

Energy-Aware TDMA-Based MAC for Sensor Networks

Khaled A. Arisha
Honeywell International Inc.
Advanced Systems Tech. Group
7000 Columbia Gateway Drive
Columbia, MD 21046

Moustafa A. Youssef*
Department of Computer Science
Univ. of Maryland at College Park
College Park, MD 20742

Mohamed F. Younis*
Dept. of Comp. Sc. and Elect. Eng.
Univ. of Maryland Baltimore County
1000 Hilltop Circle
Baltimore, MD 21250

Abstract

Networking unattended sensors is expected to have a significant impact on the efficiency of many military and civil applications. Sensors in such systems are typically disposable and expected to last until their energy drains. Therefore, energy is a very scarce resource for such sensor systems and has to be managed wisely in order to extend the life of the sensors for the duration of a particular mission. In this paper, we present a novel approach for energy-aware management of sensor networks that maximizes the lifetime of the sensors while maintaining desired quality of service attributes related to sensed data delivery. The approach is to dynamically set routes and arbitrate medium access to minimize energy consumption and maximize sensor life. We give a brief overview of the energy-aware routing and describe a Time-Division-Multiple-Access (TDMA) -based Medium Access Control (MAC) protocol. We discuss algorithms for assigning time slots for the communicating sensor nodes. The approach is evaluated through simulation. Simulation results have confirmed the effectiveness of our new approach.

1. Introduction

Sensor networks have recently attracted significant attention for many military and civil applications, such as target tracking, surveillance and security management. Sensors monitor events in a surveillance area and observed data are collected and analyzed. Examples of sensor deployment organization for a target-tracking oriented mission can be found in [1][4]. Sensors have limited energy resources and their functionality continues until their energy drains. Therefore, energy resources for sensor networks should be managed wisely to extend the lifetime of sensors.

The sensing element of a sensor probes the surrounding environment. After performing signal

processing of the observed data, sensors communicate this data, typically using a radio-based short-haul links, to a command center usually through a relay or a data concentrator called the gateway. Gateways fuse collected data and send it to the command node for further analysis. Sensors cannot be active all the time, since signal processing and communication circuits can consume most of its energy, thus shortening the lifetime of the sensor

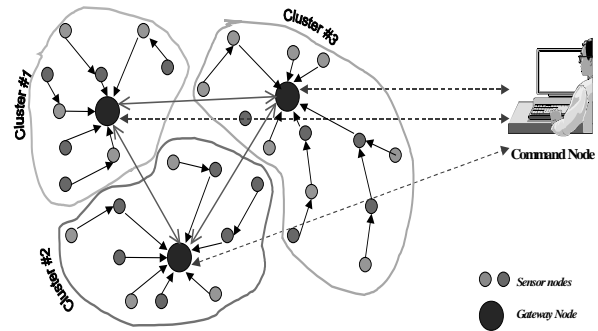


Figure 1: Multi-gateway clustered sensor network

network. Energy-aware network management is highly critical to maintain a longer lifetime while still performing its task within an acceptable level of quality.

Due to scalability requirements and to avoid overloading the gateways, network clustering is recommended through the involvement of multiple gateways, as shown in figure 1. Clusters are formed such that its gateway is located within the communication range of all of its cluster sensors. Gateways use long-haul communication to send reports fused from its cluster sensor data to other gateways, and eventually to the command node. Clustering, Inter-gateway communication, data fusion and task allocation are beyond the scope of this paper.

Sensors are assumed to be capable of operating in an active mode or a low-power standby mode.

* Work has been partially done while the author was at Honeywell International. Inc.

The sensing circuits as well as radio transmitters and receivers can be turned on and off independently. Transmission power can be adjusted based on the required range. Sensors can act as store-and-forward relay nodes. We assume the on-board clocks of both the sensors and gateways are synchronized. Clock synchronization can be achieved via the use of GPS or through the exchange of synchronization messages [3]. This assumption is justified since such capabilities are typically implemented in advanced sensors, e.g. the Acoustic Ballistic Module from SenTech Inc.[2]. Gateways are responsible of the dynamic configuration of the sensor network within its cluster.

1.1. Related Work

In wired and inherently wireless networks, the emphasis has traditionally been on the optimization of throughput and end-to-end delay. Only recently energy efficiency has received attention due to advances in wireless networks. Generally energy efficiency can be achieved at various layers of the communication protocol stack. The bulk of energy-related research has focused on the hardware level, for example [5]. Due to fundamental physical limitations, the focus has recently shifted to architectural and software levels. Given the scope of this paper we will limit the discussion to data link layer and network layer protocols.

Energy-aware routing has started to receive attention in the recent few years, motivated by advances in wireless mobile devices. A comparison between direct routing and minimum energy routing has been conducted in [12]. Other comparison between different energy-aware routing protocols is given by Toh [10]. The trade-off between extending lifetime and fair usage of sensors is analyzed in [14]. A position-based minimum energy network is proposed in [13]. A signaling channel is used in [8] to intelligently turn off nodes that are not active, however nodes use a complex probe mechanism. Store-and-forward schemes of wireless networks, such as IEEE 802.11, have a sleep mode in which nodes are turned off [9][10].

A power-aware Time Division Multiple Access (TDMA) Medium Access Control (MAC) protocol that coordinates the delivery of data to receivers based on the base station control is given in [7]. There are three phases in this TDMA: up-link phase in which nodes transmit data to the base station,

down-link phase in which the base station transmits data to the nodes, and reservation phase in which nodes request new connections. The base station dictates a frame structure within its range. A frame consists of a number of data cells and a traffic control cell. Nodes with scheduled traffic are indicated in a list, which allows nodes without traffic to rapidly reduce power. The traffic control is transmitted by the base station and contains information about the subsequent data-cells, including when the next traffic control cell will be transmitted. Nodes explicitly request transmission from the base station, in a distributed manner, during the reservation phase. In our approach, the gateway performs the slot assignment based on its routing decisions. Our approach, as explained later, has four phases some of them have different functionality than their approach. Their approach requires the three phases to be present in every frame while in our approach the data send phase (up-link phase in their approach) is more frequent than the other phases leading to less control overhead and thus higher bandwidth efficiency. The gateway informs each node of its state so that a node can turn itself off. They did not discuss the effect of transmission errors on collision and network performance.

In this paper, we focus on the network management by the gateway of its cluster sensors, mainly the MAC mechanism. The MAC-layer protocol uses centralized reservation made by the gateway and has a less control overhead. The next section briefly describes our approach for the energy-aware routing and explains the details of the MAC layer protocol. Section 3 discusses the validation environment for our approach and analyzes the results of the simulation experiments. Finally, conclusion and potential extensions are presented.

2. Energy-aware Network Management

The main objective of our approach is to extend the lifetime of the sensors through topology adjustment, energy aware routing and MAC. Messages are routed through multi-hop paths to preserve the sensor transmission energy. Message traffic between sensors is arbitrated in time to avoid collision and to allow turning off the unneeded sensors. Gateway nodes assume responsibility in its cluster for sensor organization and routing/MAC

management. Since the gateway organizes the sensors in the cluster, it can account for energy commitment to data processing, remaining sensor energy, sensor locations and acceptable quality of service such as message latency. It can as well enhance the robustness and effectiveness of the MAC because the decision to turn a node receiver off can be more accurate and deterministic than a decision based on local MAC protocol [8]. Sensors can be in one of four states: sensing, relaying, sensing-relaying and inactive. The gateway uses model-based energy consumption for the data processor, radio transmitter and receiver to track the sensor's energy level. Periodically, the gateway adjusts the energy model by querying the actual levels of the sensor energy.

2.1. Energy-aware Routing

Our routing approach is centralized in the gateway node for each cluster. We set routes for sensor data from the active sensor node to the gateway in order to optimize an objective function. The problem can be viewed as the transpose of a single-source routing algorithm, i.e. single destination routing. The objective function describes a path optimization problem, which is proved to be of polynomial time complexity [15]. Our algorithm is based on the one-to-some shortest path problem, which is performed the best by Dijkstra algorithm [16]. The objective function extends the definition of the cost function of Dijkstra algorithm to consider the nature of the sensor network. The new cost factors are classified as energy, routing, and delay related factors. The energy related factors consider communication cost, remaining sensor energy, energy consumption rate, relaying enabling cost and sensing enabling cost. Routing factors consider connection per relaying sensor. Delay factors are concerned with transmission delay and queuing cost.

The sensor energy model in the gateway is maintained through received packet updates. Each packet received changes the capacity of the nodes along the path from the initiator sensor down to the gateway. The gateway uses its routing table to keep track the nodes along the path. A refresh phase is scheduled periodically to correct deviations in the energy model due to model inaccuracy, packet drop due to communication error, or packet drop due to buffer overflow. During the periodic refresh phase, each node sends a state-refresh packet. Then, during

the routing phase each node turns its receiver on at a pre-specified time to hear the gateway routing decision. In case of lost refresh packets, the node maintains its previous state.

Routing setup can be dynamically adjusted to optimally respond to changes in the sensor organization. Rerouting decision is based on three criteria: sensor reorganization such as an event that requires reselection of active sensors, node battery energy level if it drops to a certain level and energy model adjustment after refresh updates. Details of our routing approach can be found in [21][22].

2.2. MAC Layer Protocol

Although the new routing protocol is independent of the MAC layer protocol, choosing a certain MAC layer protocol may enhance the performance. Recent research results pointed out that the wireless network interface consumes a significant fraction of the total power. Measurements show that on a typical application like web-browser or email, the energy consumed when the interface is on and idle is more than the cost of receiving packets. This is because the interface is generally longer idle than actually receiving packets. Furthermore, switching between states (i.e. off, idle, receiving, transmitting) consumes time and energy [6]. Therefore, in a wireless system the medium access protocols can be adapted and tuned to enhance energy efficiency.

We choose to implement a time division multiple access (TDMA) based MAC layer whose slot assignment is managed by the gateway. The gateway informs each node about slots in which it should listen to other nodes' transmission and about the slots, which the node can use for its own transmission. The advantages of using a TDMA MAC layer are:

- Clock synchronization is built in the TDMA protocol. Recall that we need synchronization for the energy model refresh and sending rerouting decision from the gateway to the nodes.
- Collision among the nodes can be avoided since each node has its own assigned time slots. Problems can occur with the existence of communication errors: a packet containing the slot assignment can be dropped. If a node that does not hear the gateway decision turns itself off, then no collision can occur. However, we

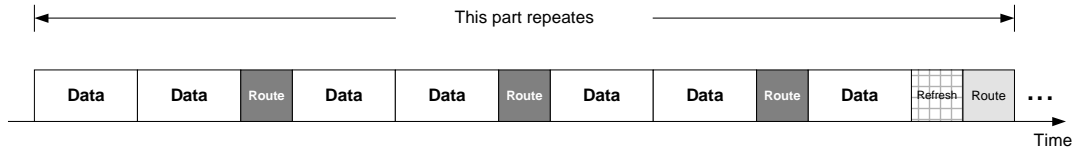


Figure 2: MAC protocol time-based phases

choose to implement the other alternative that a node retains its previous state if it does not receive a routing packet from the gateway in the pre-specified time slot, which leads to potential collisions. However, this collision probability is limited due to the following reasons:

- A node's new state and forwarding table is highly probable to remain the same during consecutive rerouting phases.
- The wrong state of the node will be corrected during the next rerouting cycle, which means that the collision period is limited.
- If the node's *previous* state was inactive, no collision will happen.
- If the node's *new* state is inactive, no packets will be destined to it reducing the collision probability (recall that a node can overhear other nodes' transmissions.)
- If the node receives a packet that is not in its forwarding table, this packet is dropped.
- Collision can only occur if the node happens to use the same time slot for transmission as a neighboring node since during transmission, we use the minimum transmission power required for reaching the destination. The same thing happens during receiving.

In the following subsections, we present the details of the MAC layer protocol.

2.2.1. Protocol Phases and Packet Format

The protocol consists of four main phases: data transfer, refresh, event-triggered rerouting, and refresh-based rerouting phase. In the data transfer phase, sensors send their data in the time slots allocated to them. Relays use their forwarding tables to forward this data to the gateway. Inactive sensor nodes remain off until the time for sending a status update or to receive route broadcast messages.

The refresh phase is designated for updating the sensor model at the gateway. This phase is periodic and occurs after multiple data transfer phases, Thus minimizing the routing overhead compared to the

payload data. Periodic adjustments to sensor status enhance the quality of the routing decisions and correct any inaccuracy in the assumed sensor models. During the refresh phase, each node in the network uses its pre-assigned time slot to inform the gateway of its state (energy level, state, position, etc). Any node that does not send information during this phase is assumed to be nonfunctioning. If the node is still functioning and a communication error caused its packet to be lost, its state may be corrected in the next refresh phase. The slot size in this phase is less than the slot size in the data transfer phase as will be explained later. Figure 2 shows an example of a typical sequence of phases.

As previously discussed in subsection 2.1, rerouting is performed when the sensor energy drops below a certain threshold, after receiving a status update from the sensors and when there is a change in the sensor organization. Since the media access in our approach is time-based, rerouting has to be kept as a synchronous event that can be prescheduled. To accommodate variations in the rate of causes of rerouting, two phases are designated for rerouting and scheduled at different frequencies. The first phase is called event-based rerouting and allows the gateway to react to changes in the sensor organization and to drops in the available energy of one of the relay sensors below a preset acceptance level. The second rerouting phase occurs immediately after the refresh phase terminates. During both phases, the gateway runs the routing algorithm and sends new routes to each node in its pre-assigned slot number and informs each sensor about its new state and slot numbers as shown in table 1. Given that events might happen at any time and should be handled within acceptable latency, the event-based rerouting phase is scheduled more frequently than the refresh-based rerouting. If there has not been any events requiring messages rerouting, the event-triggered rerouting phase becomes an idle time.

Table 1: Description of MAC Protocol Phases

Phase	Initiator	Schedule	Actions
Data send	Active sensors	Assigned time slot	Send/forward data packets
Refresh	All sensors	Pre-assigned time slot	Inform gateway of sensor state
Refresh-based rerouting	Gateway	After refresh phase	Setup routes based on updated model.
Event-triggered rerouting	Gateway	Periodic	Setup routes to handle changes in sensor selection and energy usage.

The lengths of the refresh and reroute phases are fixed since each node in the sensor network is assigned a slot to use in transmission during the refresh phase and to receive in it during the reroute phases. Similarly, the length of the data transfer phase is fixed. Although the number of active nodes changes from a rerouting phase to another, the length of the data transfer phase should be related to the data sending rate and not to the number of active nodes. If the length of the data transfer phase is dependent on the number of active nodes, then a node may consume power while it has nothing to do. It should be noted that during system design the size of the data transfer phase should be determined to accommodate the largest number of sensors that could be active in a cluster. Since the length of all phases is fixed, the period of the refresh and rerouting phases can be agreed upon from the beginning and does not have to be included in the routing packets.

The description for the packets of the corresponding phases is shown in the table 2. In the data packet used in the data transfer phase includes the originating sensor ID so that the gateway can adjust the energy model for the sender and relay sensors. In addition the sensor ID identifies the location and context of the sensed data for application-specific processing. The refresh packet includes the most recent measurement of the available energy. The optional location coordinates can be used to support sensor mobility.

Table 2: Description of various packet types

Source	Target	Type	Contents
Sensor	Gateway	Data	Orig ID, Data
Sensor	Gateway	Refresh	Orig ID, Source battery level, Source Location
Gateway	Inactive Sensor	Rerouting	Dest ID
Gateway	Sensing Sensor	Rerouting	Dest ID, Data send rate, Trans range, Time slots
Gateway	Relaying Sensor	Rerouting	Dest ID, Forward table, Time slots

The content of a routing packet depends on the new state of the recipient sensor node. If the sensor is to be in an *Inactive*, the packet simply includes the ID of the destination node. In case of a node that is set to sense the environment, the packet includes the data sending rate and the time slots during which these data to be sent. In addition, these sensing nodes will be told the transmission range, which the node has to cover. Basically the transmission power should be enough to reach the next relay on the path from this node to the gateway, as specified in the routing algorithm. Relay sensors will receive the forwarding table that identifies where data packet to be forwarded to and what transmission to be covered.

The forwarding table consists of ordered triples of the form: (time slot, data-originating node, transmission range). The “time slot” entry specifies when to turn the receiver on in order to listen for an incoming packet. The “source node” is the sensor node that originated this data packet, and “transmission power” is the transmission power to use to send the data. This transmission power should be enough to reach the next relay on the path from the originating node to the gateway. It should be noted that the intermediate nodes on the data routes are not specified. Thus it is sufficient for the relaying nodes to know only about the data-originating node. The transmission range ensures that the next relay node, which is also told to forward that data packet, can clearly receive the data packet and so on. Such approach significantly simplifies the implementation since the routing table size will be very small to maintain and the changes to the routes will be quicker to communicate among the sensors. Such

simplicity is highly desirable to fit the limited computational resources that sensors would have. We rely on the sensor organization and smart data fusion to tolerate lost data packets by allocating redundant sensors and applying analytical techniques.

2.2.2. Slot Size and Assignment

The slot sizes for the refresh and reroute phases are equal since they cover all sensor nodes in the cluster. Both slots are smaller than the slot for the data transfer phase. This is due to two reasons. First, the routing packet is typically less than the data packet. Second, during the data transfer phase many nodes are off which allows for larger slot sizes. In the other phases, all nodes must be on and communicating with the gateway. To avoid collision while packets are in transit, the slot size in the refresh and reroute phases should be equal to the time required to send a routing packet with maximum length plus the maximum propagation time in the network, as calculated by the gateway. The slot size of the data-transfer phase equals the time required to send a data packet with maximum length plus the maximum propagation time in the network.

Slot assignment is performed by the gateway and communicated with the nodes during the rerouting phases. Different algorithms can be used for slot assignment. We assign each node a number of slots for transmission based on its current load. This leads us to two approaches for handling the TDMA-based MAC slot assignment problem, namely breadth and depth techniques. In the breadth slot assignment technique we follow a breadth-first-search (BFS), commonly used for graph parsing, to assign time slot numbers starting from the outmost active sensors. These outermost sensors are all sensing enabled since they are the source nodes of our data, and thus the initiator nodes in the routes towards the gateway. Such assignment is supposed to provide contiguous time slot numbers assigned for each relaying node to receive at, and thus saving the energy consumed in switching between on and off states. The other technique, namely depth assignment is based upon a depth-first-search (DFS) like. It tends to assign time slots contiguously over each route from the sensing node towards the gateway. Although this approach does not save the

energy of switching between on and off states as the breadth technique, it still avoids the buffer overflow problem. In most cases each received packet will not wait in the buffer of the relay node and will be forwarded in the next time slot.

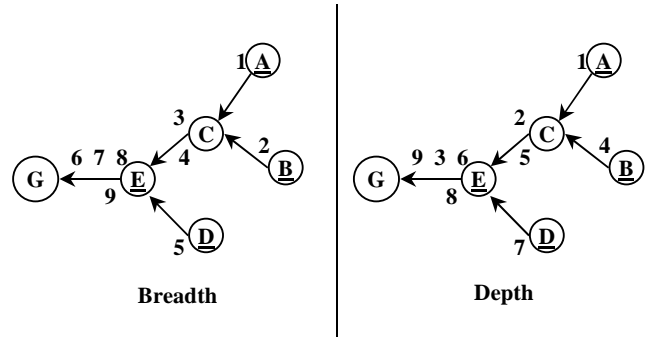


Figure 3: An example of slot assignment techniques

Figure 3 shows an example of the two slot assignment techniques. Nodes *A*, *B*, and *D* acts as sensor so they are assigned one slot for transmitting their data. Node *C* serves as a relay for nodes *A* and *D*, so it is assigned two slots. Node *E* acts as a sensor and a relay. It is assigned one slot for transmitting its own sensor data and 3 slots to relay other nodes' packets. In this example, the gateway informs each node of the slots it is going to receive packets from other nodes and the slots it can use to transmit the packets.

Now for the breadth technique, the gateway informs nodes *A*, *B* and *D* to transmit their packets at time slots 1, 2 and 5 respectively. For node *C*, it is informed to listen to packets at time slots 1 and 2, and to forward them at time slots 3 and 4 respectively. Node *E* is assigned to turn its receiver on at time slots 3-5 (corresponding to the transmission slots of nodes *C* and *D*) to receive packets. And that it can use time slot 6 to transmit its own packet, as well as time slots 7-9 to forward packets. It should be noted here that this slot assignment algorithm provides contiguous slot numbers for each node, thus reducing the energy needed to switch between on and off states. However, it might lead to instantaneous buffer overflow. For example, if node *E* in figure 3 has only a buffer for 2 packets, then it can happen that it receives, in slots 3-5 3 packets from nodes *C* and *D*. This may lead to packet drop due to buffer overflow. However, if transmission and receiving slots were

interleaved, this overflow cannot happen, as in the depth technique.

For the same example we apply the depth technique, as shown in the right side figure 3. For the packet generated by node *A*, it is assigned time slots 1 to send by node *A*, 2 to forward by node *C*, 3 to forward by node *E* to the Gateway. Similarly, packets generated by node *B* are assigned time slots 4, 5 and 6 to be sent by nodes *B*, *C* and *E* respectively. Similarly, node *D*'s packets are sent at time slots 7 and 8 by nodes *D* and *E* respectively. For node *E*'s own packets, they are assigned time slot 9. It is obvious that this technique avoids packet drops due to buffer overflow. However, nodes switch more frequently between on and off states.

The performance of both the depth and breadth techniques is compared via simulation, as reported in the next section. It is worth noting that our previous study [21][22] has demonstrated an improvement of an order of magnitude in the lifetime of the sensor network when combining our energy aware routing with the initial version of our new MAC protocol. Summary of the results of our study is given in appendix A.

3. Performance Evaluation

In this section, we use simulation to study the performance of the new MAC layer protocol. We further compare the performance of the two proposed techniques, the breadth and depth slot assignment. We use the following performance metrics:

- Packet drop count: the number of dropped data packets due to buffer overflow.
- Number of on/off state changes per sensor.
- Node lifetime: until the sensor energy level drops to zero.
- End-to-End Delay: the time it takes a data packet from the sensing node to the gateway.
- Throughput: the rate of data packets arrived to the gateway.
- Average energy consumed per packet: the average energy consumed in transmitting and receiving a data packet.

It should be noted that we considered a number of other energy-related performance metrics related to the network lifetime, time-to-network partitions, etc. In this paper we only report the results of the

above metrics. The reader is referred to [21][22] for the results for the other metrics. The following subsection describes the simulation environment followed by a summary and analysis of the simulation results.

3.1. Environment Setup

A cluster is set of 100 nodes placed randomly in a 1000×1000 meter square area. The gateway position is determined randomly within the cluster boundaries. A free space propagation channel model [16] is assumed with data rate set to 2Mb/s. Packet lengths are 10 kbit for data packets and 2 kbit for routing and refresh packets. The buffer size at each node is 15 packets. Each node has an initial energy of 2 joules. A node is considered non-functional if its energy level reaches zero.

For a node in the sensing state, packets are generated at a constant rate of one packet/second. This value is consistent with the specifications of the Acoustic Ballistic Module from SenTech Inc.[2]. Each data packet is time-stamped when it is generated to allow the calculation of average delay per packet. In addition, each packet has an energy field that is updated during the packet transmission to calculate the average energy per packet. A packet drop probability is taken equal to 0.01. This is used to make the simulator more realistic and to simulate the deviation of the gateway energy model.

We assume that the cluster is tasked with a target-tracking mission. The initial set of sensing nodes is chosen to be the nodes on the convex hull of the sensors of the cluster. The selected sensing nodes change as the target moves. Since targets are assumed to come from outside the cluster, the sensing circuitry of all boundary nodes is always turned on. The sensing circuitry of other nodes are usually turned off but can be turned on according to targets movement.

As mentioned before, rerouting occurs when a node's energy level falls below a percentage of its initial energy. This percentage is taken equal to 80%. Each time this threshold is reached, it is reset to 0.8 of its previous value.

For energy-consumption, we used the communication energy consumption model used in [12][19], the computation energy consumption model used in [19][20], and the sensing energy-consumption model used in [18].

Targets are assumed to start at a random position outside the convex hull. These targets are characterized by having a constant speed chosen uniformly from the range four meters per second to six meters/second and a constant direction chosen uniformly depending on the initial target position in order for the target to cross the convex hull region. For the purpose of this experiment we assume that only one target will be active at any time. Each target remains active until it leaves the deployment region area. In this case, a new target is generated.

3.2. Performance Results

Now we describe the results of our experiments based on the performance metric. All the performance metrics are plotted against increasing buffer sizes at the sensor nodes.

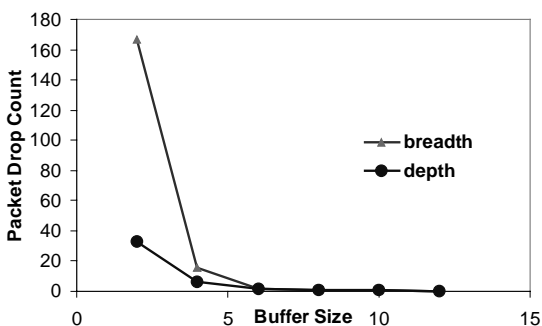


Figure 4: Effect of buffer size on packet drop count

In Figure 4, we can see the expected advantage of the depth technique over the breadth. The number of packets dropped due to buffer overflow in the case of the depth slot assignment is not zero. This is because either we don't know when a node will generate its data, if it is a sensing node, or a node retains its buffer when the slot assignment changes.

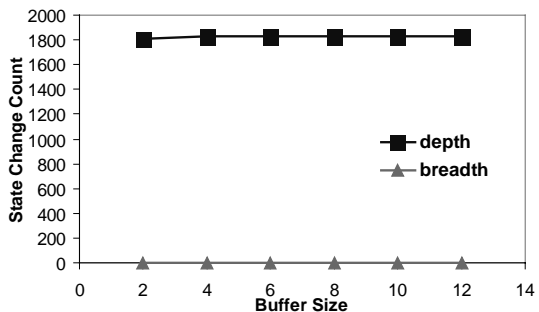


Figure 5: Effect of Buffer size on transmitter state change count

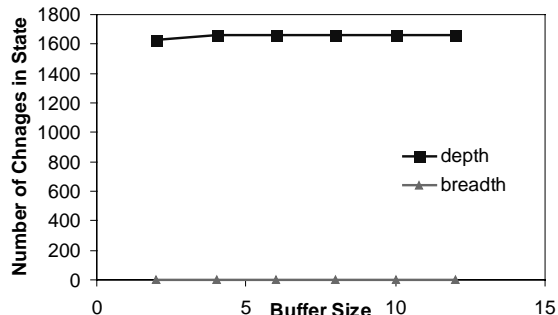


Figure 6: Effect of Buffer size on receiver state change count

Figures 5 and 6 show that for the breadth method, the number of changes in state is zero. Thus the breadth technique saves energy. The number of state changes for the transmitter is higher than for the receiver. This is expected as each node at least transmits what it receives (if it doesn't generate new packets.) So the number of transmission slots is larger than the number of receiving slots. Thus it is more probable to change state while you are transmitting than when you are receiving.

As the buffer size increases, the number of packets that reaches the gateway increases slightly leading to a more accurate model at the gateway. This also explains the decrease of the average energy consumed per packet shown in figure 10.

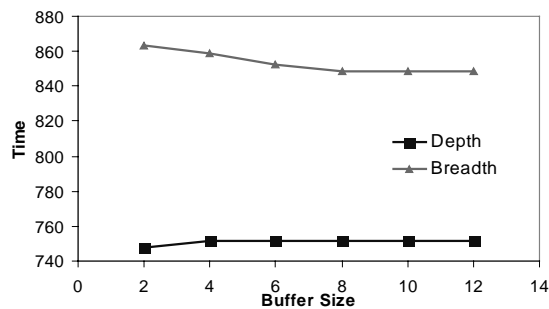


Figure 7: Effect of Buffer size on Node lifetime

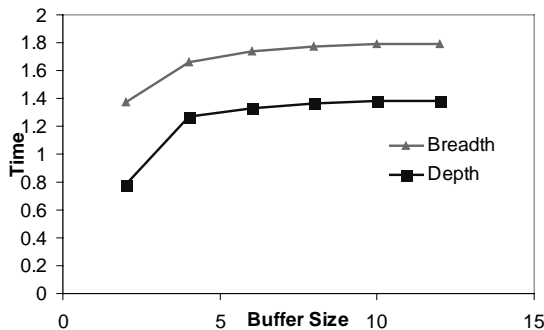


Figure 8: End-to-End Delay

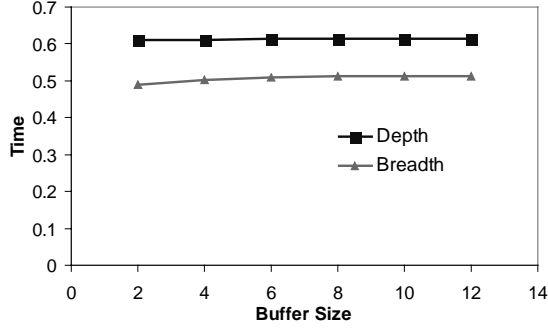


Figure 9: Effect of Buffer size on Throughput

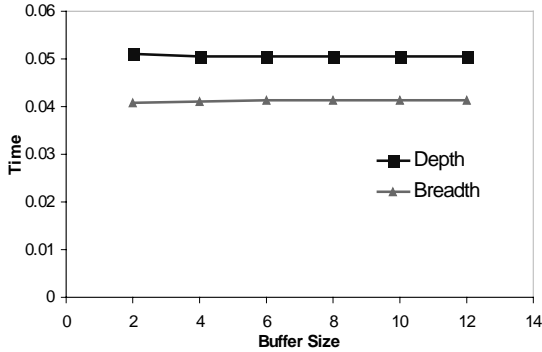


Figure 10: Effect of Buffer size on Energy per

As expected in figure 8, when the buffer size increases, the average delay per packet increases due to the increased queuing delay. However in figure 9, the throughput does not decrease as less number of packets is dropped due to more available buffer size.

Average delay per packet is lower in case of depth technique as a packet never waits for a slot if there are no cells in buffer; e.g. a packet goes in slot 3-4-5-6 to reach the gateway. In case of breadth technique, a cell may wait for a slot in the next cycle if a packet arrives after the contiguous transmission period has ended.

Throughput is lower in case of breadth technique as the number of packets dropped is higher

Lifetime in case of breadth technique is higher as more packets are dropped and not forwarded saving the energy of the nodes, but with lower throughput. However, it decreases in case of breadth technique as more packets arrive leading to consuming the energy of the nodes (note that the other effect which is more packets means more accurate model is not as effective as the other effect)

In summary, the above results show that the breadth technique is better when the energy required for changing the sensor's state between on and off is critical. However, the depth technique is more

reliable regarding packet delivery since it avoids packet drops due to buffer overflow. The depth technique is also superior with respect to end-to-end delay as well as throughput.

4. Conclusion and Potential Trends

In this paper, we have presented a novel approach for an energy-aware management of sensor networks. A gateway node acts as a cluster-based centralized network manager that sets routes for sensor data, monitors latency throughout the cluster, and arbitrates medium access among sensors. The gateway tracks energy usage at every sensor node and changes in the mission and the environment. The gateway configures the sensors and the network to operate efficiently in order to extend the life of the network.

We have also presented in details a new MAC layer protocol. We have proposed two major techniques for slot assignment. Simulation results demonstrate a comparative evaluation of the breadth and depth slot assignment techniques with increasing buffer sizes. The simulation results demonstrated that the breadth technique is recommended in case the energy consumed for changing the sensor's state is high. On the other hand, the depth technique offers more reliable data packet delivery since it is more tolerant to packet drops caused by buffer overflow. The depth technique also gives better results regarding end-to-end delay as well as throughput.

Our future plan includes extending the routing model to allow for node mobility. We would like also to study approaches for clustering the sensor network and inter-cluster communication, i.e. scaling our approach. We are interested in studying dynamic and reservation-based TDMA slot assignment techniques in the MAC layer. Another potential issue is to allow event driven/on-demand sensing scenario instead of the continuous periodic sensed data flow. Finally, the approach can be extended to be applicable to other sensor based networks, e.g. automobile, airplane control, industrial control.

References

- [1] R. Burne, I. Kadar, J. Whitson and E. Eadan, "A Self-Organizing, Cooperative UGS Network for Target Tracking," in the *Proceedings of the SPIE Conference on Unattended Ground Sensor Technologies and Applications II*, Orlando Florida, April 2000.
- [2] "Data sheet for the Acoustic Ballistic Module", SenTech Inc., <http://www.sentech-acoustic.com/>
- [3] T. Srikanth and S. Toueg, "Optimal Clock Synchronization," *Journal of the ACM*, Vol. 34, No. 3, pp. 626-645, July 1987.
- [4] A. Buczak and V. Jamalabad, "Self-organization of a Heterogeneous Sensor Network by Genetic Algorithms," *Intelligent Engineering Systems Through Artificial Neural Networks*, C.H. Dagli, et al. (eds.), Vol. 8, pp. 259-264, ASME Press, New York, 1998.
- [5] P.J.M. Havinga, G.J.M. Smit, "Design Techniques for Low Power Systems", *Journal of Systems Architecture*, Vol. 46, No. 1, 2000.
- [6] P. Havinga, G. Smit, M. Bos, "Energy efficient adaptive wireless network design", *The Fifth Symposium on Computers and Communications (ISCC'00)*, Antibes, France, July 2000.
- [7] P. Havinga, G. Smit, "Energy-efficient TDMA medium access control protocol scheduling", *Proceedings Asian International Mobile Computing Conference (AMOC 2000)*, Nov. 2000.
- [8] Suresh Singh and C.S. Raghavendra, "PAMAS: Power Aware Multi-Access protocol with Signaling for Ad Hoc Networks", *ACM Computer Communications Review*, July 1998.
- [9] Christian Röhl, Hagen Woesner, Adam Wolisz, "A Short Look on Power Saving Mechanisms in the Wireless LAN Standard Draft IEEE 802.11," *Proc. of the 6th WINLAB Workshop on third generation Wireless Systems*, New Brunswick, New Jersey, March 1997.
- [10] Shugong Xu and Tarek Saadawi, "Does the IEEE 802.11 MAC Protocol Work Well in Multihop Wireless Ad Hoc Networks?" *IEEE Communications Magazine*, June 2001.
- [11] C-K. Toh, "Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless Ad Hoc Networks," *IEEE Communications Magazine*, June 2001.
- [12] W. Rabiner Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocols for Wireless Microsensor Networks," *Hawaii International Conference on System Sciences (HICSS '00)*, January 2000.
- [13] V. Rodoplu and T. H. Meng, "Minimum energy mobile wireless networks", *IEEE Jour. Selected Areas Comm.*, vol. 17, no. 8, pp. 1333-1344, Aug. 1999.
- [14] Jae-Hwan Chang, Leandros Tassiulas, "Energy Conserving Routing in Wireless Ad-hoc Networks", *IEEE Infocom 2000*.
- [15] Chen S., "Routing Support for providing guaranteed End-To-End Quality of Service," Ph.D. Thesis Dissertation, University of Illinois at Urbana-Champaign, 1999.
- [16] Dijkstra, E. W., "A Note on Two Problems in Connection with Graphs," *Numerische Mathematik*, 1:269-271, 1959.
- [17] J.B. Andresen, T.S. Rappaport, and S. Yoshida, "Propagation Measurements and Models for Wireless Communications Channels," *IEEE Communications Magazine*, Vol. 33, No. 1, January 1995.
- [18] Manish Bhardwaj, Timothy Garnett, and Anantha P. Chandrakasan, "Upper Bounds on the Lifetime of Sensor Networks", *In Proceedings of ICC 2001*, June 2001. (12)
- [19] W. Heinzelman, A. Sinha, A. Wang, and A. Chandrakasan, "Energy-Scalable Algorithms and Protocols for Wireless Microsensor Networks," *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP '00)*, June 2000.
- [20] Amit Sinha and A. Chandrakasan, "Energy Aware Software", *Proceedings of the 13th International Conference on VLSI Design*, pp. 50-55, Calcutta, India. January 2000.
- [21] M. Younis, M. Youssef and K. Arisha, "Energy-aware routing in cluster-based sensor networks," *submitted*.
- [22] M. Youssef, M. Younis and K. Arisha, "A Constrained Shortest-Path Energy-aware Routing for Wireless Sensor Networks," *Wireless Communication and Networking Conference (WCNC)*, Orlando, Florida, March 2002.

Appendix A

A.1. Effect of Interaction between Routing Protocol and MAC Layer

We run an experiment to determine the effect of using the interaction of the routing algorithm decision and turning on and off the receiver at the MAC layer. Figures 11 through 13 show the results for different performance metrics. It is clear that turning the receiver circuitry on and off based on the interaction between the routing algorithm and the MAC layer has significant effect on saving power and hence the lifetime of the network. Moreover, the interaction between the routing algorithm and the MAC layer has a positive effect on the average delay per packet. With the receiver always turned on, nodes die quickly leading to selection of longer paths which increases the average delay per packet. The results show one order of magnitude enhancement in the network lifetime, 78% enhancement in the average energy consumed per packet, and 23% enhancement in the average delay per packet.

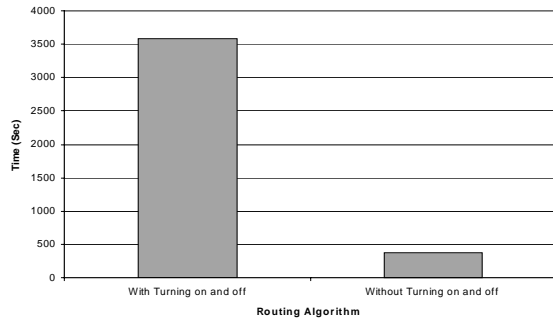


Figure 11: Effect on network lifetime.

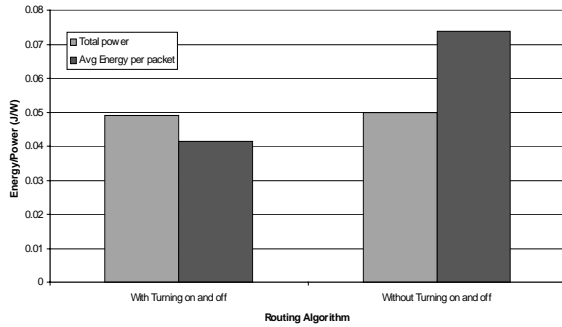


Figure 12: Effect on power metrics.

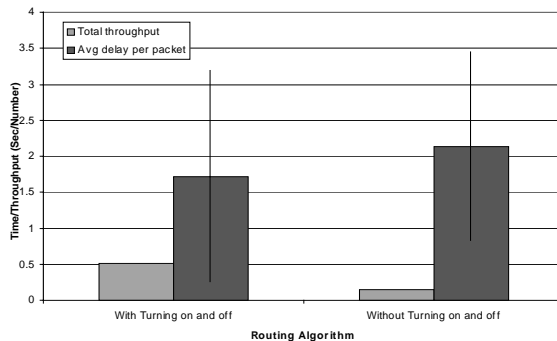


Figure 13: Effect on throughput and delay metrics.

A.2. Comparison between Routing Algorithms

We ran a set of experiments to compare the performance of our approach with other routing algorithms. The results are shown in figures 14 through 17. The figures show that the new algorithm gives a relatively good performance for all the metrics. Other algorithms may slightly outperform our algorithm in some metrics. However, the same algorithms perform poorly on other metrics. For example, the minimum distance routing algorithm gives a 1.57 improvement factor over the new algorithm in terms of average delay per packet. However, our algorithm outperforms this algorithm

by a factor of 13.91 in terms of time to network partitioning, as indicated in Fig. 14.

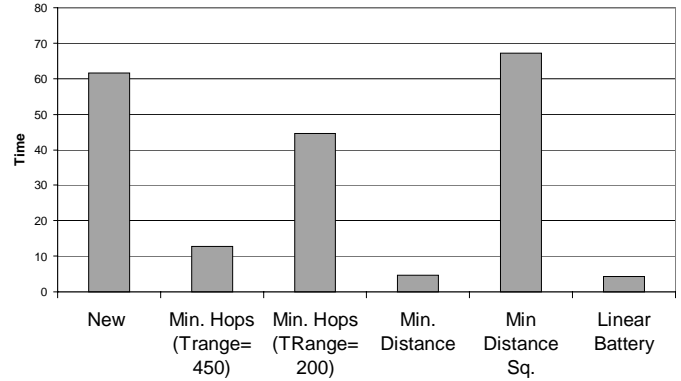


Figure 14: Time to network partition for routing algorithms.

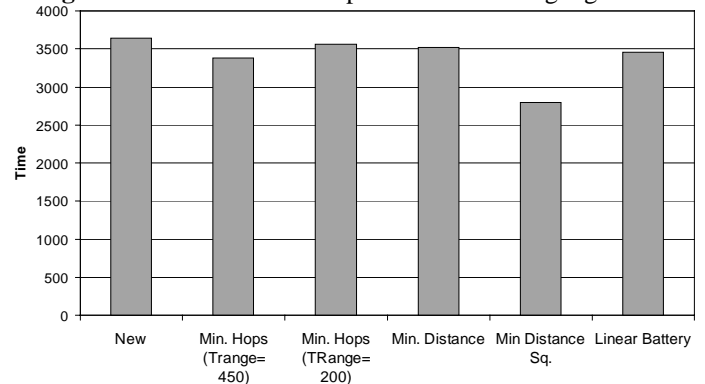


Figure 15: Time for last node to die under various algorithms.

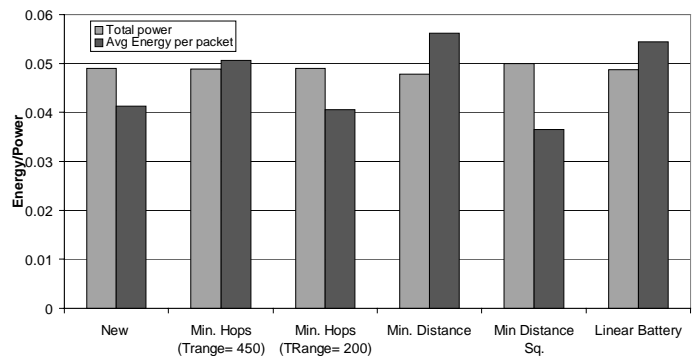


Figure 16: Comparing energy metrics among routing algorithms.

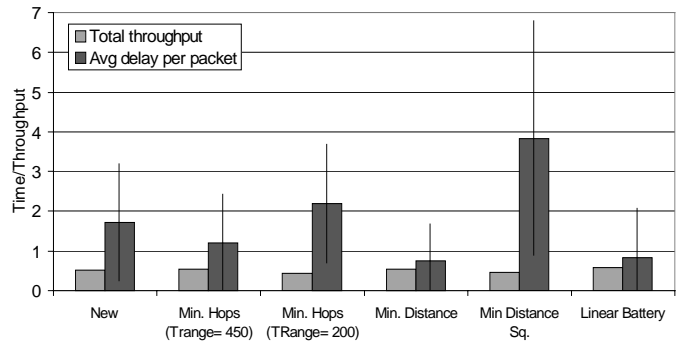


Figure 17: Throughput and delay for routing algorithms.