node is destination}]\} \end{aligned} \quad (18)$$

where

$$\begin{aligned} p_{xmt} \equiv & \Pr[\text{node has cont pkt to xmt}] \\ = & (1 - p'_0) \left(1 + \sum_{i=1}^{\infty} (p'_0)^i \Pr[\text{node misses } i \text{ cont slots}]\right) \end{aligned} \quad (19)$$

and $p'_0 = e^{-\frac{\Lambda_c t_s}{n}}$. Note that $\Pr[\text{awake for cont slot}] = 1$ for $p_s = 1$, as expected. The time a node will stay awake during a contention slot can be broken into two parts: (1) the time a node stays awake during the contention phase of the slot, t_{wc_c} , (which is comprised of the minislots) and (2) the time a node stays awake during the non-contention phase of the contention slot, $t_{wc_{nc}}$, (when the slot owner is permitted to transmit its non-contention traffic). From Figure 1, for a contention slot with k minislots each of length t_{ms} , the latter is

$$t_{wc_{nc}} = (t_s - t_b - kt_{ms}) \times \Pr[\text{awake for non-cont slot}]. \quad (20)$$

To calculate the former, we must examine each minislot individually using the slotted ALOHA with periodic server vacation analysis of [20].

At a minimum, a node will wake up to receive the minislot destination bitmap for every minislot and

$$\min(t_{wc_c}) = kt_{mms}. \quad (21)$$

To find the mean value, we must calculate the probability that a node will stay awake for each minislot. This is the probability that either the node will attempt to transmit in this minislot or it will be the destination for another node's successful transmission. In general,

$$\begin{aligned} \Pr[\text{awake for minislot}] = & \Pr[\text{attempt xmsn in minislot}] \\ & + \Pr[\text{not attempt xmsn in minislot}] \\ & \quad \times \Pr[\text{successful xmsn in minislot}] \\ & \quad \times \Pr[\text{node is destination}]. \end{aligned} \quad (22)$$

This is a recursive equation, because, from [20], the probability of both an attempted transmission and a successful transmission depend up the outcome of the prior minislots. We can achieve a closed form solution for the duty cycle if we make the assumption that a node stays awake for all of the minislots (this can clearly be seen to be a conservative estimation for t_{wc_c}). Thus,

$$t_{wc_c} = kt_{ms}. \quad (23)$$

Combining (20) and (23), we have

$$\begin{aligned} t_{wc} = & kt_{ms} + (t_s - t_b - kt_{ms}) \\ & \quad \times \Pr[\text{awake for non-cont slot}]. \end{aligned} \quad (24)$$

Substituting (15), (16), (18) and (24) into (11), we finally arrive at our result for the mean duty cycle of

$$\begin{aligned} \bar{\Pi} = & \frac{t_{mf}}{t_f} + n \left\{ p_{nc} \frac{t_s - t_{IFS} - (t_{prop} + t_{guard})}{t_f} \right. \\ & \times \frac{1}{n} \left(1 - e^{-\lambda_{nc} t_f}\right) \left(1 + (n-1) \times \Pr[\text{node is dest}]\right) \\ & + \frac{(1 - p_{nc})}{t_f} \left[p_{xmt} + (1 - p_{xmt}) p_s + (1 - p_{xmt})(1 - p_s) \right. \\ & \quad \times \frac{1}{n} \left(1 - e^{-\lambda_{nc} t_f}\right) \left(1 + (n-1) \times \Pr[\text{node is dest}]\right) \left. \right] \\ & \times \left[kt_{ms} + (t_s - t_b - kt_{ms}) \right. \\ & \quad \left. \times \frac{1}{n} \left(1 - e^{-\lambda_{nc} t_f}\right) \left(1 + (n-1) \times \Pr[\text{node is dest}]\right) \right] \left. \right\}. \end{aligned} \quad (25)$$

Checking the limits, the minimum duty cycle represents the case where there is no traffic at all (i.e., $\lambda_{nc} = \Lambda_c = 0$). In this case, $p_{nc} = p_0 = 1$, $e^{-\lambda_{nc} t_f} = 1$ and (25) reduces to

$$\bar{\Pi} = \frac{t_{mf}}{t_f} \quad (26)$$

(the term that includes kt_{ms} is zeroed out in this case) which is the expected result from (4). For $\lambda_{nc} \rightarrow \infty$ with no contention traffic present ($\Lambda_c = 0$), $p_{nc} = p_0 = 1$, $e^{-\lambda_{nc}t_f} = 0$ and (25) reduces to

$$\bar{\Pi} = \frac{t_{mf}}{t_f} + \frac{t_s - t_{IFS} - (t_{prop} + t_{guard})}{t_f} \times (1 + (n-1) \times \Pr[\text{node is dest}]) \quad (27)$$

For $\Lambda_c \rightarrow \infty$ with no non-contention traffic present ($\lambda_{nc} = 0$), $p_{nc} = p_0 = 0$, $e^{-\lambda_{nc}t_f} = 1$ and (25) reduces to

$$\bar{\Pi} = \frac{t_{mf} + p_s n k t_{ms}}{t_f}. \quad (28)$$

Finally, for the limit in which both $\lambda_{nc} \rightarrow \infty$ and $\Lambda_c \rightarrow \infty$, $p_{nc} = p_0 = 0$, $e^{-\lambda_{nc}t_f} = 0$ and

$$\begin{aligned} \bar{\Pi} &= \frac{t_{mf}}{t_f} + \frac{n}{t_f} \left[[p_{xmt} + (1-p_{xmt})p_s + (1-p_{xmt})(1-p_s)] \right. \\ &\quad \times \frac{1}{n} (1 + (n-1) \times \Pr[\text{node is dest}]) \left. \right] \\ &\quad \times [k t_{ms} + (t_s - t_b - k t_{ms})] \\ &\quad \times \frac{1}{n} (1 + (n-1) \times \Pr[\text{node is dest}]) \left. \right]. \quad (29) \end{aligned}$$

We plot (25) for $p_s = 1$ and $\Pr[\text{node is dest}] = 1$ (every packet is transmitted to all nodes) as a function of the non-contention packet arrival rate in Figure 4 and the contention packet arrival rate in Figure 5. As expected, duty cycle increases with increasing packet arrival rate and reflects the bounds derived in (26) through (29). The only exception is in Figure 5, where the duty cycle decreases slightly as λ_{nc} increases for large values of Λ_c . This is because as Λ_c increases, more slots are seized as contention slots and the time a node stays awake during a contention slot is less than that for a non-contention slot. This is due to the added overhead of the contention beacon and is apparent in a comparison of (16) and (24). We also plot the results in Figures 6 and 7 for the case where the mean node degree is three and a packet is randomly transmitted to a single neighbor based on a uniform distribution ($\Pr[\text{node is dest}] = 0.33$). It should be noted that in the cases in which $\Lambda_c = 0$, the protocol does not make use of the contention mode and the plots are not dependent on the assumption used to arrive at (23). For the plots of Figures 5-7, the frame size is 1 s, the minislot size is 0.001 s and the beacon time is 0.001 s. There are 5 slots in a frame, 50 minislots in a contention slot and the channel data rate is 1 Mbps. The duty cycle performance is normalized to the frame time which can

therefore be scaled to values more representative of typical wireless sensor applications (usually 1 to 10 ms) as desired.

B. Throughput and delay

In this section, we derive the throughput and delay for our proposed scheme. We address the non-contention and contention modes individually. These results can be combined to arrive at the overall throughput and delay using the results of [19].

The throughput and delay for the non-contention mode (S^{nc} and D^{nc} , respectively) of the proposed scheme can be found by accounting for (9) and (10) in the analysis [19]. The throughput is

$$S^{nc} = \begin{cases} \frac{\Lambda_{nc} \bar{L}}{R} & \Lambda_{nc} \leq S_{\max}^{nc} \left(\frac{R}{\bar{L}} \right) \\ S_{\max}^{nc} & \text{otherwise} \end{cases} \quad (30)$$

for

$$S_{\max}^{nc} = \frac{t_{wnc} + (1-p_{nc})(t_{IFS} - t_b - k t_{ms})}{t_s} \quad (31)$$

where \bar{L} is the mean non-contention packet size in bits. The delay is

$$\begin{aligned} \bar{D}^{nc} &= \frac{t_s}{2} + \frac{\lambda_{nc} \left(\frac{1 - (1-p_{nc})^\kappa}{p_{nc}} \right)^2}{2 \left(1 - \lambda_{nc} \left(\frac{1 - (1-p_{nc})^\kappa}{p_{nc}} \right) \right)} (1 + \Upsilon) \\ &\quad + [t_f - t_{wc_nc}] \left(\frac{\kappa (1-p_{nc})^{\kappa+1} - (\kappa+1)(1-p_{nc})^\kappa + 1}{p_{nc}} \right) \\ &\quad + \left[\kappa (t_f - t_{wc_nc}) - t_{IFS} + t_b + k t_{ms} \right] (1-p_{nc})^\kappa \\ &\quad - t_f + t_{IFS} + t_{nc} + t_{wc_nc} + t_{prop} \end{aligned} \quad (32)$$

where

$$\begin{aligned} \Upsilon &= \left(\frac{1-p_{nc}}{(1-(1-p_{nc})^\kappa)^2} \right) \\ &\quad \times [(1-p_{nc})^{2\kappa} + ((\kappa-1)^2 - 1)(1-p_{nc})^\kappa \\ &\quad - (2\kappa^2 - 2\kappa - 1)(1-p_{nc})^{\kappa-1} \\ &\quad + \kappa^2 (1-(p_{nc})^2)(1-p_{nc})^{\kappa-2} - 1], \end{aligned} \quad (33)$$

and

$$\kappa = \frac{t_{wnc}}{t_{wc_nc}}. \quad (34)$$

The throughput and delay for the contention mode (S^c and D^c , respectively) can again be derived from the analysis in [19]. If we make the assumption that all pending contention

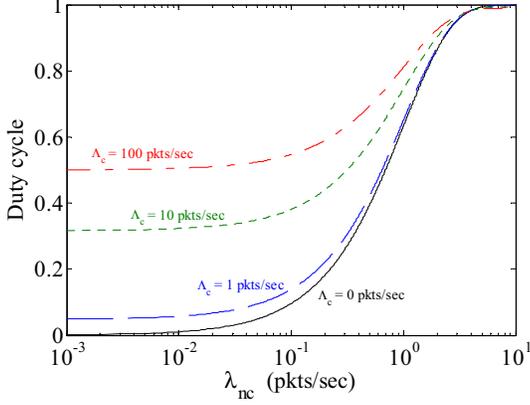


Figure 4. Duty cycle plotted as a function of non-contention packet arrival rates for various values of contention packet arrival rate with $\Pr[\text{node is dest}] = 1$.

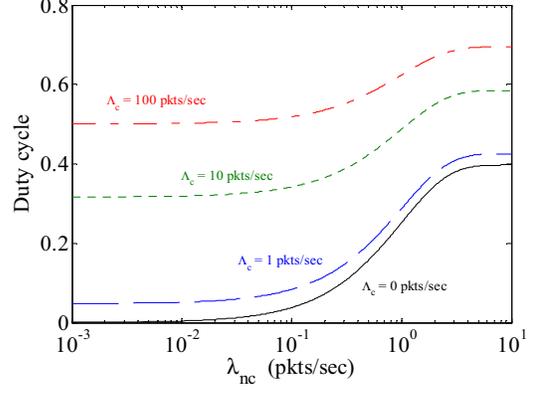


Figure 6. Duty cycle plotted as a function of non-contention packet arrival rates for various values of contention packet arrival rate with $\Pr[\text{node is dest}] = 0.33$.

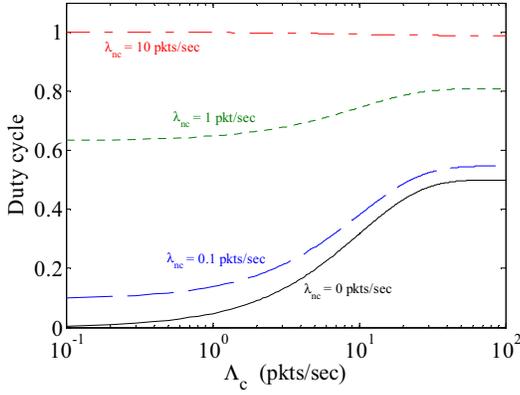


Figure 5. Duty cycle plotted as a function of contention packet arrival rates for various values of non-contention packet arrival rate with $\Pr[\text{node is dest}] = 1$.

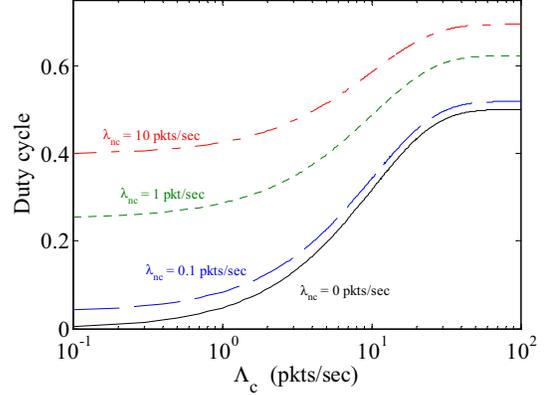


Figure 7. Duty cycle plotted as a function of contention packet arrival rates for various values of non-contention packet arrival rate with $\Pr[\text{node is dest}] = 0.33$.

packets with the exception of those whose destination node is asleep are successfully transmitted in a contention slot, then the repetition of a contention slot only reduces the overall contention mode throughput if it is repeated when no new contention packet has arrived in the previous slot. In this case, rather than calculating the throughput across the contention slot as in [19], we must now calculate it across multiple slots as in

$$S^c = \frac{L_c}{R} \sum_{i=0}^{\infty} \left(\frac{Q(0)}{(i+1)t_s} \right) (p'_0)^i \Pr[\text{node misses } i \text{ cont slots}] \quad (35)$$

where L_c is the mean contention packet size in bits and $Q(0)$ is the mean number of nodes with a contention packet pending for transmission at the beginning of a contention slot.

Turning to the contention delay, our analysis reflects the observation that when a node misses a contention slot, it must wait until the next contention slot to receive the intended contention mode packet. Thus, for every missed contention slot, the packet delay is increased by t_s and the mean is thus

$$D^c = D_{p_s=1}^c + \sum_{i=1}^{\infty} i t_s \Pr[\text{node misses } i \text{ cont slots}] \quad (36)$$

where $D_{p_s=1}^c$ is the delay associated with the contention mode for $p_s = 1$ (i.e., $\Pr[\text{missed cont slot}] = 0$) which can be found numerically as shown in [19].

C. Effect of p_s

To clearly see the role of p_s , we return to the probability of a missed contention slot. A node will sleep through a contention slot if (1) it is not scheduled to be awake during the original non-contention slot that was redesignated as a contention slot, (2) it does not have any contention traffic to transmit and (3) it does not sample for a contention beacon. Thus, from (18), it can be shown [21] that as p_s increases, the $\Pr[\text{a node misses a cont slot}]$ increases and, from (9) and (10), p_{nc} decreases. From the results in the previous subsection, we can see that non-contention mode throughput decreases and

non-contention delay increases with decreasing p_{nc} . We also see that the non-contention throughput also decreases and contention delay also increases with increasing $\Pr[\text{a node misses a cont slot}]$. Thus, throughput and delay performance in both modes drops off as p_s increases.

To see the effect of p_s on duty cycle (and, therefore, energy consumption), we plot (25) as a function of p_s in Figure 10. As expected, in most cases, we see that the duty cycle strictly decreases (and hence the energy consumption decreases) with decreasing p_s . There are two exceptions to this observation in which the duty cycle appears to remain constant with respect to p_s . The first can be seen in Figure 10 when the contention packet arrival rate is zero. In this case, the duty cycle is clearly independent of p_s . This is also reflected in (25) which reduces to

$$\bar{\Pi} = \frac{t_{mf}}{t_f} + n \left\{ \frac{t_s - t_{IFS} - (t_{prop} + t_{guard})}{t_f} \right. \\ \left. \times \frac{1}{n} (1 - e^{-\lambda_{nc} t_f}) (1 + (n-1) \times \Pr[\text{node is dest}]) \right\} \quad (37)$$

because $e^{-\Lambda_c t_f} = 1$, $p_0 = 1$, $p_c = 0$, and, thus, $p_{nc} = 1$ for $\Lambda_c = 0$ from (7), (9), and (10). The second exception appears to occur when the non-contention packet arrival rate is very large and the non-contention mode is saturated. Despite the minimal impact of a change in p_s in this case, it can be shown [21] that the duty cycle is still a function of p_s as $\lambda_{nc} \rightarrow \infty$ (with the exception of the above-mentioned case in which $\Lambda_c = 0$).

In general, then, it can be seen that when contention flow traffic is non-zero and the non-contention mode is not saturated, the parameter provides p_s provides a mechanism to

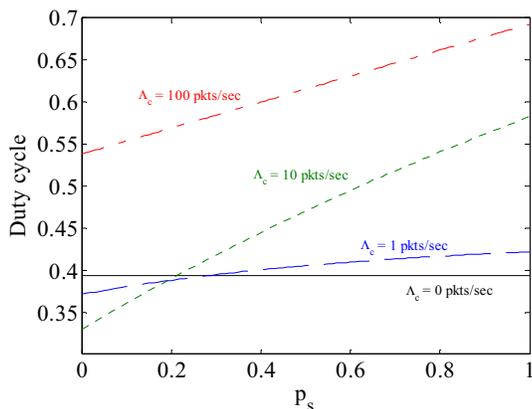


Figure 10. Duty cycle plotted as a function of the sampling probability for various values of contention packet arrival rate with a non-contention packet arrival rate of 0.5 pkts/sec and $\Pr[\text{node is dest}] = 1$.

trade-off delay and throughput performance for energy efficiency. As p_s is decreased, throughput decreases and delay increases, but energy consumption (as captured by the duty cycle) decreases. Alternately, as p_s is increased, throughput increases and delay decreases, but energy consumption increases.

V. SIMULATION RESULTS

We now compare the analysis of the previous section with simulation results obtained using the OPNET Modeler Suite. The simulation includes 10 nodes with a channel data rate of 1 Mbps, a frame length of 1 s, a beacon duration and minislot size of 1 ms, and a data interframe space of 0.1 ms. From these inputs, the frame destination bitmap size is 0.1 ms and the slot size is 99.99 ms. The per node non-contention packet arrival rate is allowed to vary from zero to 10 packets/sec for various values of the aggregate contention packet arrival rate.

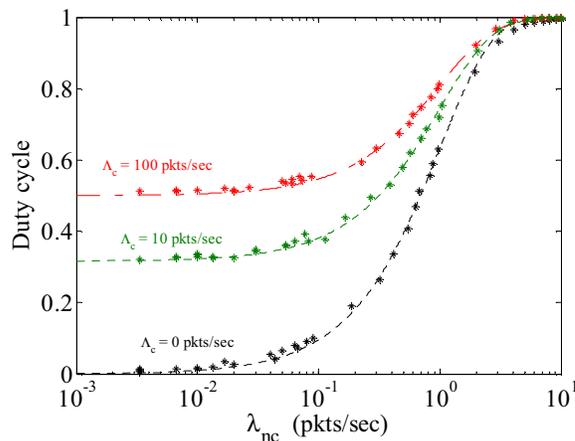


Figure 8. Comparison of the analysis from Section IV (dashed lines on plot) with steady state OPNET simulation results (discrete points on plot) for the duty cycle plotted as a function of non-contention packet arrival rates for various values of contention packet arrival rate with $\Pr[\text{node is dest}] = 1$.

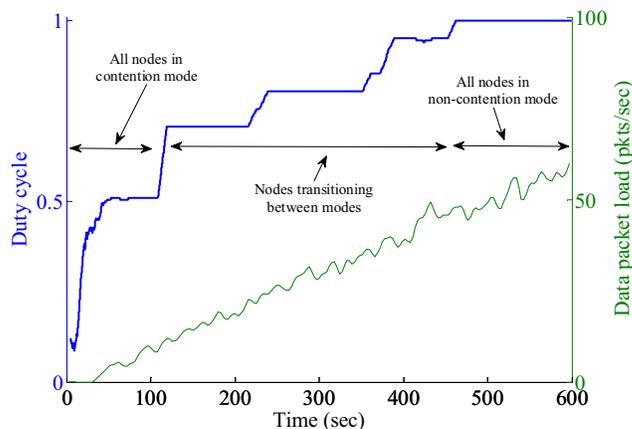


Figure 9. Transient results for duty cycle as a function of increasing data packet arrival rate. The control packet arrival rate is constant at 0.1 pkts/sec.

The frame time value has been selected to reduce simulation run time but can be scaled to values more representative of typical wireless sensor applications (1 to 10 ms) because the duty cycle performance is normalized to the frame time. The upper bound is simulated by allowing nodes to remain awake throughout the contention portion of the contention slots.

We examine both the steady state and the transient results. The former are compared with the analysis of the previous section for $p_s = 1$ and $\text{Pr}[\text{node is dest}] = 1$ (see Figure 6) in Figure 8. The duty cycle seen in the simulations closely follows that from (25) for all values of λ_{nc} and Λ_c .

The transient results are provided in Figure 9. In this plot, the control flow remains constant at 0.1 pkts/sec while the data packet flow is increased from zero to more than 50 pkts/sec. As a function of time, we plot the duty cycle on the left axis and the per node data packet arrival rate on the right axis. We can see three regions of operation emerge as the data packet arrival rate is increased. In the first region, all of the nodes are transmitting the data packets in the contention mode. Here, the duty cycle is defined by the contention mode traffic flow and its performance mirrors that in the $\lambda_{nc} = 0$ curve in Figure 5. In this region, the duty cycle reaches a maximum defined by (28). In the next region, nodes begin to transition the data packet flow to the non-contention mode and the duty cycle rises in increments as nodes complete the transition. Finally, all nodes will have completed the transition and the duty cycle rises towards the maximum value in Figure 4.

VI. CONCLUSIONS

In this work, we proposed an energy-efficient flow-specific medium access scheme based on an adaptive sleep and wake cycle using preamble sampling. We provided analysis to show that the duty cycle (and hence the energy consumption) of the proposed scheme decreases as the packet arrival rate decreases. We also introduced a preamble sampling probability parameter that was shown to be capable of managing the trade-off between delay and throughput and energy efficiency. Simulation results were provided to validate the analysis and together they demonstrate that our proposed scheme approaches the low duty cycles needed to support the rigorous demands of energy-constrained wireless sensor network applications.

REFERENCES

[1] T. O. Walker, M. Tummala, and J. McEachen, "Traffic-adaptive, flow-specific medium access control for wireless networks," *Proc. of the IEEE Wireless Communications and Networking Conf.*, Budapest, Hungary, 5-8 Apr. 2009.

[2] T. O. Walker, M. Tummala, J. McEachen and J. B. Michael, "Medium Access for Large-scale Wireless Networks for Missile Defense," *Proc. 2009 IEEE Int. Conf. on System of Systems Engineering*, Albuquerque, NM, Jun. 2009, pp. 1-5.

[3] L. Kleinrock, and F. Tobagi, "Packet switching in radio channels: Part I - Carrier sense multiple-access modes and their throughput-delay characteristics," *IEEE Trans. on Commun.*, vol. 23, no. 12, pp. 1400-1416, Dec. 1975.

[4] V. Rajendran, K. Obraczka, and J.J. Garcia-Luna-Aceves, "Energy-efficient, collision-free medium access control for wireless sensor networks," *Wireless. Networks*, vol. 12, no. 1, pp. 63-78, Feb. 2006.

[5] I. Rhee, A. Warrier, M. Aia, and J. Min, "Z-MAC: A hybrid MAC for wireless sensor networks," *Proc. of the 3rd Int'l Conf. on Embedded Networked Sensor Systems*, San Diego, California, Nov. 2005, pp. 90-101.

[6] P. Vincent, M. Tummala, and J. McEachen, "A New Method for Distributing Power Usage across a Sensor Network," *Proc. of 3rd Ann. IEEE Comm. Society on Sensor and Ad Hoc Comm. and Networks*, Sept. 2006, vol. 2, pp. 518-526.

[7] W. A. Lintz and J. C. McEachen, "A Method for Emphasizing Signal Detection in Wireless Sensor Network Radio Frequency Array Operation," *Proc. of 42nd Ann. Hawaii Inter. Conf. on System Sciences*, Hawaii, Jan. 5-8, 2009.

[8] I. Demirkol, C. Ersoy, and F. Alagoz, "MAC protocols for wireless sensor networks: a survey," *IEEE Communications Magazine*, vol. 44, no. 4, pp. 115-121, Apr. 2006.

[9] W. Ye, J. Heidemann, and D. Estrin, "Medium Access Control with Coordinated Adaptive Sleeping for Wireless Sensor Networks," *IEEE/ACM Trans. Net.*, vol. 12, no. 3, pp. 493-506, Jun. 2004.

[10] "IEEE Standard for Information technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999)*, Jun. 12, 2007.

[11] Y.-C. Tseng, C.-S. Hsu and T.-Y. Hsieh, "Power-saving protocols for IEEE 802.11-based multi-hop ad hoc networks," *21st Proc. Ann. Joint Conf. of the IEEE Computer and Communications Societies*, 2002, pp. 200-209.

[12] S. Singh, and C. S. Raghavendra, "PAMAS—power aware multi-access protocol with signaling for ad hoc networks," *SIGCOMM Comput. Commun. Rev.*, vol. 28, no. 3, pp. 5-26, Jul. 1998.

[13] J. M Rabaey et al., "PicoRadio supports ad hoc ultra-low power wireless networking," *IEEE Computer*, vol. 33, no. 7, pp. 42-48, July 2000.

[14] A. El-Hoiydi, "Aloha with preamble sampling for sporadic traffic in ad hoc wireless sensor networks," *IEEE Intl. Conf. on Communication*, vol. 5, April 2002, pp. 3418-3423.

[15] A. El-Hoiydi and J. D. Decotignie, "WiseMAC: An ultra low-power MAC protocol for multi-hop wireless sensor networks," *1st Intl. Workshop on Algorithmic Aspects of Wireless Sensor Networks*, Turku, Finland, July 16, 2004.

[16] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," *Proc. of the 2nd Intl. Conf. on Embedded Networked Sensor Systems*, Baltimore, MD, Nov. 03-05, 2004, pp. 95-107.

[17] T. V. Dam and K. Langendoen, "An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks," *1st ACM Conf. Embedded Networked Sensor Sys.*, Los Angeles, CA, Nov. 2003.

[18] G. Lu, B. Krishnamachari, and C. S. Raghavendra, "An Adaptive Energy-Efficient and Low-Latency MAC for Data Gathering in Wireless Sensor Networks," *Proc. 18th Int'l. Parallel and Distrib. Processing Symp.*, Apr. 2004.

[19] T. O. Walker, M. Tummala, and J. McEachen, "Traffic-adaptive, flow-specific medium access for wireless networks," Naval Postgraduate School, Monterey, CA, Tech. Rep., in draft.

[20] T. O. Walker, M. Tummala, and J. McEachen, "Performance analysis of slotted ALOHA with periodic server vacations for energy-efficient medium access," *Forty-second Asilomar Conf. on Signals, Systems and Computers*, Monterey, CA, Nov. 2009, accepted for publication.

[21] T. O. Walker, "Traffic-adaptive, flow-specific medium access control for wireless networks," Ph.D. dissertation, Dept. Elec. And Comp. Engr., Naval Postgraduate School, Monterey, CA, under preparation.