

On Channel Adaptive Energy Management in Wireless Sensor Networks

Xiao-Hui Lin and Yu-Kwong Kwok

Department of Electrical and Electronic Engineering
The University of Hong Kong, Pokfulam Road, Hong Kong
Corresponding Author: Yu-Kwong Kwok (Email: ykwok@hku.hk)

Abstract—Energy constraints in a wireless sensor network are crucial issues critically affecting the network lifetime and connectivity. To realize true energy saving in a wireless environment, the time varying property of the wireless channel should also be taken into account. Unfortunately, this factor has long been ignored in most existing state-of-the-art energy saving protocols. Neglecting the effects of varying channel quality can lead to an unnecessary waste of precious battery resources, and, in turn, can result in the rapid depletion of sensor energy and partitioning of the network.

In this paper, we propose a channel adaptive energy management protocol, called CAEM, that can exploit this time varying nature of the wireless link. Specifically, CAEM leverages on the synergistically cross-layer interaction between physical and MAC layers. Thus, each sensor node can intelligently access the wireless medium according to the current wireless link quality and the predicted traffic load, to realize an efficient utilization of the energy. Extensive simulation results indicate that CAEM can achieve as much as 40% reduction in energy dissipation compared with traditional protocols without channel adaptation.

Index Terms—mobile computing, wireless sensor networks, power saving, channel state dependent, adaptive cross-layer protocols.

I. INTRODUCTION

The advances in MEMS (micro-electronic-mechanical systems) based sensor technology, coupled with low-power, low-cost digital signal processors (DSPs) and radio frequency (RF) circuits have spurred the proliferation of wireless sensor networks in a wide spectrum of civilian and military applications, such as environment monitoring, battlefield surveillance, and home networking, for collecting, processing and disseminating wide ranges of complex environmental data [1], [4], [6], [8]. In most cases, a sensor network is composed of hundreds or thousands of inexpensive and portable devices, baring formidable sensing, computing and wireless communication capabilities, and this collection of tiny sensors can form an autonomous and robust data computing and communication distributed system for automated information gathering and distributed sensing.

Such a sensor network can be deployed in remote areas or hostile terrains, without the infrastructure support from the outside world [25]. Normally, a sensor node in the network is battery powered, equipped with low data rate wireless radio, and with limited memory and computation ability. This exerts serious physical constraints on the application of single sensor, and thus, all the sensor nodes in the group must collaborate together to form a dense, fault tolerant network so as to realize an efficient utilization of the precious network resources—wireless channel, memory and battery energy.

Among all these constraint factors, the most crucial one is the energy consumption, which is also the most substantial challenge facing the designer of small but long-lived sensor nodes. Because in many application scenarios, a myriad of sensor nodes are spread across a large geographical area. For example, the sensors are deployed in a forest or battlefield, to gather data and monitor the environmental changes [1],

[31]. These sensor nodes are driven by their limited carried battery energy and are expected to work for months.

Efficient energy management involves all levels of the sensor system hierarchy, from hardware to software architecture, and from operating system to communication protocols. Indeed, all the system components can critically influence the energy dissipation, depending on the applications. Thus, energy-awareness must be incorporated into every level of the system design and operation to maximize the network lifetime and connectivity [5], [11], [18], [23], [28]. Network resiliency is necessary to provide flexibility to strike a balance between energy and system performance. For example, in the system operation, the computation energy can be traded off for output quality according to varying environmental conditions [15], [17], [26]: in dynamic voltage scaling (DVS) [24], we can actively and adaptively adjust the supply voltage in conjunction with the clock frequency, in response to the fluctuations in a CPU's utilization; at application layer, we can use different keys of varying length, thus allowing trade-off between expended computation energy and data security [30]; through the design of operating system for sensors, we can also let different components of the node enter various states (active/idle/sleep) to save energy according to the environmental variations, at the expense of some degree of system performance degradation—latency overhead, long response time, etc. [21].

All the above-mentioned research efforts are focused on the system computation aspect. In a wireless sensor node, the main energy consumers include: sensing unit, computation unit and communication unit. In fact, for an embedded system, energy consumed by the communication component for radio transmission dominates that by the computation counterpart. For example, the energy expended in a sensor node by Rockwell Inc. for transmitting 1 bit is around 2000 times of that for executing one instruction [19], [27]. Thus, this fact also provides large room for energy efficiency enhancement, and has attracted much attention from researchers for further exploitation. In [24], [20], by adopting the similar idea of DVS, a technique of dynamic modulation scaling (DMS) is proposed. Specifically, DMS can adaptively change the modulation level to lower the overall energy consumption, according to the number of queued packets in the system, while bounding the packet delay at an acceptable level. In [19], DMS is combined with packet fair queuing algorithm, resulting in an energy-efficient packet scheduling protocol.

However, in dealing with energy management problem, one major drawback in the above existing state-of-the-art protocols is that the time varying nature of the wireless channels among the sensor terminals is ignored, let alone exploited. Under hostile deployment circumstances, reliability of data transmission can be reinforced by either increasing the transmission power level or adding forward error correction (FEC) to the raw data [3]. The first method is infeasible because doing that can rapidly deplete the limited sensor energy, which is expected to sustain for months. Moreover, this may also introduce interference to the wireless transmission at other terminals. Her

This research was supported by a grant from the Research Grants Council of the HKSAR under project number HKU 7157/04E.

have to resort to FEC to overcome this unreliable link problem. As the channel quality changes with time, the amount of incorporated error protection should also vary with the instantaneous channel condition to make sure the bit error rate (BER) is above the required level. Therefore, the poorer the wireless link, the more amount of error protection redundancy in the transmitted packet, and vice versa [13].

In overcoming the wireless link fluctuation problem, the introduction of FEC can incur extra energy dissipation, which is unavoidable in combating the adverse channel condition. The power consumption sources include the following two major aspects.

- 1) From the computation point of view, packet redundancy can lead to additional expended energy that goes into encoding and decoding data at two communication sides. The needed energy is drawn from the limited battery source, and thus, should be taken into consideration in power management.
- 2) With respect to the packet transmission, extra energy will also be incurred during the message communication, as the length of every frame will increase after the error protection is included. This means that if the raw transmission rate (different from actual link throughput in the latter parts) remains the same, the radio circuits (transceiver, receiver, output amplifier, synthesizer and PLL/VOC, etc.) will be on for a longer duration, and thus, will consume more energy.

The above two aspects, especially the second one, have made the design of energy resource management scheme in wireless sensor networks very challenging [7]. Because now the quality of the wireless channel is a time varying function, or the energy needed in transmitting one useful information bit fluctuates with time, how to manage the energy resource at a network system level is extremely crucial in the sense that a judicious management scheme can enhance the precious battery resource utilization and maximize the network lifetime. This motivates us to design an energy-aware packet scheduling scheme for the sensor network under a channel fluctuating environment.

In this paper, we present a channel adaptive energy management algorithm, namely, CAEM, for efficient packets scheduling and queuing in a sensor network, with time varying characteristic of wireless channel taken into consideration. Specifically, due to the realistic fact that a wireless link with worse channel quality can result in more energy expenditure (more error protection redundancy incorporated), we can buffer the packet temporarily until the channel quality recovers to the required threshold. To implement this basic idea in the real situation, we propose a network system in which each sensor can decide the state of its communication component (active/sleep/idle) with respect to the current link condition.

Nevertheless, packet buffering may lead to communication latency and buffer overflow, and this is unfair to the sensor node with bad channel quality. Therefore, to avoid this, we also design a fair scheduling and queuing algorithm to ensure that every sensor can equally access the wireless channel under such a fluctuating environment, i.e., a sensor node is never starved from transmission even its communication link is bad. Thus, we can achieve a balance between energy efficiency and fairness. Our proposed system and scheduling scheme can be applicable in enhancing energy awareness in many typical existing sensor networks, which consist of sensor nodes, sinks (for data gathering and aggregation) and base station (for transmitting processed data to the outside world).

The rest of this paper is organized as follows. In Section II, we briefly give some background on the sensor network architecture, and then we present a realistic wireless channel model. In Section III, we discuss how to incorporate channel adaptation into energy management, with many related factors taken into consideration, such as channel condition and predicted traffic load. In the same section, we also present two channel adaptive energy conservation schemes. We give

the simulation results in Section IV together with our interpretations. Section V concludes the paper.

II. BACKGROUND

A. Network System Architecture

To enable the scalability and energy efficiency in a sensor network composed by hundreds or thousands of sensor nodes, a cluster based hierarchy is an ideal solution. As the data collected by sensors in vicinity is highly correlated, and the communication between each sensor and the end-user can be very energy and bandwidth consuming, the data should be processed locally to get rid of data redundancy. With this approach, the whole network can be divided into clusters, and in each cluster a sensor node is elected as the *cluster head* to perform local information filtering, data aggregation and data forwarding for the sensors in that cluster. The group of cluster heads elected from the entire network in turn forms a subnetwork, among which the traffic load is routed. Therefore, a cluster based network organization eases the network management and reduces the energy needed for communicating useful data to the end-user. There are multiple methods for the organization of a cluster based sensor network [2], [9], [10], [22], and the management of such network hierarchy is beyond the scope of this paper.

Our proposed CAEM is suitable for such a cluster based sensor network, which consists of sensor nodes for data collection, sinks (cluster heads) for data reception from the sensors and information aggregation, and base station for the wireless connection to the outside world. Figure 1 is an illustration for such sensor network architecture, which is also a typical operational scenario of sensor networks in reality for remote data gathering.

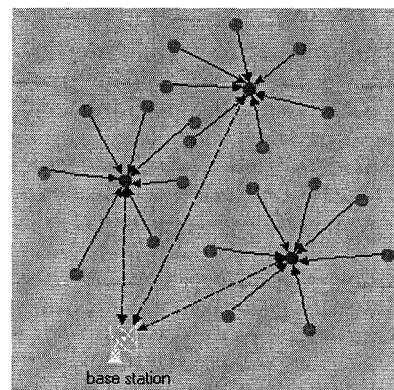


Fig. 1. A hierarchically clustered sensor network.

Different from conventional cellular or ad hoc networks in which provision of high QoS (high network throughput, low packet delay) or terminal mobility management is of higher priority, we are more concerned with the network lifetime in a sensor network. A major reason is that in cellular/ad hoc networks, the terminal's battery can be recharged whenever needed. However, energy replenishment is infeasible for a sensor terminal which works in a distant or hostile area. The depletion of the battery energy means the failure of the node and partial partitioning of the network, resulting in the "blind area" of the corresponding location.

B. Realistic Channel Model

Using a channel adaptive physical layer is one of the major distinctive features of our approach in contrast to existing work in sensor energy saving that uses simple channel model in which th

varying nature is neglected. In practice, the wireless channel between every two sensor terminals is time varying and is governed by physical effects in signal propagation: path loss, shadowing, and microscopic fading [7], [12]. Path loss refers to the change in received signal strength versus the distance between the transmitter and the receiver. Shadowing loss refers to the change in received signal strength due to variations in terrain structure and transmission conditions. These two factors fluctuate in macroscopic time scale (2–5 seconds). On the other hand, microscopic fading refers to the variation of signal strength due to multipath propagation and, as such, fluctuates in microscopic time scale.

In this paper, we assume that sensor nodes are static or in a low mobility (less than 1 Km/hr) and, therefore, the coherence time of the fading channel is of the order of 80–100 ms. The combined effect of the above signal propagation factors is characterized by the CSI (Channel State Information), which is the measured SNR (signal-to-noise ratio) of known pilot symbols on a feedback channel (or signaling channel in this paper). Since a packet or physical frame duration in our system is around several milliseconds, it is justified for us to assume that the CSI remains approximately constant for the duration of at least one frame.

In this paper, we adopt the adaptive physical layer design called ABICM [13], in which variable throughput modulator and channel coding are used. Specifically, when CSI is available at the transmitter, the transmitter performs “burst-by-burst” throughput adaptation [7] with respect to the CSI. For instance, when the CSI indicates that the channel is of a good quality, the transmitter employs a high-order modulation (e.g., 16QAM) and high-rate error correction code so as to boost the instantaneous throughput. On the other hand, when the channel quality is poor, the transmitter employs a lower order modulation (e.g., BPSK) and low-rate error protection so as to protect the packet transmission at the expense of lower instantaneous throughput.

Indeed, the salient concept of adaptive physical layer has been widely deployed in various wireless systems such as 3G1x, EV-DV, UMTS, HSDPA, and IEEE 802.11a/g. Note that using ABICM in our study is just for illustration only, and other schemes (e.g., the one suggested in [7]) can also be used. For details of the ABICM scheme and its applications in MAC protocols, the reader can refer to [13].

In our study, we use a 4-mode ABICM configuration [13] and, thus, there are four distinct possible throughput levels: 2 Mbps, 1 Mbps, 450 kbps, and 250 kbps, respectively (after adaptive channel coding and modulation).

III. CHANNEL ADAPTIVE ENERGY MANAGEMENT — CAEM

A. System Model and Tone Signaling

The communication component in a sensor node is the largest source of energy consumer, thus when there is no data transmission, the radio should enter a sleep mode to save energy. Besides this mechanism, as mentioned above, the wireless channel also has a great influence on the energy expenditure in packet transmission, i.e., packet transmission through a link of high quality consumes less energy than that needed through a “bad” link. Based on this observation, in CAEM, each sensor node should possess the ability to decide the state of its communication unit with respect to the current condition of the wireless link between it and the sink (cluster head). Thus, a real-time monitoring of the change of the CSI of the wireless link is necessary for a sensor node.

In a wireless cellular network, the base station can instruct the mobile terminal to continuously adjust the error protection in the uplink channel, according to the signal strength or BER of the received packets, through a dedicated signaling control channel [3]. However, this is impractical in implementation for a small-size, low-cost, and energy-constrained sensor node, due to at least three reasons: (1) the wireless bandwidth is very scarce, and such signaling control channel can

reduce the available radio resource; (2) additional signaling channel increases the firmware complexity at both sides, as we need to consider problems such as encoder/decoder, synchronization, frequency synthesis, etc., for this signaling channel; and (3) apart from problems mentioned above, the cost, size, and energy consumption of the sensor will also increase.

In a wireless sensor network, we assume that all nodes in the cluster use the same common data channel for packet transmission, and the traffics are from sensors to the sink (cluster head) for information processing. This is a typical operational scenario of sensor networks for data collection and fusion. A sensor does not need to keep its radio on or keep listening to the channel, when there is no packet for transmission. Instead, it can let the radio enter a sleep mode to save energy. Thus, a distributed MAC control protocol like IEEE 802.11 is unsuitable for such a sensor system. Because in such a system, each terminal may possibly act as a router for any surrounding neighbor, and therefore, it has to keep overhearing the channel as it does not know when the next transmission is going to take place. The simplicity of the traffic mode (from sensor to sink only) and the cluster hierarchy of the sensor network lead to a simpler design in MAC layer management. The organization of a sensor cluster is somewhat like a cell system in cellular networks. Consequently, inspired by the signaling channel used in cellular networks, we can adopt a much more simplified tone channel as the CSI and channel state (receive/collision/idle) indicator for the medium access control of each sensor node. Mathematically, the signal from this tone channel can be expressed as:

$$T(t) = A \sum_{n=1}^K g(t - nT_p) \quad (1)$$

where A is the amplitude of the signal, $g(t)$ is a pulse signal with period of T_p , and $K = T_s / (T_a + T_b)$ decides the length of the duration of tone signal. Specifically, we can express $g(t)$ in one period as:

$$g(t) = \begin{cases} 1 & 0 \leq t < T_a \\ 0 & T_a \leq t < T_p \end{cases} \quad (2)$$

Figure 2 illustrates the pulse series in the tone channel. Note that the distance between two adjacent pulses can be tuned to identify the current channel state, and the related parameters are listed in Table I.

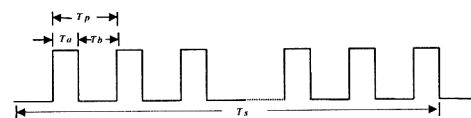


Fig. 2. Pulse series in the tone channel.

TABLE I
USING DIFFERENT PULSE INTERVALS TO IDENTIFY CHANNEL STATES
(TONE DURATION IN MSEC; T_a , T_b , AND T_p IN μ SEC)

Collision	0.5	3	5	8
Receive	0.5	3	10	13
Transmit	0.5	3	15	18

In a cluster, all the sensor nodes use the same data channel for packet transmission. The states of the data channel include: (a) receive-

sink is receiving data packets from a node in the cluster; (b) transmit—the sink is sending processed data to the base station (we do not consider this in this paper at this stage); (c) idle—no packet is being received or transmitted, and the data channel is free; (d) collision—more than two nodes are using the data channel to transmit data at the same time, causing packet collision at the cluster head. The status of the data channel can be expressed by different pulse intervals using Equation (1).

Therefore, the interval variation in the tone channel acts as a signal indicating whether a sensor node can access the data channel or not. As harmonization of medium access among sensor nodes is very important, we specify that: (1) when the data channel is free, the cluster head must periodically broadcasts idle tone pulse series, with a period of 50 ms, and the cluster head should send a succession of tone pulses with duration of 1 ms; (2) to help the sending sensor to adaptively adjust the amount of error protection according to the real-time variation of the wireless channel, on detecting packet coming, the cluster head should send out receive tone pulses with duration of 0.5 ms for every 10 ms; (3) the cluster head only sends out collision tone pulses once with duration of 0.5 ms, on detecting packet corruption; (4) once the data channel recovers to the idle state, the cluster head broadcasts idle tone pulses.

B. Medium Access Control in CAEM

A sensor node has two radio sets: tone radio and data radio, working at different frequencies. Both radios should be off to save energy if the sensor has no packet to transmit. When the sensor has packets to send to its cluster head, it first turns on its tone radio, sensing the channel to see whether it is free or not. If negative (receives tone pulses from the cluster head with state other than idle, such as receive or collision), it keeps monitoring the tone channel. If it senses that the data channel is free (receives idle tone pulses), it should further measure the received tone signal and check whether it is above the required SNR requirement. If not, it continues monitoring the tone channel; otherwise, it backs off for a random period of time, which equals to $2^{i-1} \times \text{rand}(0, 1) \times 20 \times cw$, where $\text{rand}(0, 1)$ is a random number generator that generates a number evenly distributed between (0,1), i is the number of times this packet has been retransmitted (the maximal value is 6), and cw is the contention window size.

After this period of back-off time, the sensor again checks whether the channel is free and its quality requirement is still satisfied. If both conditions are positive, the sensor can now turn on its data radio and transmit its buffered packets. If either of the two conditions is not satisfied, the sensor node returns back to the sensing state, and keeps monitoring the channel. Packet collision can happen if more than two sensors transmit at the same time. In case of packet collision, the cluster head sends out collision tone pulses, notifying the related sending nodes this situation. Note that a sensor node should keep its tone radio on during data packet transmission, and on receiving collision tone pulses, it stops packet transmission by turning off the data radio, and returns to the sensing state. This is different from CSMA/CA medium access control mechanism adopted in other mobile computing systems such as an 802.11 wireless LAN in which the terminal cannot transmit and receive at the same time because there is only one radio system equipped, and hence it is incapable of collision detection and has to use "collision avoidance" to reduce the chance of packet collision. In CAEM, however, we can employ CSMA/CD to detect collision with the assistance of the tone channel, thereby reducing energy wasted in packet collision.

As a cluster head broadcasts idle tone pulses periodically when the data channel is free, it can cost some time for a sensor to track this tone signal, thus incurring some delay overhead. After finishing packet

transmission, a sensor can turn off the data and tone radios, and enter a sleep state. In case a cluster head collapses or switches (as described in LEACH [10]), causing the sensors in the cluster unable to receive tone signals, a sensor should power both radios off and enter a sleep state. The sensor state transition diagram is illustrated in Figure 3. Similarly, based on the data channel states described above, the cluster head should broadcast different tone pulse signals accordingly, to notify sensor nodes in the cluster the current channel state. And this state transition diagram is depicted in Figure 4.

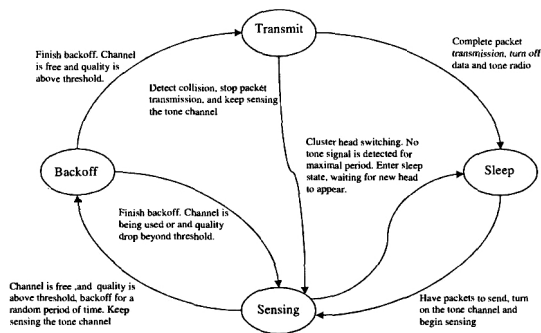


Fig. 3. State transition diagram of a sensor node.

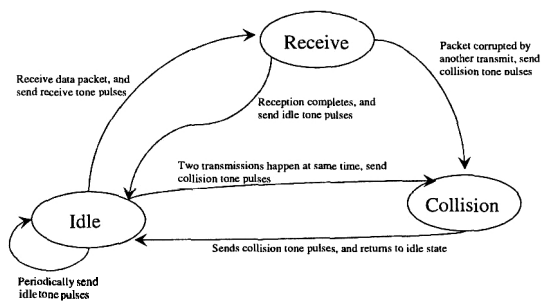


Fig. 4. State transition diagram of a cluster head.

Compared with the signaling channel in a cellular network, the tone channel for a sensor network has the following superior features: (1) ease of implementation—as it does not need any complicated auxiliary circuits such as multiplexing/demultiplexing, encoding/decoding and synchronization units, and thus, the size and cost will not increase considerably; (2) energy efficient—the cluster head only periodically broadcasts idle tone pulses with a short time duration if the data channel is free, or sends out collision/receive tone pulses when the data channel is occupied, with a low-power tone radio; (3) the open ISM frequency spectrum provides the flexibility in network deployment, without the strict spectrum constraint as in cellular networks.

Our proposed CAEM scheme is based on the following assumptions:

- 1) The tone signaling channel has no interference with the common data channel. And these two channels share the same propagation characteristics, i.e., have the same attenuation and fading parameters.
- 2) The propagation conditions (attenuation, fading, etc.) between the source and destination terminals is assumed to be the same in both directions, or the propagation gains in both directions are the same, i.e., $G_{ij} = G_{ji}$.
- 3) The channel gain remains stationary for the duration of a packet transmission.

A summary of our CAEM scheme is described in order. By measuring the attenuation of the received tone signal, each sensor car

tinuously monitor the CSI change of the data channel, thus can: (1) energy-efficiently schedule its data transmission based on the current channel condition; and (2) adjust the incorporated error protection as needed in the data channel. At the same time, from the tone pulse interval, each sensor node in the cluster can also know the current state of the data channel.

C. Adaptive Threshold Adjustment

In our study, based on the channel quality and after using a 4-mode ABICM configuration, there are four distinct possible throughput levels: 2 Mbps, 1 Mbps, 450 kbps, and 250 kbps, respectively (after adaptive channel coding and modulation). As mentioned above, transmitting packets through wireless link with possibly the highest quality can achieve energy efficiency from at least two aspects: (1) packet transmitting and receiving time is reduced (the operating time duration for radio circuits is shortened); and (2) incurred encoding/decoding energy overhead at both communication sides is also decreased. Obviously, throughput level of 2 Mbps is the most desired energy saving mode, and therefore, we can temporarily buffer packets until the link quality is above this threshold.

However, this packet buffering scheme favors only the sensor nodes with highest link quality, as they can get most bandwidth shares, and those nodes with link quality “not so good” may get starved. As a result, an unlucky node has to wait until its channel quality recovers, even it has many packets waiting in the buffer. This unfairness can lead to a very serious problem at nodes with link quality below thresholds—packet overflow and long queuing delay, and thus, it can further result in loss of gathered data or temporal interruption of monitoring in some areas. This problem is aggravated by the fact that the channel quality is a time varying function, and the link quality cannot always satisfy the highest threshold of 2 Mbps, under a harsh working environment, leading to instability in network performance.

An instinctive solution to this problem is to relax the threshold constraint by adaptively adjusting the threshold according to the current traffic load and buffers queue length, and thus, nodes with different wireless link quality can fairly have equal opportunities to get deserved amount of transmission slots. In most operation scenarios, there is a trade-off between the energy efficiency and network performance, and thus, we should achieve a balance point in between, i.e., although the energy conservation is the first priority in our study, a smooth gathered data flow from a particular observing sensor is also needed to keep necessary real-time surveillance on the related area [16]. This motivates us to devise a suitable adaptive threshold adjustment scheme, combined with the above channel adaptive energy management, to realize an efficient energy utilization, while reducing buffer overflow to the best effort. In addition, this adjustment scheme should be fully distributed, low complexity in computation to conserve energy. Thus some centralized or complicated packet scheduling algorithms such as round robin or WFQ (weighted fairness queuing) [19] are not suitable for a sensor network.

We assume that an event occurs and is observed by a sensor. The sensor collects data and generates a packet at time T_K . Without loss of generality, the next packet generating time can be expressed as T_{K+1} . Thus, the series of time instants $\{T_K, T_{K+1}, T_{K+2}, T_{K+3}, \dots\}$ denotes the arrival times of a sequence of packets as illustrated in Figure 5.



Fig. 5. Packet arrivals.

We further denote $q(T_K)$ as queue length of the buffer of a sensor node at time instant T_K . To prevent buffer overflow, a real-time monitoring of the instantaneous queue length $q(T_K)$ and its variation is necessary to predict the future traffic load. However, this should not be executed for every incoming packet, as doing so can lead to great computation overhead. The sampling interval should be fixed at S , i.e., for every S incoming packets (in our simulation, we let $S = 5$). Now we get a sequence of queue lengths $\{q(T_K), q(T_{K+S}), q(T_{K+2S}), q(T_{K+3S}), \dots\}$, sampled at time instants $\{T_K, T_{K+S}, T_{K+2S}, T_{K+3S}, \dots\}$. Therefore, the variation of the queue length can be defined as:

$$V(T_{K+iS}) = q(T_{K+iS}) - q(T_{K+(i-1)S}) \quad (3)$$

where $i = 1, 2, \dots$. For instance, at time T_{K+2S} , we compare $V(T_{K+2S})$ and $V(T_{K+S})$ by $\Delta V = V(T_{K+2S}) - V(T_{K+S})$. Thus, ΔV reflects the instantaneous queue variation tendency and can be used as a traffic load predictor, i.e., if $\Delta V \geq 0$, the queue length has an increasing tendency; otherwise, if $\Delta V < 0$, the queue length is likely to decrease.

Based on the above analysis, now we can devise a simple adaptive threshold adjustment strategy: at time T_{K+2S} , if $\Delta V \geq 0$, to avoid overflow, we should lower the transmission threshold for one class (there are 4 classes/thresholds corresponding to 4 throughput levels), to let this node have more chances to send its buffered packet. For example, the original channel quality threshold is corresponding to 2 Mbps, now it should be adjusted to quality threshold of 1 Mbps; if $\Delta V < 0$, we should increase transmission threshold to the highest value to save energy, for example, if the original value is 250 Kbps, then the new threshold should be 2 Mbps.

We keep monitoring the incoming traffic, and once the queue length exceeds a value l_{Th} , the threshold adjustment mechanism is started up (in our simulation $l_{Th} = 15$). The pseudo code for this threshold adjustment strategy is given in Figure 6, and the whole system architecture is shown in Figure 7.

Pseudo Code for Threshold Adjustment

```

/* Execute for every S incoming packet
Th is the current transmission threshold. Corresponding to different channel
quality, there are four transmission thresholds, from the highest to the lowest:
th1, th2, th3, th4.
*/
At each (K+NS)st packet arrival epoch do {
  compute V(T_{K+NS}) = q(T_{K+NS}) - q(T_{K+(N-1)S});
  if (q(T_{K+NS}) <= l_{th})
    null;
  else {
    compute ΔV = V(T_{K+NS}) - V(T_{K+(N-1)S});
    if (ΔV >= 0 && Th ≠ th4)
      lower down Th;
    else if (ΔV < 0 && Th ≠ th1)
      Th = th1;
  }
}

```

Fig. 6. Pseudo code for adaptive threshold adjustment.

IV. SIMULATIONS AND ANALYSIS

The frequent switching (on/off) of the data radio unit can incur considerable amount of startup energy and time overhead (e.g., the RFM radio needs 20 μ secs to switch from sleep mode to active mode [29]).

specify that the minimum number of packets sent for one transm

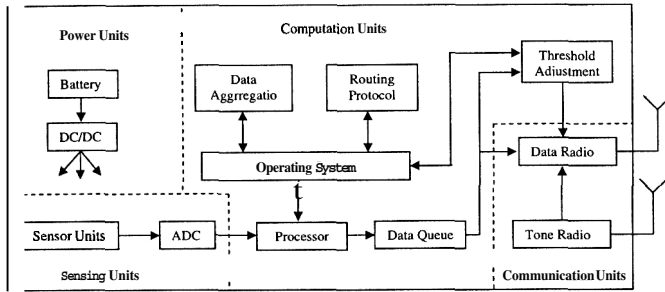


Fig. 7. System architecture for the sensor network.

is 3. And to ensure fairness among sensor nodes, the maximal number of packets sent per transmission is fixed at 8. In our study at this stage, to ease data analysis, we have not taken the energy expended in encoding/decoding into consideration, as they are negligible compared with energy cost in transmitter/receiver electronics [19].

As mentioned above, CAEM is suitable for any cluster based sensor network, as a tone signal from the cluster head is necessary. This means that CAEM can be integrated with any cluster based sensor network. In our study, we choose LEACH (Low-Energy Adaptive Clustering Hierarchy), a well-known clustering protocol for sensor network, as our reference. In LEACH, each sensor collects raw data and sends the data to a cluster head for data aggregation, thus reducing communication overhead and energy cost. The novel characteristic of LEACH is that each sensor takes turn to assume the responsibility of cluster head, consequently, no sensor nodes energy resource is unfairly over-exploited, and hence, can realize a graceful energy consumption evenly distribution in the whole network [10]. Basically, in LEACH, each sensor decides whether or not to become a cluster head (CH) for the current round, based on the predetermined percentage of nodes that can become CHs and the number of times this sensor has been CH thus far. Specifically, this sensor node n generates a random number evenly distributed between $[0, 1]$, and if the number is less than threshold, the node becomes the CH for the current round. The threshold is set as:

$$T(n) = \begin{cases} \frac{P}{1 - P \times (r \bmod \frac{1}{P})} & \text{if } n \in G \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

where P ($0 < P \leq 1$) is the desired percentage of sensor node that can become CH (in our simulation we set $P = 0.05$), r is the current round, and G is the set of nodes that have not been CH in the past rounds. To avoid communication interference among clusters, in our study we assume different clusters working in different frequency spectra. For more detailed information about LEACH, the reader can refer to [10].

A. Simulation Environment

The simulation parameters we used are listed as in Table II:

We choose pure LEACH without channel adaptiveness, and two schemes with channel adaptive as our references. These two schemes are hybrids of pure LEACH with CAEM. In Scheme 1, we have integrated adaptive threshold adjustment such that the transmission threshold can be adjusted based on the current queue length and predicted future traffic load. While in Scheme 2, the transmission threshold is fixed at the highest value, i.e., 2 Mbps for the whole simulation time in our study. The initial thresholds in both schemes are chosen as 2 Mbps. Each sensor node is a Poisson source, i.e., the generated packet follows a Poisson arrival.

To evaluate the three protocols, we increase the traffic load until the network gets saturated. We compare the protocols by using the following metrics:

TABLE II
PHYSICAL SIMULATION PARAMETERS

Parameter	Value
Testing Field	1000m x 1000m
Number of Nodes	100
Bandwidth	2, 1, 0.45, 0.25 Mbps
Percentage of CH	5%
Transmit Power for Data Channel	0.66 W
Receive Power for Data Channel	0.305 W
Sleep Power for Data Channel	3.5 mW
Transmit Power for Tone Channel	92 mW
Receive Power for Tone Channel	36 mW
Packet Length	2 Kbits
Sensing Delay	8 μ secs
Contention Window Size (CW)	10 μ secs
Buffer Size	50

- 1) **Energy Consumption and Network Lifetime:** High values of these metrics are what we desire as an efficient energy management scheme can prolong the network lifetime.
- 2) **Network Performance:** In our study we consider three performance aspects:
 - average packet delay: measured in ms, the packet delay is the time duration for a packet transmitted from its source to the sink (including queuing and transmission time);
 - aggregate network throughput: the average number of data packets arriving at their destinations per second in the whole network, measured in kbps; and
 - successful packet delivery rate: the ratio of the number of packets successfully received by sinks to the total number of packets generated.
- 3) **Standard Deviation of Queue Length:** This parameter reflects short-term fairness among sensor nodes that can get the share of wireless bandwidth.

Due to space limitations, results on network performance are not presented in this paper. For the full set of results, the reader is referred to the long version of this paper [14].

B. Energy Consumption and Network Lifetime

Figure 8 shows the average remaining energy in each sensor with elapsed time. The initial battery energy level is 10 Joules, and traffic load is fixed at 5 packets/second for each sensor. We can see that a judicious channel adaptive power management scheme can greatly reduce energy consumed, in that two CAEM LEACH schemes outperform pure LEACH without channel adaptive in battery efficiency, as sending packets through link of high quality can considerably shorten the packet transmit/receive time in both communication sides. It is evident that a CAEM LEACH without adaptive threshold adjustment (Scheme 2) is the most efficient in battery consumption because in this scheme, the transmission threshold is fixed at the highest value (2 Mbps in simulations) and the traffic load and buffer overflow are not considered. Thus, the packet transmission time is the shortest among the three protocols. However, this can lead to serious deterioration in network performance as will be illustrated in latter sections.

The energy consumption rate can directly influence a sensor node's lifetime, as depletion of battery resource will cause the failure of a sensor. Thus, we are also interested in the number of sensor nodes alive versus elapsed time. The results are presented in Figure 9. The simulation environment is the same, i.e., initial energy of 10 joules

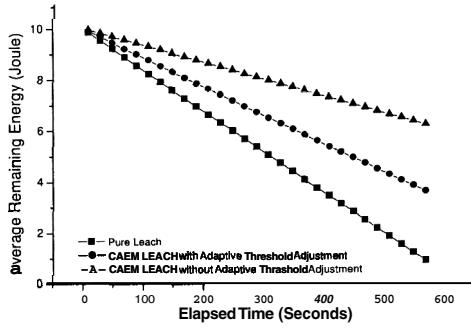


Fig. 8. Average remaining power versus time.

traffic load of 5 packets/sec for each sensor node. As all protocols are based on LEACH in which all the sensors rotate to take responsibility for being the cluster head and no particular sensor is unfairly exploited in battery consumption, thus all the curves in the figure drop abruptly at some critical points. This means that the time duration between the first exhausted node and the last one is quite short, or the difference in energy level from node to node does not vary greatly. We again observe the same results—CAEM LEACH schemes can extend a sensor node’s lifetime. We further call a network “dead” if the percentage of nodes exhausted exceeds 10%, the two schemes can prolong the network lifetime for about 40% and 130% for Scheme 1 and Scheme 2, respectively.

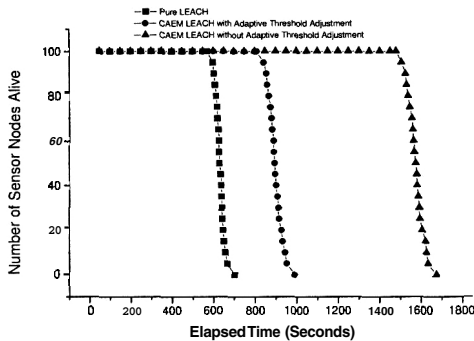


Fig. 9. Number of nodes alive versus time.

Based on the network lifetime defined above, we further increase the traffic load until the network gets saturated and measure the network lifetime. The results are shown in Figure 10, in which the labeled “Added Traffic Load” means the number of packets generated by each node per second. As expected, all the curves drop with the increase of traffic load, as more packet transmissions can speed up a sensor’s energy consumption, and further decrease network lifetime. Again, due to the same reason, Scheme 2 can achieve the longest network lifetime. The other channel adaptive scheme, Scheme 1, however, has to adjust the transmission threshold accordingly to adapt to the current queue length and the predicted traffic load. Therefore, as the network gets saturated, the difference of network lifetime between Scheme 1 and pure LEACH becomes invisible. The reason is that since Scheme 1 has to tune transmission threshold to the lowest value (250 Kbps) in each sensor for a considerable amount of time, it thus becomes a “non-channel adaptive” protocol like pure LEACH, in which the channel quality is ignored.

C. Energy Efficiency and Fairness

One important metric in measuring the energy efficiency is average energy consumed for successfully transmitting one data packet.

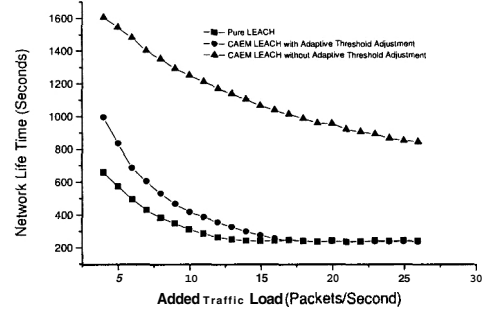


Fig. 10. Network lifetime versus traffic load.

Certainly Scheme 2 is the highest in energy efficiency as packets are always sent through the link with the best quality. Thus, we only compare pure LEACH and Scheme 1 and the results are illustrated in Figure 11. It is shown that after the channel quality is taken into consideration, we can achieve about 30-40% of energy consumed in sending one packet. In pure LEACH, the average energy expended decreases with traffic load, as sending more packets per transmission can reduce the radio startup energy overhead, thus enhancing the energy efficiency. When network traffic load is increased, Scheme 1, however, has to lower its transmission threshold more frequently, resulting in more energy consumed for each packet transmitted. The difference in average energy consumed for these two schemes will decrease if we further increase traffic load.

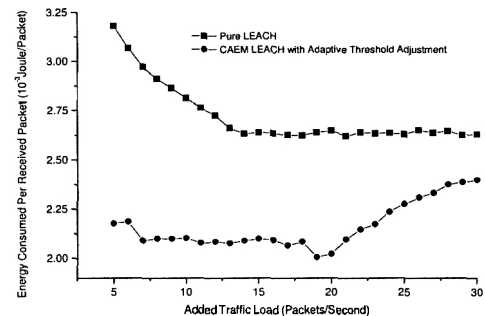


Fig. 11. Average amount of energy consumed versus traffic load.

Although sending packets through a link of high quality by temporarily keeping them in buffer can conserve energy, it can also incur packet delay or overflow. Another serious problem is that this can lead to short-term unfairness to those sensor nodes with “bad” channel conditions, causing long queues at these nodes, as the threshold requirement prohibits them from any packet transmission. This problem is more serious with Scheme 2, whose threshold is fixed at the highest value. To overcome this problem, in Scheme 1, we have adopted an adaptive transmission threshold adjustment in which the threshold can be tuned based on the queue length and the predicted incoming traffic load. In wireless packet scheduling, there are many metrics for “fairness” definition. Here, as all sensors are homogeneous Poisson sources bearing the same packet arrival rate, we can define “fairness” here as the standard deviation of queue length:

$$\delta = \sqrt{\frac{1}{N} \sum_{i=1}^N q_i - \left(\frac{1}{N} \sum_{i=1}^N q_i \right)^2} \quad (5)$$

where N is the number of sensor nodes, and q_i is the queue length for a sensor node. In our simulations in this aspect, we have set the buffer size to be substantially large enough to accommodate all gen

packets (to have an accurate calculation). Therefore, this parameter can reflect instantaneous difference in the service share each sensor can achieve. We call this “short-term fairness”, as the smaller the value, the higher the level of fairness is the protocol. In our simulations, we have taken several snapshots of the value during the observed time, and average them. The results are plotted in Figure 12. We can see that Scheme 1 exhibits a higher level of fairness in bandwidth allocation.

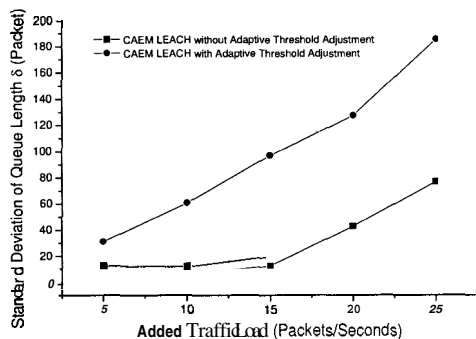


Fig. 12. Standard deviation of queue length versus traffic load.

V. CONCLUSIONS

In this paper, we have taken the time varying nature of wireless channel into account in the optimization of energy management in a sensor network. Through extensive simulations, we have demonstrated that the behavior of wireless channel can greatly influence the network energy consumption. Specifically, we have proposed two schemes, channel adaptive energy management scheme with and without threshold adjustment (Scheme 1 and Scheme 2). These two schemes can prolong network lifetime and realize a trade-off between energy and communication quality.

We have seen that the two proposed schemes can have different improvement gains in energy efficiency, and have different degrees of degradation in communication quality. It is therefore hard to judge which one is superior to the other. This depends on many environmental factors, such as operation scenarios, specific data types (time-sensitive or not), sensor density, and data redundancy, etc. Thus, more research work needs to be done in the future to find the respective application scenarios for the two schemes, with all the related factors taken into consideration.

REFERENCES

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A Survey on Sensor Networks,” *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, Aug. 2002.
- [2] S. Bandyopadhyay and E. J. Coyle, “An Energy Efficient Hierarchical Clustering Algorithm for Wireless Sensor Networks,” *Proc. INFOCOM 2003*, vol. 3, pp. 1713–1723, Apr. 2003.
- [3] G. Bianchi and I. Tinnirello, “Channel-Dependent Load Balancing in Wireless Packet Networks,” *Wireless Communications and Mobile Computing*, vol. 4, no. 1, pp. 43–53, 2004.
- [4] P. Bonnet, J. Gehrke, and P. Seshadri, “Querying the Physical World,” *IEEE Personal Communications*, vol. 7, no. 5, pp. 10–15, Oct. 2000.
- [5] J. Carle and D. S. Ryl, “Energy-Efficient Area Monitoring for Sensor Networks,” *IEEE Trans. Computer*, vol. 37, no. 2, pp. 40–46, Feb. 2004.
- [6] C.-Y. Chong and S. P. Kumar, “Sensor Networks: Evolution, Opportunities, and Challenges,” *Proc. IEEE*, vol. 91, no. 8, pp. 1247–1256, Aug. 2003.
- [7] E. Cianca, A. Luise, M. Ruggieri, and R. Prasad, “Channel-Adaptive Techniques in Wireless Communications: An Overview,” *Wireless Communications and Mobile Computing*, vol. 2, no. 8, pp. 799–813, 2002.
- [8] H. Gharavi and K. Ban, “Multihop Sensor Network Design for Wide-Band Communications,” *Proc. IEEE*, vol. 91, no. 8, pp. 1221–1234, Aug. 2003.
- [9] A. Hac, *Wireless Sensor Network Designs*, John Wiley & Sons, 2003.
- [10] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-Efficient Communication Protocol for Wireless Microsensor Networks,” *Proc. 33rd Annual Hawaii International Conference on System Sciences*, pp. 1–10, Jan. 2000.
- [11] X. Hong, M. Gerla, and R. Bagrodia, “The Mars Sensor Network: Efficient, Energy Aware Communications,” *Proc. MILCOM 2001*, pp. 418–422, Oct. 2001.
- [12] M. Kubisch, H. Karl, and A. Wolisz, “Distributed Algorithms for Transmission Power Control in Wireless Sensor Networks,” *Proc. WCNC 2003*, pp. 558–563, Mar. 2003.
- [13] Y.-K. Kwok and V. K.-N. Lau, “A Novel Channel-Adaptive Uplink Access Control Protocol for Nomadic Computing,” *IEEE Trans. Parallel and Distributed Systems*, vol. 13, no. 11, pp. 1150–1165, Nov. 2002.
- [14] X.-H. Lin and Y.-K. Kwok, “On Channel Adaptive Energy Management in Wireless Sensor Networks,” Technical Report EEE-2005-01, Mar. 2005 (available on request).
- [15] G. Lu, B. Krishnamachari, and C. S. Raghavendra, “An Adaptive Energy-Efficient and Low-Latency MAC for Data Gathering in Wireless Sensor Networks,” *Proc. IPDPS 2004*, pp. 26–30, Apr. 2004.
- [16] D. Maniezzo, K. Yao, and G. Mazzini, “Energetic Trade-off Between Computing and Communication Resource in Multimedia Surveillance Sensor Network,” *Proc. IEEE Int’l Workshop on Mobile and Wireless Communications Network*, pp. 373–376, Sept. 2002.
- [17] R. Min, M. Bhardwaj, and S.-H. Cho, “Energy-Centric Enabling Technologies for Wireless Sensor Networks,” *IEEE Trans. Wireless Communications*, vol. 9, no. 4, pp. 28–39, Aug. 2002.
- [18] V. Raghunathan, C. Schurgers, S. Parg, and M. B. Srivastava, “Energy-Aware Wireless Microsensor Networks,” *IEEE Signal Processing Magazine*, vol. 19, no. 2, pp. 40–50, Mar. 2002.
- [19] V. Raghunathan, S. Ganeriwal, and M. Srivastava, “Energy Efficient Wireless Packet Scheduling and Fair Queuing,” *ACM Trans. Embedded Computing Systems*, vol. 3, no. 1, pp. 3–23, Feb. 2004.
- [20] C. Schurgers, O. Aberthorne, and M. B. Srivastava, “Modulation Scaling for Energy Aware Communication Systems,” *Proc. ISLPED 2001*, pp. 96–99, Aug. 2001.
- [21] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. Srivastava, “Optimizing Sensor Networks in the Energy-Latency-Density Design Space,” *IEEE Trans. Mobile Computing*, vol. 1, no. 1, pp. 70–80, Jan.-Mar. 2002.
- [22] C.-C. Shen, C. Srisathapornphat, and C. Jaikaeo, “Sensor Information Networking Architecture and Applications,” *IEEE Personal Communications*, vol. 8, no. 4, pp. 52–59, Aug. 2000.
- [23] E. Shih, B. H. Calhoun, and S. Cho, “Energy-Efficient Link Layer for Wireless Microsensor Networks,” *Proc. IEEE Computer Society Workshop on VLSI*, pp. 16–21, Apr. 2001.
- [24] A. Sinha and A. Chandrakasan, “Dynamic Power Management in Wireless Sensor Networks,” *IEEE Design and Test of Computers*, vol. 18, no. 2, pp. 62–74, Mar.-Apr. 2001.
- [25] K. Sourabi, J. Gao, and G. J. Pottie, “Protocols for Self-Organization of a Wireless Sensor Network,” *IEEE Personal Communications*, vol. 7, no. 5, pp. 16–27, Oct. 2000.
- [26] B. Tavli and W. B. Heinzelman, “TRACE: Time Reservation Using Adaptive Control for Energy Efficiency,” *IEEE J. Selected Areas in Communications*, vol. 21, no. 10, pp. 1506–1515, Dec. 2003.
- [27] A. Woo and D. E. Culler, “A Transmission Control Scheme for Media Access in Sensor Networks,” *Proc. MOBICOM 2001*, pp. 221–235, July 2001.
- [28] F. Ye, G. Zhong, S. Lu, and L. Zhang, “PEAS: A Robust Energy Conserving Protocol for Long-Lived Sensor Networks,” *Proc. ICNP 2002*, pp. 200–201, Nov. 2002.
- [29] W. Ye, J. Heidemann, and D. Estrin, “An Energy-Efficient MAC Protocol for Wireless Sensor Networks,” *Proc. INFOCOM 2002*, pp. 1567–1576, June 2002.
- [30] L. Yuan and G. Qu, “Design Space Exploration for Energy-Efficient Secure Sensor Network,” *Proc. ASAP 2002*, pp. 88–97, July 2002.
- [31] J. Zhu and S. Papavassiliou, “On the Energy-Efficient Organization and the Lifetime of Multi-Hop Sensor Networks,” *IEEE Communications Letter*, vol. 7, no. 11, pp. 537–539, Nov. 2003.