A CROSS-LAYER APPROACH IN SUPPORT OF REAL-TIME DATA OVER WIRELESS SENSOR NETWORKS

Albeiro Cortés Cabezas
Department of Electronic Engineering, Surcolombiana University, Grupo de Tratamiento de Señales y Telecomunicaciones, Pastrana Av. Neiva Colombia
E-Mail: albecor@usco.edu.co

ABSTRACT

It is widely known that energy efficiency is very important in wireless sensor networks (WSN). As a result, a common protocol design guideline has been to trade off some performance metrics such as throughput and delay for energy, which also goes well in line with many WSN applications. However, there are other applications with real-time constraints, such as those involved in surveillance or control loops, for which WSN still need to be energy efficient but also need to provide better performance, particularly latency and jitter. This article presents LEMR, a cross-layer design-based communication stack that not only preserves the energy efficiency of current alternatives but also coordinates the transfer of packets from source to destination in such a way that latency and jitter are improved considerably. Coordination is based on hop-distance to the sink, a new MAC layer, and physical layer information. Our approach adopts the Clear Channel Assessment (CCA) and Low Power Listening (LPL) strategies along with channel polling, which is a proven energy-saving method involving the physical and MAC layers. Simulation experiments demonstrate the superiority of LERM in terms of latency and jitter and energy when compared with well-known protocols, such as TMAC, S-MAC and SCP-MAC

Keywords: wireless, sensor, network, MAC, routing, energy, latency.

1. INTRODUCTION

Rapid advances in micro-electro-mechanical systems (MEMS) and low power digital electronics have made possible the development of wireless sensor networks (WSN). A WSN typically comprises of small wireless devices deployed over the physical environment, which cooperate in sensing, processing and communication tasks to provide data about the variables of interest for monitoring and decision making purposes. There are many WSN monitoring applications developed so far in the areas of security, environmental applications, transportation, home automation, civil engineering, etc.

It is well known that wireless sensor devices are very limited in terms of communication and processing capabilities, and storage and energy resources, and that energy is of prime importance. As a result, most algorithms and protocol designs have been made energy efficient. Further, given the characteristics of typical monitoring applications for WSN, a common design practice has been to save energy at the expense of more relaxed QoS performance guarantees, such as low cannel utilization, and longer delays and jitter. For example an approach often used for saving energy has been to design MAC protocols that turn the radios off as much as possible. While this practice saves precious energy, it does it at the expense of worse latency and throughput. Therefore, most of these energy efficient protocols limit the use of WSN to those applications where a prompt response, a delay bounded message delivery, is not important. While this may be a good practice for many applications, there are some applications, such as surveillance and real-time control systems, for which WSN not only need to continue to be energy efficient but also provide better performance.

In this paper we propose LERM (Latency, Energy, MAC and Routing) to fill this gap. LERM is a cross-layer design protocol for wireless sensor networks involving the physical, data link, and network layers of the communication protocol stack. LERM uses physical layer information to improve the reliability in packet transmissions. It includes a new medium access control layer for low energy consumption, and a new node coordination function for low latency and jitter packet forwarding. Through extensive simulation experiments we show the superiority of the LEMR protocol in terms of energy consumption, latency and jitter compared with S-MAC, TMAC and SCPMAC, three well-known energy-efficient MAC layer protocols.

The rest of the paper is organized as follows. Section 2 lists the most important causes of energy wastage in WSN. Section 3 includes a brief related work on cross-layer design approaches and medium access control protocols. Section 4 describes the LEMR protocol. Section 5 presents a performance analysis. Section 6 presents the results of the performance evaluation. Finally, Section 7 concludes the paper.

2. ENERGY AND PERFORMANCE-RELATED ISSUES IN WSN

In this section, a list of the most important issues related to energy consumption, latency and jitter in WSN are included, mostly from the MAC layer point of view. These issues, which should guide the design of algorithms and protocols for WSN according to the application needs, are the following:

- **Collisions:** Collisions should be avoided because the extra energy wasted in frame retransmissions. Recall that communication is the most energy spending function in WSN. Collisions need to be avoided because they also affect the average latency and jitter.

- **Overhead:** Control messages and long headers in frames need to be avoided as much as possible, as they imply extra expensive communication costs.
- **Overhearing:** Overhearing is the energy consumed by the nodes by being constantly listening and decoding frames that are not meant for them. This is a consequence of using a shared media in which nodes do not know a priori whether the transmissions are for them or not.
- **Idle listening:** Idle listening refers to the energy expended by the nodes by having their circuits on and ready to receive while there is no activity in the network. This is particularly important in WSN, as nodes tend to use the channel sporadically. Strategies to turn nodes on and off are very important in WSN. Turning nodes on and off has also latency and jitter implications.
- **Complexity:** Complexity refers to the energy expended as a result of having to run computationally expensive algorithms and protocols. One of the most important design goals in WSN is therefore simplicity.
- **Interference Range (I_R):** a transmitter node can interfere with other nodes farther away from its transmission range, reducing the channel utilization and increasing the latency as well.
- **Unfair Medium Access:** Unfair medium access is a common issue in CSMA-based MAC protocols. Unfair medium access may lead to channel monopolization requiring additional storage space in other nodes for queueing packets, which may lead to additional latency and jitter.
- **Single Global Schedule (SGS):** Some MAC protocols establish and maintain schedules to let each node know when to listen for possible transmission and when to sleep. Since nodes may be listening to more than one schedule, authors in [2] claim for *Single Global Schedule* to reduce energy consumption.
- **False Wake Ups:** False wake ups are a common drawback of MAC protocols using *Channel Polling*. It occur when one node polling the channel identifies signals from out of transmission range nodes and wake up thinking that there might be a transmission directed to them.

3. RELATED WORK

In the last several years, many MAC and routing protocols for WSN have been developed, either separately or using a cross-layer design approach. A common design guideline in most of these works has been the focus on energy savings even at the expense of other not so important variables, such as latency and jitter.

For example, the work presented in [3] considers the physical, MAC, and routing layer in the design but it restricts the link schedules to the class of interference-free time division multiple access (TDMA) schedules. We are rather interested in contention-based link schedules, which are more challenging. The approaches proposed in [4] and [5] have some similarities with LEMR. However, the first one is designed for multicast communication and defines time zones, while LEMR is intended for unicast and broadcast communication. The second protocol uses a random grouping method for the assignment of the group number of each node, while LEMR uses hops distance from the sink.

Many MAC layer protocols have been designed with the goal of saving energy solving the known idle listening and overhearing problems and reducing the overhead. For example, MAC protocols like SMAC [6], TMAC [7] and SCP-MAC [8], divide the time in two periods, one for listening, in which the radio is turned on for communication among the nodes, and one for sleeping, during which the radio is turned off to save energy. In protocols using this approach, the duration of the duty cycle is an important issue since it trades off energy savings for latency. Several strategies have been utilized here. For example, S-MAC sets a fixed duty cycle that wastes energy during low traffic periods. To overcome some of the issues in fixed duty cycles, the TMAC protocol [9] was proposed. In TMAC when one node detects channel activity during its listening period, it remains sensing the channel for a TA time waiting for possible communication with another neighbor. The result is that TMAC can handle bursty traffic better than S-MAC without further energy consumption penalties.

SCP-MAC [8] uses the low power listening (LPL) technique [10] to mitigate the idle listening problem and save energy. This strategy consists of transmitting a preamble in front of each packet to notify the need of a packet transmission. A receiver wakes up periodically every TP seconds and checks for activity in the channel. If the channel is found idle, the receiver goes back to sleep. If a preamble is detected, the receiver stays on and continues to listen until the packet is received. This approach has also been used in the BMAC [11] and X-MAC [12] protocols. While in S-MAC and TMAC the duty cycle is typically 10%, in SCP-MAC it can be reduced to 0.3%, which reduces the energy consumption compared with SMAC and TMAC, but with similar latency performance.

One important aspect of all these protocols is the synchronization strategy. In S-MAC and TMAC, the protocols periodically broadcast SYNC packets in order to achieve node synchronization. With this strategy each node follows a schedule to share packets with its neighbors. Nevertheless, if several neighbors follow deferent schedules, the nodes will wake up more than one time during the frame time, increasing the energy consumption. SCP-MAC uses the same synchronization method; however, the authors propose some strategies to avoid some scheduling problems, like the global schedule algorithm (GSA) presented and described in [2]. Nonetheless, the use of a single global schedule can be the cause of several problems such as jamming in the network, mainly during the busy periods, collisions, interference range problems, unfair medium access, and false wake ups, which increase the latency, as mentioned before.

Another problem with most of these MAC protocols is that they suffer from the data forwarding or cycle interruption problem in which packets must wait for one or more cycles before being forwarded to the next node because nodes out of the transmission range of sender and receiver are unaware of the transmission process. The increase in latency produced by the cycle interruption problem can be alleviated by the adaptive listening mechanism proposed in [6]; however, although it reduces the average latency by half, it also increases the energy consumption. The DMAC protocol described in [13] was also designed to alleviate the cycle interruption problem. However, this technique is only suitable for unidirectional communication flows to the sink, the synchronization network structure is not clear, there is not a coordination strategy, and suffers the idle listening problem.

The cross-layer design protocol, LERM, introduced in this paper makes use of several of these strategies to provide low energy consumption and low latency and jitter. LEMR, as explained in the next section, can be used in unicast and broadcast communications and utilizes channel polling instead of contention windows to send packets, which is more energy efficient. Besides, different to other approaches, the coordination strategy of LEMR considers the interference range, unfair medium access, and avoids false wake ups.

4. THE LEMR PROTOCOL

The LERM protocol consists of a cross-layer design that involves physical layer information, a contention-based medium access control mechanism, and a network layer forwarding mechanism working together to achieve high transmission reliability and low energy consumption, latency and jitter. LEMR adopts the Clear Channel Assessment (CCA) and Low Power Listening (LPL) strategies along with a channel polling energy-saving strategy [8], which is a very energy-efficient method that involves the physical and MAC layers. With channel polling each node wakes up briefly to check for channel activity without actually receiving data. If there is network activity during the channel polling, nodes remain listening waiting for a data packet. After a data packet is received the node goes to sleep again. Using channel polling a reduced duty cycle, as low as 1% or less, can be used, reducing the idle listening problem considerably and providing excellent energy savings. Figure-1 shows a detailed block diagram of this approach.

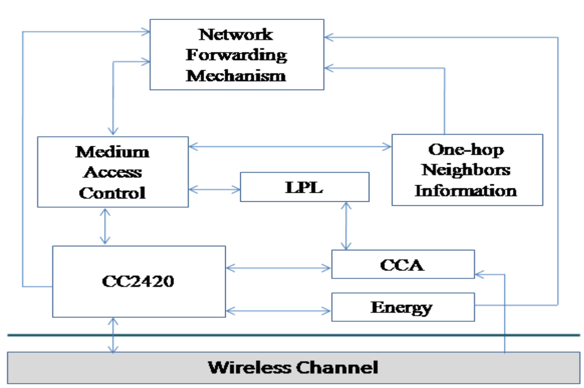


Figure-1. Cross-layer design.

In addition, LEMR reduces the average packet latency and jitter through node coordination, which expedites the forwarding of packets toward the sink. The approach used by LEMR to improve on these performance metrics consists of shifting the channel polling interval according to the distance of the node to the sink in number of hops. In this manner, intermediate nodes do not have to wait for a complete channel polling cycle to forward the packets toward the sink. In order to implement this feature, each LEMR node has a table that contains information about its one-hop neighbors and their hop-distance to the sink. This information is used by the transmitter to know when the receiver will wake up.

Medium access control

The MAC layer of LEMR is very simple. When a node has data to transmit, it randomly chooses a slot within a contention window *CW* that is localized before the channel polling of the possible receiver. The node senses the channel from the start of the contention window up to the chosen slot. If the channel is taken by another node during the random wait period, the node will go to sleep and will try to transmit the data in the next period. If the channel is idle after that number of slots, the node transmits a preamble. The duration of the preamble is such that it contains the rest of the contention window plus the channel polling time. When the transmission of the preamble finishes, the packet transmission begins. If we add the transmission, propagation and processing delays, the preamble will arrive to the following node during the channel polling time, which guarantees that the node will receive the data packet. The transmission of the preamble also overcomes problems related to small drifts in clocks.

In a dense wireless sensor network, each node has an average number *N* of neighbors to share the channel with. As *N* increases, the collision probability also increases and therefore the probability to access the channel decreases. In order to improve the channel access probability and reduce collisions, LEMR's coordination strategy classifies the neighbor nodes into N_0, N_1 and N_{-1} categories, according to whether the neighbor node has equal hop distance or local transmission, it is one hop closer or up transmission, or one hop farther to the sink or down transmission, respectively, as seen in Figure-2. LEMR then only allows channel contention among neighbors with equal hop distance (N_0), eliminating those nodes that mainly interfere with the channel. In this manner, the probability to win the channel by any node is higher with the LEMR protocol than with the TMAC, SMAC or SCP-MAC protocols, and the collision probability is lower.

 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

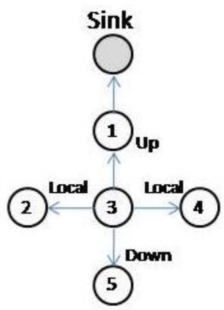


Figure-2. Three directions for data transmission.

Node coordination

The node coordination function is the most important feature of the LEMR protocol, as it is responsible for its latency, jitter and throughput performance and interference mitigation. Node coordination is based on hop-distance to the sink node. To minimize packet latency toward the sink, collisions, unfairness and false wake ups, those nodes n hops away from the sink periodically poll the channel Δ seconds after the channel polling is performed by those nodes (*n + 1*) hops from the sink. Figure-3 shows the channel polling synchronization in a linearly arranged network with four nodes, where the sink is node A, the source is node D, and nodes B and C are intermediate nodes. In the figure, t_0 is a reference time, and Δ is the time interval that separates the channel polling between adjacent nodes.

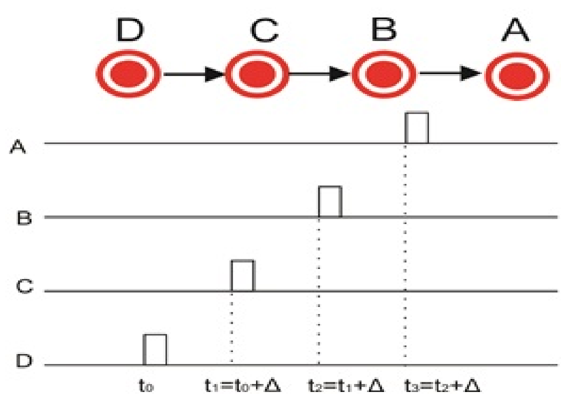


Figure-3. Coordination for a linear network with four nodes.

In this scheme, the value of Δ is very important to achieve the desired coordination. To calculate Δ, we take into account the total transmission time (T_{tx}), the time necessary before the packet transmission starts (T_S), which includes the contention window time, the channel polling time, plus a guard time, the packet processing time (T_{proc}), and the propagation delay (T_{prop}), which is almost negligible given the small area of coverage of wireless sensor nodes. To reduce the overhearing and the hidden terminal problems [14], our implementation uses the well-known RTS/CTS/DATA/ACK sequence. T_{tx} includes the packet transmission time (T_{pkt}), the time necessary to transmit the longest packet that the physical layer supports (128 bytes for the CC2420 transceiver [15]), the control packet transmission time, and the time to switch the transceiver from transmission to reception and vice versa, *Tx/Rx*. Thus,

$$T_{tx} = T_{pkt} + T_{RTS} + T_{CTS} + T_{ACK} + 3T_{Rx/Tx} \quad (1)$$

Where T_{RTS}, T_{CTS}, and T_{ACK} are the RTS, CTS and ACK packet transmission times, respectively. To calculate T_S, we consider the contention window time (T_{CW}), the channel polling time (T_{CP}), and the guard time (T_{guard}), so

$$T_S = T_{CW} + T_{CP} + T_{guard} \quad (2)$$

Note that the preamble transmission time, T_P, should be,

$$T_{CP} < T_P \leq T_{CW} + T_{CP} \quad (3)$$

Then, Δ must be at less equal to:

$$\Delta = T_{tx} + T_s + 3T_{proc} + 3T_{prop} \quad (4)$$

Figure-4 shows the transmission process that corresponds to the linear network shown in Figure-3. From the figure, it can be seen how intermediate nodes B and C can begin forwarding the packets right after they are received, thanks to the polling channel coordination.

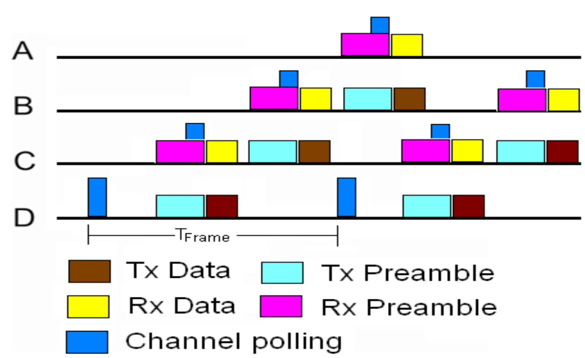


Figure-4. Transmission process from node D to node A.

Synchronization

LEMR uses distributed and localized synchronization. This is, the synchronization process starts from the sink but each node holds local information only. The synchronization starts when the sink node broadcasts a SYNC packet that contains its address, the remaining time to the next channel polling, and its remaining energy. Before the transmission of the SYNC packet, one special preamble packet called ALERT is sent. This preamble has a duration of at least the frame time (T_{Frame}) (one cycle) in order to be listened by all its neighbors. When a node within its range receives the SYNC packet, it updates its hop distance to the sink, and saves the power level at which the SYNC packet was received and the remaining

ARPN Journal of Engineering and Applied Sciences

energy of the sender. At that time, using the ALERT preamble, the node broadcasts a new SYNC packet, with its own address, the address of the sink, the hop distance to the sink, its own remaining time to the next channel polling, and its remaining energy. The node that receives this new SYNC packet assumes that it is one hop farther from the sink. In this manner, the hop distance information to the sink is propagated through the network. After this process converges, it is repeated as needed to capture any topological network changes.

SYNC packets also have a sequence number, chosen by the sink to identify each specific synchronization process. When a node receives a SYNC packet, it compares its sequence number with the one saved, and updates its table (hop distance, power received, and remaining energy) only if it is a new synchronization process. The hop distance value is updated only if the new metric value is smaller than the one saved. Otherwise, the SYNC packet is ignored.

Routing

In the design of LEMR, cross layer information is used to perform the routing function. Once the synchronization process finishes, each node holds in a table its own hop distance to the sink, the hop distance to the sink of all its one hop neighbors, the power level at which the signals were received at the physical layer and the remaining energy of each neighbor. When one node wishes to send packets to the sink, it chooses a node one-hop closer to the sink within its range like the next hop. Since a node could have several one hop neighbors to choose from, an important decision is which one to select. In order to improve the reliability of the protocol, LEMR chooses the one-hop node closer that maximize the following cost function for the link from sender to receiver.

$$C_{ij} = c_1 E_j + c_2 RSSI_{ij} \tag{5}$$

Where E_j is the remaining energy of the receiver, and $RSSI_{ij}$ is the measured power level during the SYNC process. So, LEMR chooses the path with the best signal to noise ratio and remaining energy, which is less error prone. The constants c_1 and c_2 are the weights given to each variable according to the application requirements.

5. PERFORMANCE ANALYSIS

Throughput analysis

The throughput of the flows and the utilization of the channel are affected by the type of traffic and the interferences caused by neighbor nodes several hops away. In order to handle bursty traffic, SCPMAC proposes the adaptive channel polling strategy, which consists of dynamically adding additional, high-frequency polling slots to nodes in the path when the network detects bursty traffic. This approach could be used in LEMR also, but it generates excessive collisions and wastes extra energy.

Since, the LEMR channel access is strongly coordinated and collisions within the transmission range of the nodes are minimized, an interesting question is: What is the polling inter arrival time T_{Frame} that maximizes throughput? In order to maximize the throughput, the interference effects have to be avoided. As such, when a transmitter node injects a packet into the network, it should wait until the farthest node within the interference range of the receiver retransmits it, to inject the following packet. A quick review of Figure-4, assuming equal transmission (T_R) and interference (I_R) range, shows that in order avoid interference effects, node D can inject the second packet after node B retransmits the first one, this is 3Δ after the first packet was injected. However, part of the transmission from node B to node A can coincide with the contention window in node C. Therefore, T_{Frame} has to be set as follows:

$$T_{frame} \geq 3\Delta - T_{CW} \tag{6}$$

We can generalize these results to any interference and transmission range as shown in Table-1, so LEMR allows the maximal channel utilization and throughput with the minimum energy consumption.

Table-1. I_R and T_R vs. T_{Frame}.

Interference range	T_{frame}
T_R	$\geq 3\Delta - T_{CW}$
$2T_R$	$\geq 4\Delta - T_{CW}$
$3T_R$	$\geq 5\Delta - T_{CW}$
$n * T_R$	$\geq (n+2)\Delta - T_{CW}$

Notice that there is an inferior limit to mitigate collisions and interference problems. There is a trade-off here. The bigger the value of T_{Frame}, the longer the queuing delay could be, particularly for bursty traffic. Also, the smaller T_{Frame}, the bigger throughput. However, at the same time, a bigger value of T_{Frame} also means a lower duty cycle and; therefore, lower energy consumption. In the case of periodic traffic, the frame time should be equal or less than the packet interarrival time because this improves energy consumption and has not effects in queuing delay.

Latency analysis

This subsection analyzes the multihop latency of LEMR. A simple linear network of N hops has been chosen as the scenario, where node 1 and node N are selected like source and sink respectively. In addition, we assume that the rate of packets generated at the source node is at most $1/T_{Frame}$ packets per second, which is a typical condition in wireless sensor networks. As such, the packets queuing time will be equal to zero. In addition, it can be noticed that, contrary to S-MAC, TMAC and SCP-MAC, in LEMR, under this scenario, there is neither contention nor interference problems, therefore, the probability of winning the channel is 1. Hence, a packet

VOL. 11, NO. 24, DECEMBER 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

through an N hop network will experience an average delay of Δ seconds per hop. Under this scenario the total average latency of LEMR (*E[D(N)]*)is given by:

$$E[D(N)] = N\Delta + \frac{T_{frame}}{2} \tag{7}$$

Where, $T_{Frame}/2$ is because packets can be generated at any time during the frame time. Recall that the average latency in the case of the S-MAC protocol is proportional to T_{Frame}, as it is shown in the Equation 8 [6], where *Tcs* is the carrier sense time at each hop. Note that S-MAC trades off energy for latency. The last is a typically feature of single global schedule MAC protocols. TMAC and SCPMAC protocols present a similar behavior, since they are derived from S-MAC. On the other hand, this trade off does not exist in LEMR since latency is proportional to Δ only. Since T_{Frame} is bigger than Δ, LEMR provides better latency than the other protocols.

$$E[D(N)]_{SMAC} = \frac{N * T_{frame}}{2} + 2T_{cs} + 2T_{pkt} - \frac{T_{frame}}{2} \tag{8}$$

6. PERFORMANCE EVALUATION

Simulation setup

In order to evaluate the performance of the LERM protocol under different scenarios and networks conditions, detailed simulation models where developed in the QualnetR simulator [16]. The simulation models for the S-MAC, TMAC and SCP-MAC protocols were also developed in Qualnet® for comparison purposes. In order to simulate the physical layer, the RF CC2420 [15] transceiver was modeled according to the characteristics and parameters shown in Table-2. In this model, the time each node spends being idle, polling, sensing, transmitting, and receiving are used to calculate the total power consumed.

The main parameters of the protocols under evaluation were set as follows. Our implementation of the S-MAC protocol uses the adaptive listening mechanism included in [6] with a 10% duty cycle.

The contention window for data transmission is 63ms, the contention window for the SYNC packet transmission is 32ms, and the frame time (T_{Frame}) is set to 1:1445sec. Our implementation of the TMAC protocol uses a *TA* value of 15ms, a contention window of 10ms, and a 10% duty cycle. The implementation of SCP-MAC uses adaptive channel polling. The channel polling is set to 3ms and the frame time is 300ms, therefore, the duty cycle is 1%. Besides, our SCP-MAC model uses a common single global schedule. In the implementation of LEMR, Δ is set to 26:4ms, the frame time is 264ms, and the channel polling is set to 3ms, therefore, the duty cycle is 1.14%. We used a maximum packet size of 128 bytes at the physical layer, which includes all the headers from upper layer protocols. Static routing is used in S-MAC, TMAC and SCP-MAC simulations while routing in LEMR is performed as explained in Section 4.

Table-2. Simulation parameters.

Parameter	Value
Transmission power	10dBm
Sensitivity	-95dBm
Receiver threshold	-77dBm
Bit transmission rate	250kbps
Tx Power consumption	91.4mW
Rx Power consumption	59.1mW
Sleep Power consumption	15μW
Maximum packet size	128 bytes
c_1 y c_2	1

Transmission v. interference range

The interference and transmission ranges are two important parameters in LEMR design. If there is no interference from other nodes, the expected power $P_r(d)$ at certain distance can be calculated by:

$$\frac{P_r(d_0)}{P_r(d)} = \left(\frac{d}{d_0}\right)^{\beta} \tag{9}$$

Where $P_r(d_0)$ is the reference power received at the reference distance d_0, and $\beta \geq 2$ is the path loss exponent, which is usually empirically determined by field measurements. As such, the transmission and interference ranges are related as follows:

$$P_r(T_R) * (T_R)^{\beta} = P_r(d_0) * (d_0)^{\beta} = P_r(I_R) * (I_R)^{\beta} \tag{10}$$

$$\left(\frac{I_R}{T_R}\right)^{\beta} = \frac{P_R(T_R)}{P_R(I_R)} \tag{11}$$

In our scenarios, $\beta = 4$, the transmission range is set to 100.181 meters and the sensitivity of the receiver is as in the Table-2, therefore the relation $\frac{I_R}{T_R} \approx 2.82$, which is very common in performance evaluations found in the literature.

Simulation

In order to study the performance of the four protocols under contention conditions, the linear network with three sources and one sink shown in Figure-5 was utilized. In the simulations, each source node sends 100 packets of 128 bytes each toward the sink, with a fixed data generation interval between 1 and 10 secs. The nodes are physically separated by a distance of 80m, therefore each node can only hear its side neighbors, or up or down neighbors. Each experiment is repeated 30 times to collect enough data for statistical confidence. The simulations are run for a total of 3000 sec. We assume that the nodes have enough buffers to avoid packet loss during congestion periods.

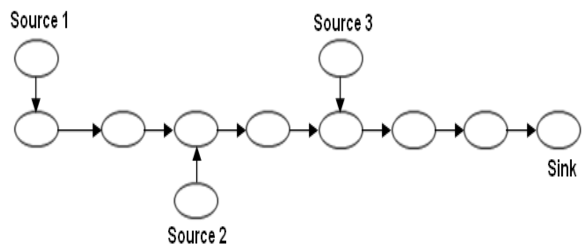


Figure-5. Linear network with three sources and one sink.

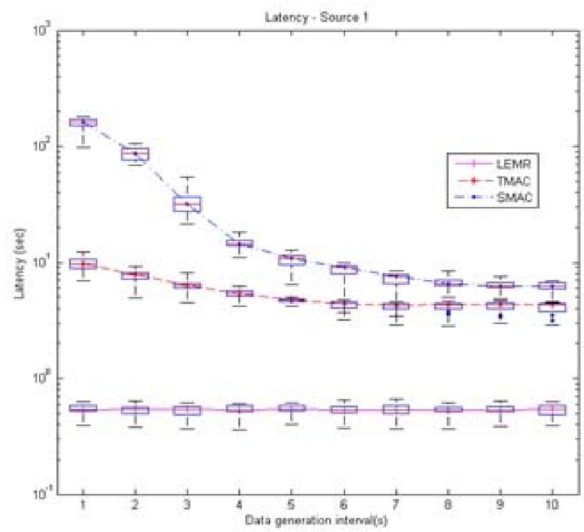
Figures-6(a), 6(b) and 6(c) show the average latency of each source for the four protocols under consideration as the data transmission interval (transmission rate) is increased. From the figures, it can be observed that the latency provided by the coordination function of the LEMR protocol is not only the best but also presents the smallest variability. LEMR outperforms S-MAC more than 100 times, and TMAC and SCP-MAC 10 times at high transmission rates, respectively, and 10 times in lower transmission rates. Queuing results, (doesn't shown here), show the duty cycle as the cause of the high latency particularly in S-MAC when the frame length is longer than data generation interval.

Although with the S-MAC protocol nodes can transmit up to two packets per frame time, nodes with same schedules have a higher collision probability that increases the average end-to-end delay. In addition, with $I_r = 2.82 \cdot T_r$ it can be observed that an intermediate node causes interference with a diameter of 6 hops, which means that the network presents a bottleneck at the intermediate nodes with a maximum average throughput of 2/7 packets per frame. Despite that TMAC and SCP-MAC are derived from S-MAC, they present less latency, because they are designed to relay more packets per frame. In contrast, because of the coordination strategy of LEMR, collisions and interference are mitigated and packets are transmitted continuously with very low latency and variability.

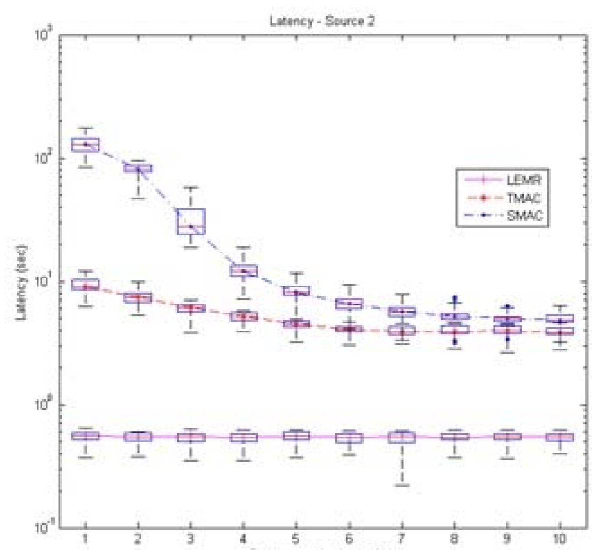
Figure-6(d) shows the average energy consumption in the network. One of the principal limitations of the S-MAC and TMAC protocols is the amount of energy wasted due to idle listening. On the other hand, the short preamble and the channel polling strategy of LEMR and SCP-MAC alleviate this problem, consuming more than 7 times less energy than SMAC and TMAC. It is worth noticing that further energy savings could have been achieved with LEMR if the duty cycle had been adjusted according to the application packet data rate. For example, in our experiments, we could have set T_{Frame} equal to 1sec (instead of 264ms), which is the smallest packet inter arrival time of the CBR sources, and therefore reduce the duty cycle considerably.

Performance evaluation using a grid scenario

In order to evaluate LEMR under most realistic conditions the 10 x 10 node grid network shown in Figure-7 was considered. In this scenario we investigate the performance of LEMR when several sources transmit at the same time. Several simulations are performed selecting the source nodes at random and taking the average. In this scenario packets of 128 bytes are generated with a uniform interarrival time between 1 and 20 sec during 3000 sec. Columns and rows are 80m apart. S-MAC, TMAC and SCP-MAC use static routing.

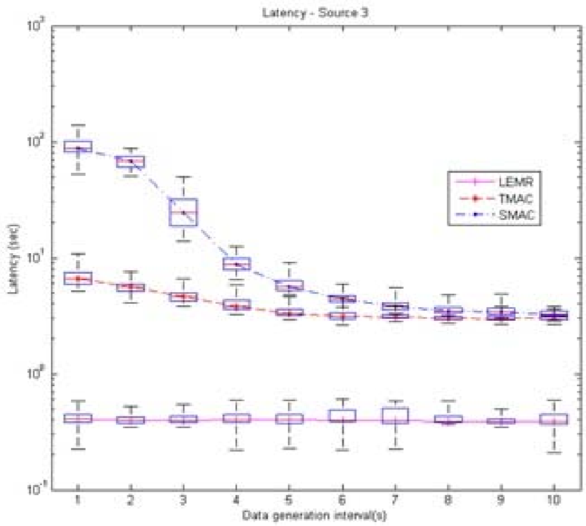


(a)Average latency seen from source 1.

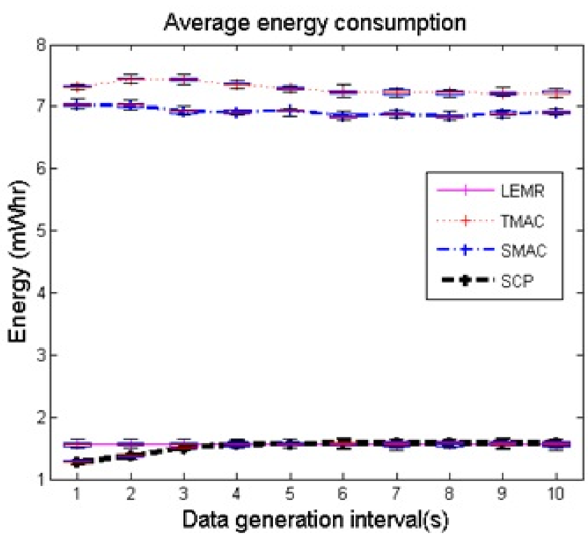


(b)Average latency seen from source 2.

ARPN Journal of Engineering and Applied Sciences



(c)Average latency seen from source 3.



(d)Average energy consumption.

Figure-6. Average latency and energy consumption of the protocols under consideration using the linear network.

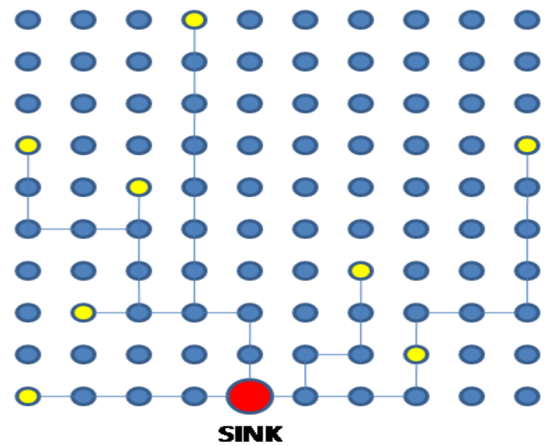


Figure-7. Grid network scenario.

Figure-8(a) shows how the average latency in LEMR remains almost constant and at least 10 and 100 times better than the other protocols as the number of sources is increased from 2 to 10. SCP-MAC presents better performance than TMAC, however when the number of sources is increased TMAC's performance is better. This is because SCP-MAC is more prone to collisions than TMAC in this scenario. Compared with the linear network scenario presented before, in this scenario nodes are subject to higher contention conditions, therefore S-MAC, TMAC and SCP show higher latency, particularly when the number of sources is increased. However, the latency of LEMR remains almost unaffected due to the immunity provided by its coordination strategy.

Figure-8(b) compares the throughput achieved by each protocol measured at the sink. As it can be seen, the LEMR, TMAC and SCP-MAC protocols have comparable performance. On the other hand, S-MAC once again, presents the worst performance. In this scenario the SMAC protocol achieves a throughput lower than 2/7 packets per frame, because there are more nodes involved than in the ideal linear scenario. In addition, as the number of sources is increased, contention and interference increase as well, decreasing the throughput.

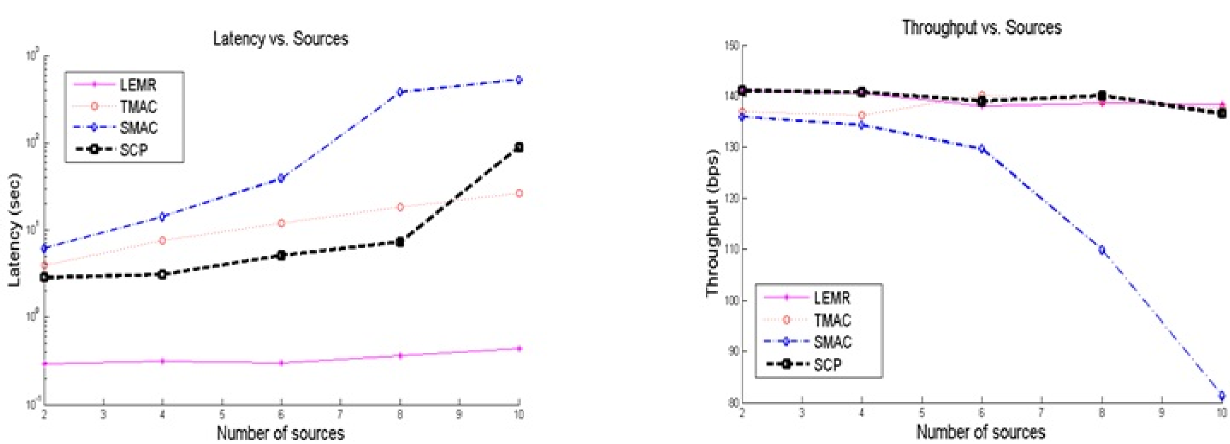


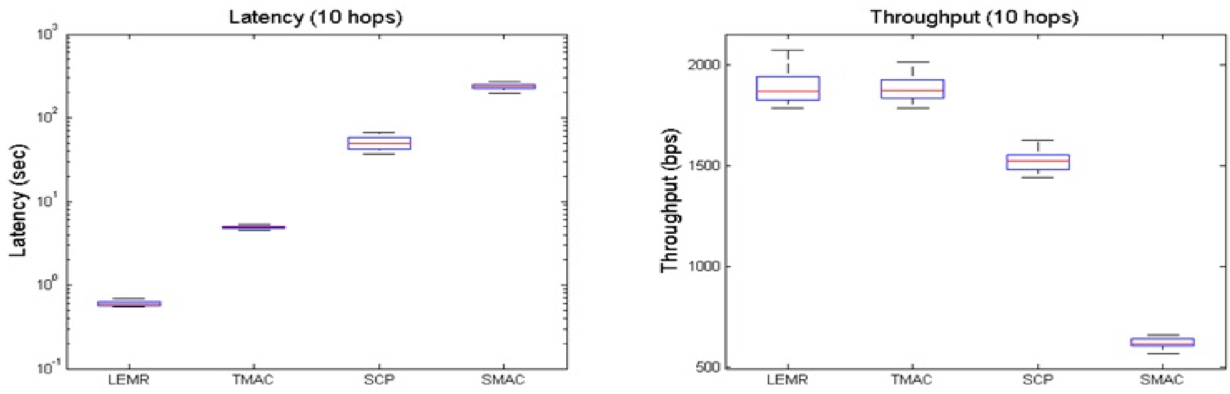
Figure-8. Average latency and throughput of the four protocols under consideration using the grid network.

VOL. 11, NO. 24, DECEMBER 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Performance evaluation using poisson traffic

LEMR has been designed to support applications that need low latency and low variability. Most of the time, these are real-time applications sending data at constant bit rates. So, a valid question is: how does LEMR perform when applications send data in a nonconstant manner? In order to evaluate this scenario, a linear network topology with 10 hops was selected in which the first node is the source and the eleventh node is the sink. We include enough buffer space at the intermediate node to avoid packet losses. Traffic is generated according to a Poisson process sending 128-byte packets at an average rate of 2.5 packets per second (λ=2.5*pps*). The simulation time was set to 1000 seconds and each experiment was repeated 30 times to collect enough of statistics.



(a) Average latency (b) Average throughput

Figure-9. Average latency and throughput in a 10-hop linear network using poisson traffic.

Figure-9(a) shows the average latency of each protocol. LEMR outperforms the TMAC protocol over 10 times, the SCP-MAC protocol over 100 times, and the S-MAC protocol over 250 times. Similarly, Figure 9(b) shows the average throughput achieved by each protocol as measured by the sink. The S-MAC protocol presents the lowest performance, followed by SCP-MAC. LEMR and TMAC are the best presenting similar performance.

The throughput results of S-MAC and SCP-MAC are similar to the ones observed in earlier sections. However, TMAC presents a higher average latency than LEMR, although both have similar average throughput. The TMAC protocol allows nodes to get the channel during short time intervals and transmit several packets during that interval. As a result, intermediate nodes may receive burst of packets that need to be stored in the queue, producing the latency plus the requirement for higher temporal storage space at each node. In contrast, the channel access coordination of LEMR allows nodes to relay one packet per frame time only and relay them immediately.

Finally, Figure-10 shows the average energy consumption for each protocol. As in past results, LEMR exhibits the best performance followed by SCP-MAC. Although LEMR and SCP-MAC use similar energy-saving strategies, the synchronization strategy used by SCP-MAC presents excessive collisions. SMAC and TMAC have higher energy consumption due to the idle listening problem.

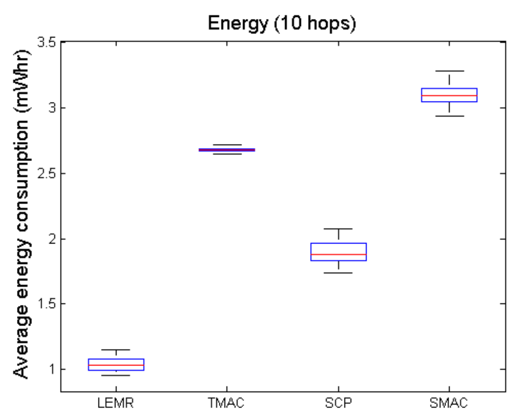


Figure-10. Average energy consumption.

7. CONCLUSIONS

This paper presents LEMR, a protocol designed to support wireless sensor network applications where latency and energy are important performance metrics. LEMR includes in its design physical, MAC, and network layer information to provide efficient MAC access and packet forwarding with low energy and latency and high reliability. This is accomplished using channel polling and an efficient node coordination strategy started from the sink. LEMR is compared with S-MAC, TMAC and SCPMAC using simulations. The results demonstrate that LEMR outperforms S-MAC, TMAC and SCP-MAC in both energy and latency metrics. LEMR consumes far less energy than S-MAC and TMAC in all the scenarios considered while SCP-MAC shows higher energy consumption in most cases. Regarding latency, LEMR outperforms S-MAC, TMAC and SCP-MAC over 100 and 10 times for small data generation intervals, respectively, and 10 times when sources transmit packets at slower transmission rates.

The single global schedule of S-MAC, TMAC and SCP-MAC was identified as the cause of more collisions, increasing the latency and energy consumption. Also, the synchronization strategy of LEMR not only mitigates the collisions but also the interference effects, which are not considered in the design of S-MAC, TMAC and SCP-MAC. Finally, another advantage of LEMR is that false wake ups are eliminated, which is an additional factor of the lower performance is observed by the other protocols using channel polling.

REFERENCES

[1] Cortéss A, Gamboa R, Peña N.M, Labrador M. Low Energy and Low Latency in Wireless Sensor Networks. In Proceedings of International Conference on Communication (IEEE ICC'09). 2009.

[2] Li Y, Ye W, and Heidemann J. Energy and Latency in Low Duty Cycle MAC Protocols. In Proceedings of IEEE Wireless Communications and Networking Conference. 2005.

[3] Madan R, Cui S, Lall S, and Goldsmith A. Cross-Layer Design for Lifetime Maximization in Interference-Limited Wireless Sensor Networks. IEEE Transactions on Wireless Communications. Vol. 5 2006.

[4] Ruzzelli A, O'Hare G, and Jurdak R. MERLIN: Cross-layer integration of MAC and routing for low duty-cycle sensor network. Elseiver Ad Hoc Networks. Vol. 6, pp. 1238-1257. 2008.

[5] Chen C.W, Weng C.C. and Ku C.J. Design of a low power and low latency MAC protocol with node grouping and transmission pipelining in wireless sensor network. Elseiver Computer Communications. Vol. 31, pp. 3725-3738. 2008

[6] Ye W, Heidemann J, and Estrin D. Medium Access Control with Coordinated Adaptive Sleeping for Wireless Sensor Networks. IEEE/ACM Transactions on Networking. Vol. 12, pp. 34-39. 2004.

[7] Dam T.V. and Langendoen K. An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks. In Proceedings of the 1st International Conference on Embedded Networked Sensor Systems. 2003.

[8] Ye W, Silva F. and Heidemann J. Ultra-Low Duty Cycle MAC with Scheduled Channel Polling. In Proceedings of the 4th International Conference on Embedded Networked Sensor Systems 2006.

[9] Langendoen K. Medium Access Control in Wireless Sensor Networks. Nova Science Publishers. pg 630. 2008.

[10] El-Hoiydi A. Aloha with Preamble Sampling for Sporadic Traffic in Ad HocWireless Sensor Networks. In Proceedings of IEEE International Conference on Communications. pp. 3418-3423, 2002.

[11] Polastre J, Hill J. and Culler D. Versatile Low Power Media Access for Wireless Sensor Networks. In Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems. pp. 95-107, 2004.

[12] Buettner M, Yee G.V, Anderson E. and Han R. X-MAC: A Short Preamble MAC Protocol for Duty-Cycled Wireless Sensor Networks. In Proceedings of the 4th International Conference on Embedded Networked Sensor Systems. pp. 307- 320, 2006.

[13] Lu G, Krishnamachari B. and Raghavendra C. S. An Adaptive Energy-Efficient and Low-Latency MAC for Data Gathering in Sensor Networks. In Proceedings of the 18th International Parallel and Distributed Processing Symposium (IPDPS04). 2004.

[14] Bharghavan V, Demers A, Shenker S. and Zhang L. MACAW: A media access protocol for wireless lans. In Proceedings of ACM Sigcomm. pp. 212-225, 1994.

[15] CC2420 DataSheet. Available in http://www.chipcon.com. Chipcon, 2005.

[16] The Qualnet simulator. Version 4.0. Available from http://www.scalable-networks.com/. Qualnet, 2008