

Medium Access Control with Coordinated, Adaptive Sleeping for Wireless Sensor Networks

Wei Ye, John Heidemann, Deborah Estrin

Abstract—This paper proposes S-MAC, a medium-access control (MAC) protocol designed for wireless sensor networks. Wireless sensor networks use battery-operated computing and sensing devices. A network of these devices will collaborate for a common application such as environmental monitoring. We expect sensor networks to be deployed in an ad hoc fashion, with individual nodes remaining largely inactive for long periods of time, but then becoming suddenly active when something is detected. These characteristics of sensor networks and applications motivate a MAC that is different from traditional wireless MACs such as IEEE 802.11 in several ways: energy conservation and self-configuration are primary goals, while per-node fairness and latency are less important. S-MAC uses three novel techniques to reduce energy consumption and support self-configuration. To reduce energy consumption in listening to an idle channel, nodes periodically sleep. Neighboring nodes form *virtual clusters* to auto-synchronize on sleep schedules. Inspired by PAMAS, S-MAC also sets the radio to sleep during transmissions of other nodes. Unlike PAMAS, it only uses in-channel signaling. Finally, S-MAC applies *message passing* to reduce contention latency for sensor-network applications that require store-and-forward processing as data move through the network. We evaluate our implementation of S-MAC over a sample sensor node, the UCB Mote. The experimental results show that, on a source node, an 802.11-like MAC consumes 2–6 times more energy than S-MAC for traffic load with messages sent every 1–10s.

Keywords—Medium access control, Sensor network, Wireless network, Energy efficiency

I. INTRODUCTION

WIRELESS sensor networking is an emerging technology that has a wide range of potential applications including environment monitoring, smart spaces, medical systems and robotic exploration. Such networks will consist of large numbers of distributed nodes that organize themselves into a multi-hop wireless network. Each node has one or more sensors, embedded processors and low-power radios, and is normally battery operated. Typically, these nodes coordinate to perform a common task.

Like in all shared-medium networks, medium access control (MAC) is an important technique that enables the successful operation of the network. One fundamental task of the MAC protocol is to avoid collisions so that two interfering nodes do not transmit at the same time. There are many MAC protocols that have been developed for wireless voice and data communication networks. Typical examples include the time division multiple access (TDMA), code division multiple access (CDMA), and contention-based protocols like IEEE 802.11 [1].

This work is in part supported by NSF under grant ANI-0220026 as the MACSS project and under grant ANI-9979457 as the SCOWR project, and by DARPA under grant DABT63-99-1-0011 as the SCADDS project and under contract N66001-00-C-8066 as the SAMAN project. The work is also support by the Center for Embedded Networked Sensing and a grant from the Intel Corporation. W. Ye (weiye@isi.edu) and J. Heidemann (johnh@isi.edu) are with the Information Sciences Institute (ISI), University of Southern California (USC). D. Estrin (destrin@cs.ucla.edu) is with the Center for Embedded Networked Sensing and the Computer Science Department, University of California at Los Angeles and USC/ISI.

To design a good MAC protocol for the wireless sensor networks, we have considered the following attributes. The first is the energy efficiency. As stated above, sensor nodes are likely to be battery powered, and it is often very difficult to change or recharge batteries for these nodes. In fact, someday we expect some nodes to be cheap enough that they are discarded rather than recharged. Prolonging network lifetime for these nodes is a critical issue. Another important attribute is scalability and adaptivity to changes in network size, node density and topology. Some nodes may die over time; some new nodes may join later; some nodes may move to different locations. A good MAC protocol should gracefully accommodate such network changes. Other typically important attributes including fairness, latency, throughput and bandwidth utilization may be secondary in sensor networks.

This paper presents sensor-MAC (S-MAC), a MAC protocol explicitly designed for wireless sensor networks. While reducing energy consumption is the primary goal in our design, our protocol has also achieved good scalability and collision avoidance by utilizing a combined scheduling and contention scheme. To achieve the primary goal of energy efficiency, we need to identify what are the main sources that cause inefficient use of energy as well as what trade-offs we can make to reduce energy consumption.

We have identified the following major sources of energy waste. The first one is *collision*. When a transmitted packet is corrupted it has to be discarded, and the follow-on re-transmissions increase energy consumption. Collision increases latency as well. The second source is *overhearing*, meaning that a node picks up packets that are destined to other nodes. The third source is *control packet overhead*. Sending and receiving control packets consumes energy too. The last major source of inefficiency is *idle listening*, *i.e.*, listening to receive possible traffic that is not sent. This is especially true in many sensor network applications. If nothing is sensed, nodes are in idle mode for most of the time. However, in many MAC protocols such as IEEE 802.11 ad hoc mode or CDMA nodes have to listen to the channel to receive possible traffic. Measurements have shown that idle listening consumes 50–100% of the energy required for receiving. For example, Stemm and Katz measure that the idle:receive:send ratios are 1:1.05:1.4 [2], while the Digtan wireless LAN module (IEEE 802.11/2Mbps) specification shows idle:receive:send ratios is 1:2:2.5 [3]. Most sensor networks are designed to operate for long time, and nodes will be in idle state for long time. Thus, idle listening is a dominant factor of energy waste in such cases.

S-MAC tries to reduce energy waste from all the above sources. In exchange it accepts some reduction in both per-hop

fairness and latency. The first technique of S-MAC is to establish the low-duty-cycle operation on nodes in a multi-hop ad hoc network. It reduces idle listening by periodically putting nodes into sleep state. In the sleep mode, the radio is completely turned off. In protocols for traditional data networks like the IEEE 802.11, bandwidth utilization is a big concern, and nodes normally operate in fully active mode. Switching to low-duty-cycle mode (called power save mode in the IEEE 802.11 standard) is an option of each node, and it normally happens when a node has been idle for long time. In S-MAC, however, the low-duty-cycle mode is the default operation of all nodes in the network. Nodes only become more active when there is traffic in the network. To reduce the control overhead and latency, S-MAC introduces coordinated sleeping among neighboring nodes.

An important feature of wireless sensor networks is the in-network data processing. It can greatly reduce energy consumption compared to transmitting all the raw data to the end node [4], [5], [6]. In-network processing requires store-and-forward processing of messages. A message is a meaningful unit of data that a node can process (average or filter, etc.). It may be long and consists of many small fragments. In this case, MAC protocols that promote fragment-level fairness actually increase message-level latency for the application. In contrast, message passing reduces message-level latency by trading off the fragment-level fairness.

In traditional wireless voice or data networks, each user desires equal opportunity and time to access the medium, *i.e.*, sending or receiving packets for their own applications. Per-hop MAC level fairness is thus an important issue. However, in sensor networks, all nodes cooperate for a single common task. At any particular time, one node may have dramatically more data to send than some other nodes. In this case fairness is not important as long as application-level performance is not degraded. In our protocol, we re-introduce the concept of message passing to efficiently transmit very long messages. The basic idea is to divide the long message into small fragments and transmit them in a burst. The result is that a node who has more data to send gets more time to access the medium. From a per-hop, MAC level perspective, this is unfair for those nodes who only have some short packets to send, since their short packets have to wait a long time for very long packets. However, as we will show later, message passing can achieve energy savings by reducing control overhead and avoiding overhearing. And it is well suited to applications where nodes support in-network processing of data.

Latency can be important or unimportant depending on what application is running and the node state. During a period that there is no sensing event, there is normally very little data flowing in the network. Most of the time nodes are in idle state. Sub-second latency is not important, and we can trade it off for energy savings. S-MAC therefore lets nodes periodically sleep if otherwise they are in the idle listening mode. In the sleep mode, a node will turn off its radio. The design reduces the energy consumption due to idle listening. However, the latency is increased, since a sender must wait for the receiver to wake up before it can send out data.

To demonstrate the effectiveness and measure the perfor-

mance of S-MAC, we have implemented it on our testbed wireless sensor nodes, *Motes*, developed by University of California, Berkeley [7] and manufactured and sold by Crossbow Technology, Inc. [8] The latest version of mote, Mica, has a 8-bit Atmel ATmega128L microcontroller running at 4 MHz. It has a low power radio transceiver module TR1000 [9] or TR3000 [10] from RF Monolithics, Inc., which operates at 916.5 MHz or 433.92MHz. The mote runs on a very small event-driven operating system called TinyOS [11]. In order to compare the performance of our protocol with some other protocols, we also implemented an IEEE 802.11-like MAC protocol on this platform.

The contributions of this paper are listed as follows.

- The scheme of periodic listen and sleep reduces energy consumption by avoiding idle listening. The use of synchronization to form virtual clusters of nodes on the same sleep schedule. These schedules coordinate nodes to minimize additional latency.
- The use of in-channel signaling to put each node to sleep when its neighbor is transmitting to another node. This method avoids the overhearing problem and is inspired by PAMAS [12], but does not require an additional channel.
- Applying message passing to reduce application-perceived latency and control overhead. Per-node fragment-level fairness is reduced since sensor network nodes are often collaborating towards a single application.
- Evaluating an implementation of our new MAC over sensor-net specific hardware.

The early work of S-MAC was published in [13]. This paper includes significant extensions in the protocol design, implementation and experiments:

- Support for traffic-adaptive sleep schedules.
- Measurement and evaluation of the trade-offs on energy, latency and throughput.

II. RELATED WORK

Medium access control is a broad research area, including work in the new area of low power and wireless sensor networks [14], [15], [16], [17]. Current MAC design for wireless sensor networks can be broadly divided into contention-based and TDMA protocols.

Contention-based MACs. The standardized IEEE 802.11 distributed coordination function (DCF) [1] is an example of the contention-based protocol, and is mainly built on the research protocol MACAW [18]. It is widely used in ad hoc wireless networks because of its simplicity and robustness to the hidden terminal problem. However, recent work [2] has shown that the energy consumption using this MAC is very high when nodes are in idle mode. This is mainly due to the idle listening. 802.11 has a power save mode, and we will discuss it shortly. PAMAS [12] made an improvement on energy savings by trying to avoid the overhearing among neighboring nodes. Our paper also exploits the same idea. The main difference of our work with PAMAS is that we do not use any out-of-channel signaling. Whereas in PAMAS, it requires two independent radio channels, which in most cases indicates two independent radio systems on each node. PAMAS does not attempt to reduce idle listening.

TDMA-based MACs. The other class of MAC protocols are based on reservation and scheduling, for example TDMA-based protocols. TDMA protocols have a natural advantage of energy conservation compared to contention protocols, because the duty cycle of the radio is reduced and there is no contention-introduced overhead and collisions. However, using TDMA protocol usually requires the nodes to form *real* communication clusters, like Bluetooth [19], [20] and LEACH [16]. Most nodes in a real cluster are restricted to communicate within the cluster. Managing inter-cluster communication and interference is not an easy task. Moreover, when the number of nodes within a cluster changes, it is not easy for a TDMA protocol to dynamically change its frame length and time slot assignment. So its scalability is normally not as good as that of a contention-based protocol. For example, Bluetooth may have at most 8 active nodes in a cluster.

Sohrabi and Pottie [15] proposed a self-organization protocol for wireless sensor networks. Each node maintains a TDMA-like frame, called super frame, in which the node schedules different time slots to communicate with its known neighbors. At each time slot, it only talks to one neighbor. To avoid interference between adjacent links, the protocol assigns different channels, *i.e.*, frequency (FDMA) or spreading code (CDMA), to potentially interfering links. Although the super frame structure is similar to a TDMA frame, it does not prevent two interfering nodes from accessing the medium at the same time. The actual multiple access is accomplished by FDMA or CDMA. A drawback of the scheme is its low bandwidth utilization. For example, if a node only has packets to be sent to one neighbor, it cannot reuse the time slots scheduled to other neighbors.

Woo and Culler [17] examined different configurations of carrier sense multiple access (CSMA) and proposed an adaptive rate control mechanism, whose main goal is to achieve fair bandwidth allocation to all nodes in a multi-hop network. They have used the motes and TinyOS platform to test and measure different MAC schemes. In comparison, our approach does not promote per-node fairness, and even trades it off for further energy savings.

Finally, we look at some work on low-duty-cycle operation of nodes, which are closely related to S-MAC. The first example is Piconet [14], which is an architecture designed for low-power ad hoc wireless networks. Piconet also puts nodes into periodic sleep for energy conservation. However, there is not any coordination and synchronization among neighboring nodes about their sleep and listen time. The scheme that Piconet uses to enable the communications among neighboring nodes is to let a node broadcast its address before it starts listening. If a sender wants to talk to a neighboring node, it must wait until it receives the neighbor's broadcast. In contrast, S-MAC tries to coordinate and synchronize neighbors' sleep schedules to reduce latency and control overhead.

Perhaps the power save (PS) mode in IEEE 802.11 DCF is the most related work to the low-duty-cycle operation in S-MAC. Nodes in PS mode periodically listen and sleep, just like that in S-MAC. The sleep schedules of all nodes in the network are synchronized together. The main difference to S-MAC is that the PS mode in 802.11 is designed for a single-hop network, where all

nodes can hear each other, simplifying the synchronization. As previously observed by [21], in multi-hop operation, the 802.11 PS mode may have problems in clock synchronization, neighbor discovery and network partitioning. In fact, the 802.11 MAC in general is designed for a single-hop network, and there are questions about its performance in multi-hop networks [22]. In comparison, S-MAC is designed to operate in a multi-hop network, and does not assume that all nodes are synchronized together. Finally, although 802.11 defines PS mode, it provides very limited policy about *when* to sleep. Whereas in S-MAC, we define and measure a complete system.

Tseng *et al.* [21] proposed three sleep schemes to improve the PS mode in the IEEE 802.11 for its operation in multi-hop networks. Among them the one named periodically-fully-awake-interval is the most closest to the scheme of periodic listen and sleep in S-MAC. However, their scheme does not synchronize the sleep schedules of any neighboring nodes. The control overhead and latency can be large. For example, to send a broadcast packet, the sender has to explicitly wake up each individual neighbor before it sends out the actual packet. Without synchronization, each node has to send beacons more frequently to prevent long-time clock drift.

III. S-MAC DESIGN OVERVIEW

S-MAC includes approaches to reduce energy consumption from all the sources that we have identified to cause energy waste, *i.e.*, idle listening, collision, overhearing and control overhead. Before describing the components in S-MAC, we first summarize our assumptions about the wireless sensor network and its applications.

Sensor networks will be composed of many small nodes to take advantage of physical proximity to the target to simplify signal processing. The large number of nodes can also take advantage of short-range, multi-hop communications to conserve energy [4]. Most communications will be between nodes as peers, rather than to a single base-station. Because there are many nodes, they will be deployed casually in an ad hoc fashion, rather than carefully positioned. Nodes must therefore self-configure.

In-network processing is critical to sensor network lifetime [5], [6]. Since sensor networks are committed to one or a few applications, application-specific code can be distributed through the network and activated when necessary or distributed on-demand. Techniques such as data aggregation can reduce traffic, while collaborative signal processing can reduce traffic and improve sensing quality. In-network processing implies that data will be processed as whole messages at a time in store-and-forward fashion, so packet or fragment-level interleaving from multiple sources only increases overall latency.

Finally, we expect that applications will have long idle periods and can tolerate some latency. In sensor networks, the application such as surveillance or monitoring will be vigilant for long periods of time, but largely inactive until something is detected. For such applications, network lifetime is critical. These classes of applications can often also tolerate some additional latency. For example, the speed of the sensed object places a bound on how rapidly the network must detect an object.