Energy-Efficient Communication Protocol for Wireless Microsensor Networks

Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan
Massachusetts Institute of Technology
Cambridge, MA 02139
{*wendi, anantha, hari*}*@mit.edu*

Abstract

Wireless distributed microsensor systems will enable the reliable monitoring of a variety of environments for both civil and military applications. In this paper, we look at communication protocols, which can have significant impact on the overall energy dissipation of these networks. Based on our findings that the conventional protocols of direct transmission, minimum-transmission-energy, multi-hop routing, and static clustering may not be optimal for sensor networks, we propose LEACH (Low-Energy Adaptive Clustering Hierarchy), a clustering-based protocol that utilizes randomized rotation of local cluster base stations (cluster-heads) to evenly distribute the energy load among the sensors in the network. LEACH uses localized coordination to enable scalability and robustness for dynamic networks, and incorporates data fusion into the routing protocol to reduce the amount of information that must be transmitted to the base station. Simulations show that LEACH can achieve as much as a factor of 8 reduction in energy dissipation compared with conventional routing protocols. In addition, LEACH is able to distribute energy dissipation evenly throughout the sensors, doubling the useful system lifetime for the networks we simulated.

1. Introduction

Recent advances in MEMS-based sensor technology, low-power analog and digital electronics, and low-power RF design have enabled the development of relatively inexpensive and low-power wireless microsensors [2, 3, 4]. These sensors are not as reliable or as accurate as their expensive macrosensor counterparts, but their size and cost enable applications to network hundreds or thousands of these microsensors in order to achieve high quality, fault-tolerant sensing networks. Reliable environment monitoring is important in a variety of commercial and military applications. For example, for a security system, acoustic, seismic, and video sensors can be used to form an ad hoc network to detect intrusions. Microsensors can also be used to monitor machines for fault detection and diagnosis.

Microsensor networks can contain hundreds or thousands of sensing nodes. It is desirable to make these nodes as cheap and energy-efficient as possible and rely on their large numbers to obtain high quality results. Network protocols must be designed to achieve fault tolerance in the presence of individual node failure while minimizing energy consumption. In addition, since the limited wireless channel bandwidth must be shared among all the sensors in the network, routing protocols for these networks should be able to perform local collaboration to reduce bandwidth requirements.

Eventually, the data being sensed by the nodes in the network must be transmitted to a control center or base station, where the end-user can access the data. There are many possible models for these microsensor networks. In this work, we consider microsensor networks where:

- The base station is fixed and located far from the sensors.
- All nodes in the network are homogeneous and energy-constrained.

Thus, communication between the sensor nodes and the base station is expensive, and there are no "high-energy" nodes through which communication can proceed. This is the framework for MIT's μ-AMPS project, which focuses on innovative energy-optimized solutions at all levels of the system hierarchy, from the physical layer and communication protocols up to the application layer and efficient DSP design for microsensor nodes.

Sensor networks contain too much data for an end-user to process. Therefore, automated methods of combining or *aggregating* the data into a small set of meaningful information is required [7, 8]. In addition to helping avoid information overload, data aggregation, also known as *data fusion*, can combine several unreliable data measurements to produce a more accurate signal by enhancing the common signal and reducing the uncorrelated noise. The classification performed on the aggregated data might be performed by a human operator or automatically. Both the method of performing data aggregation and the classification algorithm are application-specific. For example, acoustic signals are often combined using a *beamforming* algorithm [5, 17] to reduce several signals into a single signal that contains the relevant information of all the individual signals. Large energy gains can be achieved by performing the data fusion or classification algorithm locally, thereby requiring much less data to be transmitted to the base station.

By analyzing the advantages and disadvantages of conventional routing protocols using our model of sensor networks, we have developed LEACH (Low-Energy Adaptive Clustering Hierarchy), a clustering-based protocol that minimizes energy dissipation in sensor networks. The key features of LEACH are:

- Localized coordination and control for cluster set-up and operation.
- Randomized rotation of the cluster "base stations" or "cluster-heads" and the corresponding clusters.
- Local compression to reduce global communication.

The use of clusters for transmitting data to the base station leverages the advantages of small transmit distances for most nodes, requiring only a few nodes to transmit far distances to the base station. However, LEACH outperforms classical clustering algorithms by using adaptive clusters and rotating cluster-heads, allowing the energy requirements of the system to be distributed among all the sensors. In addition, LEACH is able to perform local computation in each cluster to reduce the amount of data that must be transmitted to the base station. This achieves a large reduction in the energy dissipation, as computation is much cheaper than communication.

2. First Order Radio Model

Currently, there is a great deal of research in the area of low-energy radios. Different assumptions about the radio characteristics, including energy dissipation in the transmit and receive modes, will change the advantages of different protocols. In our work, we assume a simple model where the radio dissipates $E_{elec} = 50$ nJ/bit to run the transmitter or receiver circuitry and $\epsilon_{amp} = 100$ pJ/bit/m^2 for the transmit amplifier to achieve an acceptable $\frac{E_b}{N_o}$ (see Figure 1 and Table 1). These parameters are slightly better than the current state-of-the-art in radio design[1]. We also assume an

[1] For example, the Bluetooth initiative [1] specifies 700 Kbps radios that operate at 2.7 V and 30 mA, or 115 nJ/bit.

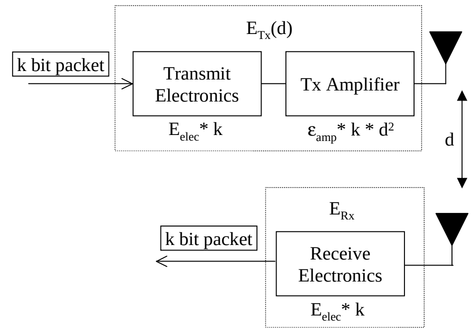

Figure 1. First order radio model.

Table 1. Radio characteristics.

Operation	Energy Dissipated
Transmitter Electronics ($E_{Tx-elec}$) Receiver Electronics ($E_{Rx-elec}$) ($E_{Tx-elec} = E_{Rx-elec} = E_{elec}$)	50 nJ/bit
Transmit Amplifier (ϵ_{amp})	100 pJ/bit/m^2

r^2 energy loss due to channel transmission. Thus, to transmit a k-bit message a distance d using our radio model, the radio expends:

$$\begin{aligned} E_{Tx}(k,d) &= E_{Tx-elec}(k) + E_{Tx-amp}(k,d) \\ E_{Tx}(k,d) &= E_{elec} * k + \epsilon_{amp} * k * d^2 \end{aligned} \tag{1}$$

and to receive this message, the radio expends:

$$\begin{aligned} E_{Rx}(k) &= E_{Rx-elec}(k) \\ E_{Rx}(k) &= E_{elec} * k \end{aligned} \tag{2}$$

For these parameter values, receiving a message is not a low cost operation; the protocols should thus try to minimize not only the transmit distances but also the number of transmit and receive operations for each message.

We make the assumption that the radio channel is symmetric such that the energy required to transmit a message from node A to node B is the same as the energy required to transmit a message from node B to node A for a given SNR. For our experiments, we also assume that all sensors are sensing the environment at a fixed rate and thus always have data to send to the end-user. For future versions of our protocol, we will implement an "event-driven" simulation, where sensors only transmit data if some event occurs in the environment.

3. Energy Analysis of Routing Protocols

There have been several network routing protocols proposed for wireless networks that can be examined in the context of wireless sensor networks. We examine two such protocols, namely direct communication with the base station and minimum-energy multi-hop routing using our sensor network and radio models. In addition, we discuss a conventional clustering approach to routing and the drawbacks of using such an approach when the nodes are all energy-constrained.

Using a direct communication protocol, each sensor sends its data directly to the base station. If the base station is far away from the nodes, direct communication will require a large amount of transmit power from each node (since d in Equation 1 is large). This will quickly drain the battery of the nodes and reduce the system lifetime. However, the only receptions in this protocol occur at the base station, so if either the base station is close to the nodes, or the energy required to receive data is large, this may be an acceptable (and possibly optimal) method of communication.

The second conventional approach we consider is a "minimum-energy" routing protocol. There are several power-aware routing protocols discussed in the literature [6, 9, 10, 14, 15]. In these protocols, nodes route data destined ultimately for the base station through intermediate nodes. Thus nodes act as routers for other nodes' data in addition to sensing the environment. These protocols differ in the way the routes are chosen. Some of these protocols [6, 10, 14], only consider the energy of the transmitter and neglect the energy dissipation of the receivers in determining the routes. In this case, the intermediate nodes are chosen such that the transmit amplifier energy (e.g., $E_{Tx-amp}(k,d) = \epsilon_{amp} * k * d^2$) is minimized; thus node A would transmit to node C through node B if and only if:

$$E_{Tx-amp}(k, d = d_{AB}) + E_{Tx-amp}(k, d = d_{BC}) < E_{Tx-amp}(k, d = d_{AC}) \quad (3)$$

or

$$d_{AB}^2 + d_{BC}^2 < d_{AC}^2 \quad (4)$$

However, for this minimum-transmission-energy (MTE) routing protocol, rather than just one (high-energy) transmit of the data, each data message must go through n (low-energy) transmits and n receives. Depending on the relative costs of the transmit amplifier and the radio electronics, the total energy expended in the system might actually be greater using MTE routing than direct transmission to the base station.

To illustrate this point, consider the linear network shown in Figure 2, where the distance between the nodes is r. If we consider the energy expended transmitting a single k-bit message from a node located a distance nr from the base station using the direct communication approach and Equations 1 and 2, we have:

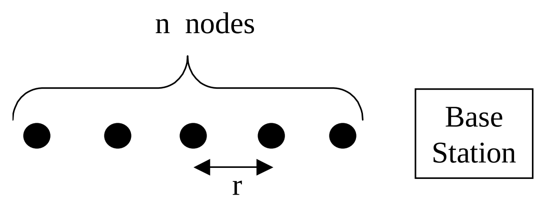

Figure 2. Simple linear network.

$$\begin{aligned} E_{direct} &= E_{Tx}(k, d = n * r) \\ &= E_{elec} * k + \epsilon_{amp} * k * (nr)^2 \\ &= k(E_{elec} + \epsilon_{amp} n^2 r^2) \end{aligned} \quad (5)$$

In MTE routing, each node sends a message to the closest node on the way to the base station. Thus the node located a distance nr from the base station would require n transmits a distance r and $n-1$ receives.

$$\begin{aligned} E_{MTE} &= n * E_{Tx}(k, d = r) + (n-1) * E_{Rx}(k) \\ &= n(E_{elec} * k + \epsilon_{amp} * k * r^2) + (n-1) * E_{elec} * k \\ &= k((2n-1)E_{elec} + \epsilon_{amp} n r^2) \end{aligned} \quad (6)$$

Therefore, direct communication requires *less* energy than MTE routing if:

$$\begin{aligned} E_{direct} &< E_{MTE} \\ E_{elec} + \epsilon_{amp} n^2 r^2 &< (2n-1)E_{elec} + \epsilon_{amp} n r^2 \\ \frac{E_{elec}}{\epsilon_{amp}} &> \frac{r^2 n}{2} \end{aligned} \quad (7)$$

Using Equations 1 - 6 and the random 100-node network shown in Figure 3, we simulated transmission of data from every node to the base station (located 100 m from the closest sensor node, at (x=0, y=-100)) using MATLAB. Figure 4 shows the total energy expended in the system as the network diameter increases from 10 m × 10 m to 100 m × 100 m and the energy expended in the radio electronics (i.e., E_{elec}) increases from 10 nJ/bit to 100 nJ/bit, for the scenario where each node has a 2000-bit data packet to send to the base station. This shows that, as predicted by our analysis above, when transmission energy is on the same order as receive energy, which occurs when transmission distance is short and/or the radio electronics energy is high, direct transmission is more energy-efficient on a global scale than MTE routing. Thus the most energy-efficient protocol to use depends on the network topology and radio parameters of the system.

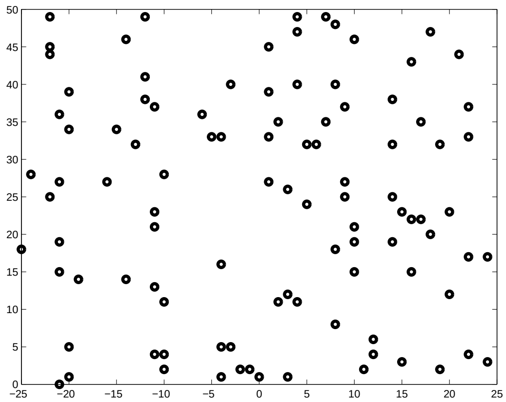

Figure 3. 100-node random network.

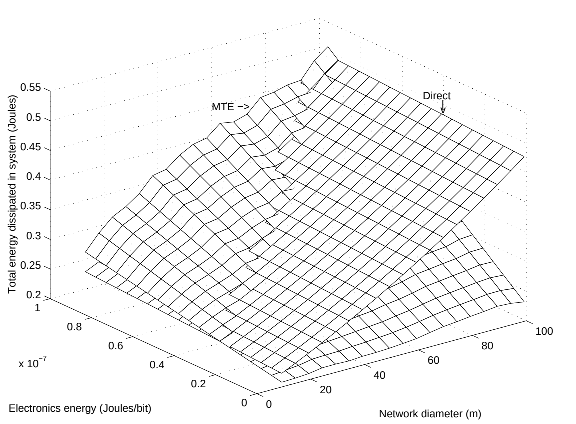

Figure 4. Total energy dissipated in the 100-node random network using direct communication and MTE routing (i.e., E_{direct} and E_{MTE}). $\epsilon_{amp} = 100$ pJ/bit/m^2, and the messages are 2000 bits.

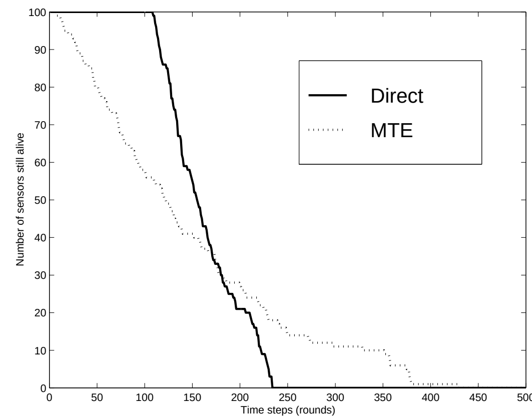

Figure 5. System lifetime using direct transmission and MTE routing with 0.5 J/node.

It is clear that in MTE routing, the nodes closest to the base station will be used to route a large number of data messages to the base station. Thus these nodes will die out quickly, causing the energy required to get the remaining data to the base station to increase and more nodes to die. This will create a cascading effect that will shorten system lifetime. In addition, as nodes close to the base station die, that area of the environment is no longer being monitored. To prove this point, we ran simulations using the random 100-node network shown in Figure 3 and had each sensor send a 2000-bit data packet to the base station during each time step or "round" of the simulation. After the energy dissipated in a given node reached a set threshold, that node was considered dead for the remainder of the simulation. Figure 5 shows the number of sensors that remain alive after each round for direct transmission and MTE routing with each node initially given 0.5 J of energy. This plot shows that nodes die out quicker using MTE routing than direct transmission. Figure 6 shows that nodes closest to the base station are the ones to die out first for MTE routing, whereas nodes furthest from the base station are the ones to die out first for direct transmission. This is as expected, since the nodes close to the base station are the ones most used as "routers" for other sensors' data in MTE routing, and the nodes furthest from the base station have the largest transmit energy in direct communication.

A final conventional protocol for wireless networks is clustering, where nodes are organized into clusters that communicate with a local base station, and these local base stations transmit the data to the global base station, where it is accessed by the end-user. This greatly reduces the distance nodes need to transmit their data, as typically the local base station is close to all the nodes in the cluster.

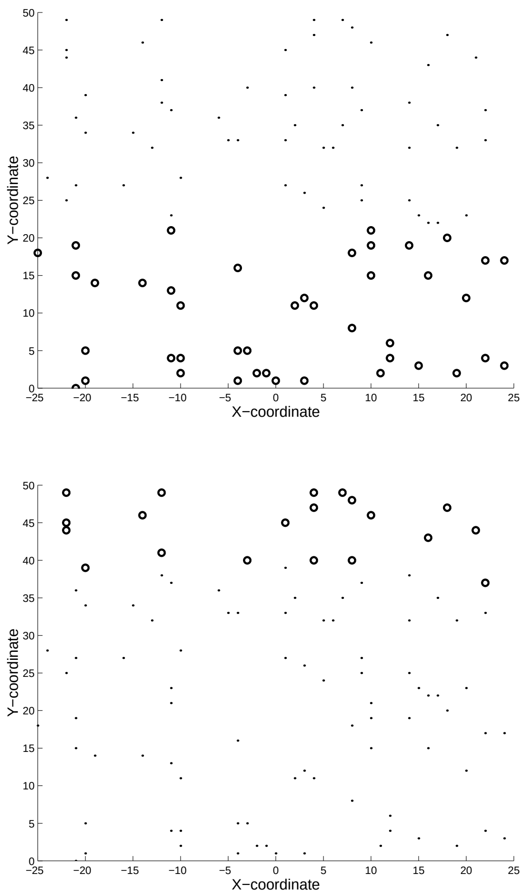

Figure 6. Sensors that remain alive (circles) and those that are dead (dots) after 180 rounds with 0.5 J/node for (a) direct transmission and (b) MTE routing.

Thus, clustering appears to be an energy-efficient communication protocol. However, the local base station is assumed to be a high-energy node; if the base station is an energy-constrained node, it would die quickly, as it is being heavily utilized. Thus, conventional clustering would perform poorly for our model of microsensor networks. The Near Term Digital Radio (NTDR) project [12, 16], an army-sponsored program, employs an adaptive clustering approach, similar to our work discussed here. In this work, cluster-heads change as nodes move in order to keep the network fully connected. However, the NTDR protocol is designed for long-range communication, on the order of 10s of kilometers, and consumes large amounts of power, on the order of 10s of Watts. Therefore, this protocol also does not fit our model of sensor networks.

4. LEACH: Low-Energy Adaptive Clustering Hierarchy

LEACH is a self-organizing, adaptive clustering protocol that uses randomization to distribute the energy load evenly among the sensors in the network. In LEACH, the nodes organize themselves into local clusters, with one node acting as the local base station or *cluster-head*. If the cluster-heads were chosen a priori and fixed throughout the system lifetime, as in conventional clustering algorithms, it is easy to see that the unlucky sensors chosen to be cluster-heads would die quickly, ending the useful lifetime of all nodes belonging to those clusters. Thus LEACH includes randomized rotation of the high-energy cluster-head position such that it rotates among the various sensors in order to not drain the battery of a single sensor. In addition, LEACH performs local data fusion to “compress” the amount of data being sent from the clusters to the base station, further reducing energy dissipation and enhancing system lifetime.

Sensors elect themselves to be local cluster-heads at any given time with a certain probability. These cluster-head nodes broadcast their status to the other sensors in the network. Each sensor node determines to which cluster it wants to belong by choosing the cluster-head that requires the minimum communication energy[2]. Once all the nodes are organized into clusters, each cluster-head creates a schedule for the nodes in its cluster. This allows the radio components of each non-cluster-head node to be turned off at all times except during its transmit time, thus minimizing the energy dissipated in the individual sensors. Once the cluster-head has all the data from the nodes in its cluster, the cluster-head node aggregates the data and then transmits the compressed data to the base station. Since the base station is far away in the scenario we are examining, this is a high energy transmission. However, since there are only a few cluster-heads, this only affects a small number of nodes.

As discussed previously, being a cluster-head drains the battery of that node. In order to spread this energy usage over multiple nodes, the cluster-head nodes are not fixed; rather, this position is self-elected at different time intervals. Thus a set C of nodes might elect themselves cluster-heads at time t_1, but at time $t_1 + d$ a new set C' of nodes elect themselves as cluster-heads, as shown in Figure 7. The decision to become a cluster-head depends on the amount of energy left at the node. In this way, nodes with more energy remaining will perform the energy-intensive functions of the network. Each node makes its decision about whether to be a cluster-head independently of the other nodes in the

[2] Note that typically this will be the cluster-head closest to the sensor. However, if there is some obstacle impeding the communication between two physically close nodes (e.g., a building, a tree, etc.) such that communication with another cluster-head, located further away, is easier, the sensor will choose the cluster-head that is spatially further away but “closer” in a communication sense.

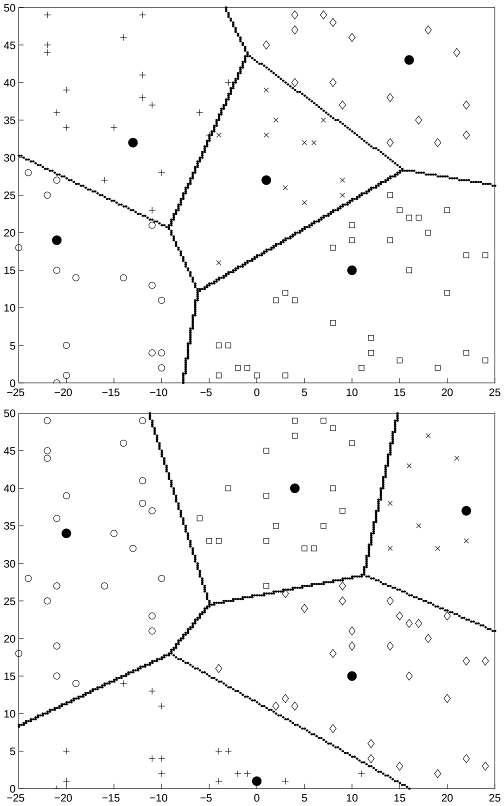

Figure 7. Dynamic clusters: (a) cluster-head nodes = C at time t_1 (b) cluster-head nodes = $C\prime$ at time $t_1 + d$. All nodes marked with a given symbol belong to the same cluster, and the cluster-head nodes are marked with a $\bullet$.

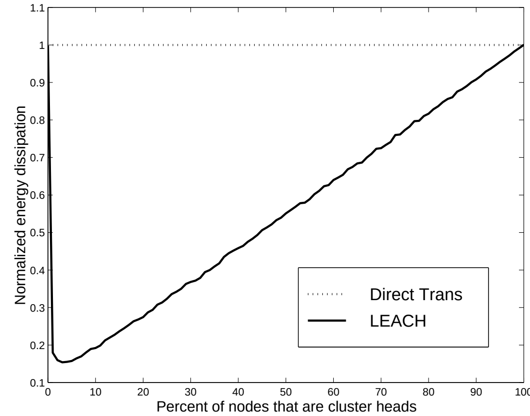

Figure 8. Normalized total system energy dissipated versus the percent of nodes that are cluster-heads. Note that direct transmission is equivalent to 0 nodes being cluster-heads or all the nodes being cluster-heads.

network and thus no extra negotiation is required to determine the cluster-heads.

The system can determine, a priori, the optimal number of clusters to have in the system. This will depend on several parameters, such as the network topology and the relative costs of computation versus communication. We simulated the LEACH protocol for the random network shown in Figure 3 using the radio parameters in Table 1 and a computation cost of 5 nJ/bit/message to fuse 2000-bit messages while varying the percentage of total nodes that are cluster-heads. Figure 8 shows how the energy dissipation in the system varies as the percent of nodes that are cluster-heads is changed. Note that 0 cluster-heads and 100% cluster-heads is the same as direct communication. From this plot, we find that there exists an optimal percent of nodes $\hat{N}$ that should be cluster-heads. If there are fewer than $\hat{N}$ cluster-heads, some nodes in the network have to transmit their data very far to reach the cluster-head, causing the global energy in the system to be large. If there are more than $\hat{N}$ cluster-heads, the distance nodes have to transmit to reach the nearest cluster-head does not reduce substantially, yet there are more cluster-heads that have to transmit data the long-haul distances to the base station, and there is less compression being performed locally. For our system parameters and topology, $\hat{N} = 5\%$.

Figure 8 also shows that LEACH can achieve over a factor of 7 reduction in energy dissipation compared to direct communication with the base station, when using the optimal number of cluster-heads. The main energy savings of the LEACH protocol is due to combining lossy compression with the data routing. There is clearly a trade-off between the quality of the output and the amount of compression achieved. In this case, some data from the individual signals is lost, but this results in a substantial reduction of the overall energy dissipation of the system.

We simulated LEACH (with 5% of the nodes being cluster-heads) using MATLAB with the random network shown in Figure 3. Figure 9 shows how these algorithms compare using $E_{elec} = 50$ nJ/bit as the diameter of the network is increased. This plot shows that LEACH achieves between 7x and 8x reduction in energy compared with direct communication and between 4x and 8x reduction in energy compared with MTE routing. Figure 10 shows the amount of energy dissipated using LEACH versus using direct communication and LEACH versus MTE routing as the network diameter is increased and the electronics energy varies. This figure shows the large energy savings achieved using LEACH for most of the parameter space.

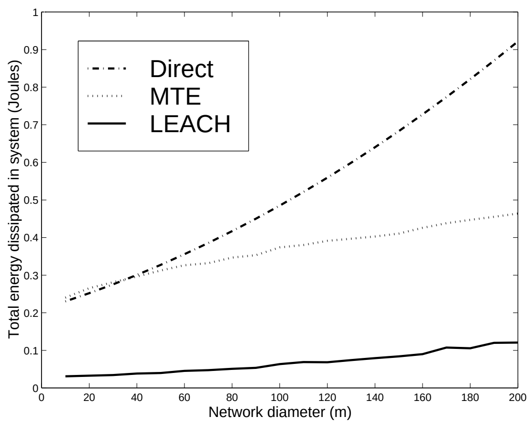

Figure 9. Total system energy dissipated using direct communication, MTE routing and LEACH for the 100-node random network shown in Figure 3. $E_{elec} = 50$ nJ/bit, $\epsilon_{amp} = 100$ pJ/bit/m^2, and the messages are 2000 bits.

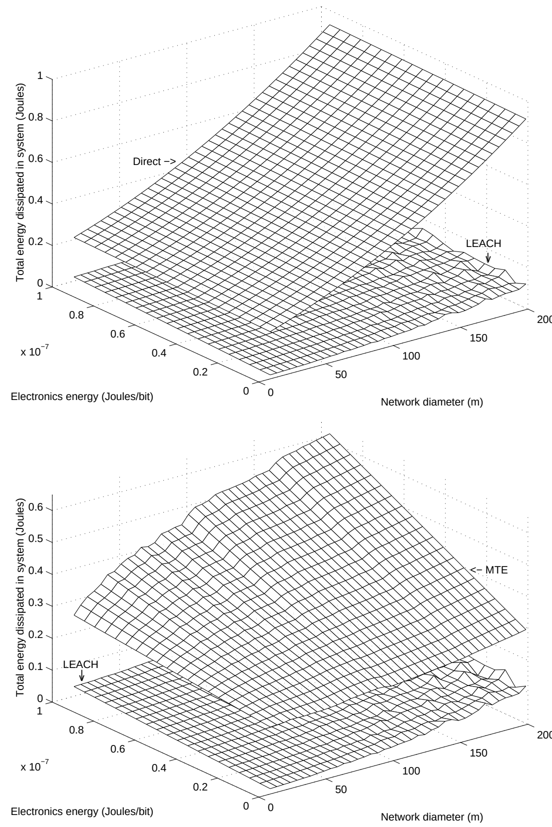

Figure 10. Total system energy dissipated using (a) direct communication and LEACH and (b) MTE routing and LEACH for the random network shown in Figure 3. $\epsilon_{amp} = 100$ pJ/bit/m^2, and the messages are 2000 bits.

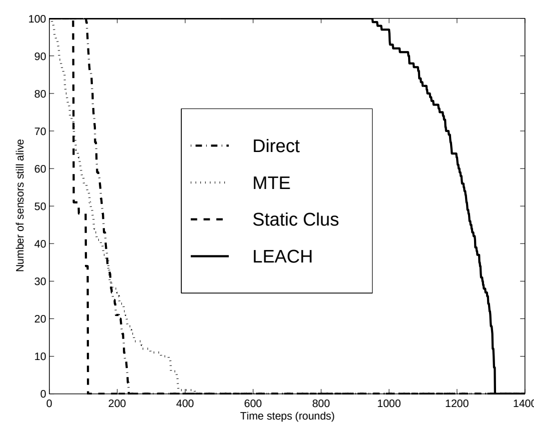

Figure 11. System lifetime using direct transmission, MTE routing, static clustering, and LEACH with 0.5 J/node.

In addition to reducing energy dissipation, LEACH successfully distributes energy-usage among the nodes in the network such that the nodes die randomly and at essentially the same rate. Figure 11 shows a comparison of system lifetime using LEACH versus direct communication, MTE routing, and a conventional static clustering protocol, where the cluster-heads and associated clusters are chosen initially and remain fixed and data fusion is performed at the cluster-heads, for the network shown in Figure 3. For this experiment, each node was initially given 0.5 J of energy. Figure 11 shows that LEACH more than doubles the useful system lifetime compared with the alternative approaches. We ran similar experiments with different energy thresholds and found that no matter how much energy each node is given, it takes approximately 8 times longer for the first node to die and approximately 3 times longer for the last node to die in LEACH as it does in any of the other protocols. The data from these experiments is shown in Table 2. The advantage of using dynamic clustering (LEACH) versus static clustering can be clearly seen in Figure 11. Using a static clustering algorithm, as soon as the cluster-head node dies, all nodes from that cluster effectively die since there is no way to get their data to the base station. While these simulations do not account for the setup time to configure the dynamic clusters (nor do they account for any necessary routing start-up costs or updates as nodes die), they give a good first order approximation of the lifetime extension we can achieve using LEACH.

Another important advantage of LEACH, illustrated in Figure 12, is the fact that nodes die in essentially a "ran-

Table 2. Lifetimes using different amounts of initial energy for the sensors.

Energy (J/node)	Protocol	Round first node dies	Round last node dies
0.25	Direct	55	117
	MTE	5	221
	Static Clustering	41	67
	LEACH	394	665
0.5	Direct	109	234
	MTE	8	429
	Static Clustering	80	110
	LEACH	932	1312
1	Direct	217	468
	MTE	15	843
	Static Clustering	106	240
	LEACH	1848	2608

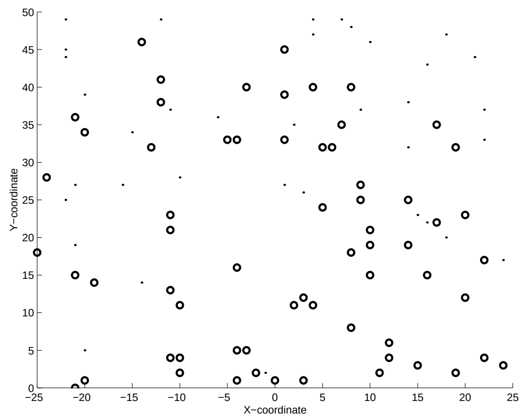

Figure 12. Sensors that remain alive (circles) and those that are dead (dots) after 1200 rounds with 0.5 J/node for LEACH. Note that this shows the network 1020 rounds further along than Figure 6.

dom" fashion. If Figure 12 is compared with Figure 6, we see that the order in which nodes die using LEACH is much more desirable than the order they die using direct communication or MTE routing. With random death, there is no one section of the environment that is not being "sensed" as nodes die, as occurs in the other protocols.

5. LEACH Algorithm Details

The operation of LEACH is broken up into *rounds*, where each round begins with a set-up phase, when the clusters are organized, followed by a steady-state phase, when data transfers to the base station occur. In order to minimize overhead, the steady-state phase is long compared to the set-up phase.

5.1 Advertisement Phase

Initially, when clusters are being created, each node decides whether or not to become a cluster-head for the current round. This decision is based on the suggested percentage of cluster heads for the network (determined a priori) and the number of times the node has been a cluster-head so far. This decision is made by the node n choosing a random number between 0 and 1. If the number is less than a threshold $T(n)$, the node becomes a cluster-head for the current round. The threshold is set as:

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

where P = the desired percentage of cluster heads (e.g., $P = 0.05$), r = the current round, and G is the set of nodes that have not been cluster-heads in the last $\frac{1}{P}$ rounds. Using this threshold, each node will be a cluster-head at some point within $\frac{1}{P}$ rounds. During round 0 ($r = 0$), each node has a probability P of becoming a cluster-head. The nodes that are cluster-heads in round 0 cannot be cluster-heads for the next $\frac{1}{P}$ rounds. Thus the probability that the remaining nodes are cluster-heads must be increased, since there are fewer nodes that are eligible to become cluster-heads. After $\frac{1}{P} - 1$ rounds, $T = 1$ for any nodes that have not yet been cluster-heads, and after $\frac{1}{P}$ rounds, all nodes are once again eligible to become cluster-heads. Future versions of this work will include an energy-based threshold to account for non-uniform energy nodes. In this case, we are assuming that all nodes begin with the same amount of energy and being a cluster-head removes approximately the same amount of energy for each node.

Each node that has elected itself a cluster-head for the current round broadcasts an advertisement message to the rest of the nodes. For this "cluster-head-advertisement" phase, the cluster-heads use a CSMA MAC protocol, and all cluster-heads transmit their advertisement using the same transmit energy. The non-cluster-head nodes must keep their receivers on during this phase of set-up to hear the advertisements of all the cluster-head nodes. After this phase is complete, each non-cluster-head node decides the cluster to which it will belong for this round. This decision is based on the received signal strength of the advertisement. Assuming symmetric propagation channels, the cluster-head advertisement heard with the largest signal strength is the cluster-head to whom the minimum amount of transmitted energy is needed for communication. In the case of ties, a random cluster-head is chosen.

5.2 Cluster Set-Up Phase

After each node has decided to which cluster it belongs, it must inform the cluster-head node that it will be a member of the cluster. Each node transmits this information back to the cluster-head again using a CSMA MAC protocol. During this phase, all cluster-head nodes must keep their receivers on.

5.3 Schedule Creation

The cluster-head node receives all the messages for nodes that would like to be included in the cluster. Based on the number of nodes in the cluster, the cluster-head node creates a TDMA schedule telling each node when it can transmit. This schedule is broadcast back to the nodes in the cluster.

5.4 Data Transmission

Once the clusters are created and the TDMA schedule is fixed, data transmission can begin. Assuming nodes always have data to send, they send it during their allocated transmission time to the cluster head. This transmission uses a minimal amount of energy (chosen based on the received strength of the cluster-head advertisement). The radio of each non-cluster-head node can be turned off until the node's allocated transmission time, thus minimizing energy dissipation in these nodes. The cluster-head node must keep its receiver on to receive all the data from the nodes in the cluster. When all the data has been received, the cluster head node performs signal processing functions to compress the data into a single signal. For example, if the data are audio or seismic signals, the cluster-head node can beamform the individual signals to generate a composite signal. This composite signal is sent to the base station. Since the base station is far away, this is a high-energy transmission.

This is the steady-state operation of LEACH networks. After a certain time, which is determined a priori, the next round begins with each node determining if it should be a cluster-head for this round and advertising this information, as described in Section 5.1.

5.5. Multiple Clusters

The preceding discussion describes how the individual clusters communicate among nodes in that cluster. However, radio is inherently a broadcast medium. As such, transmission in one cluster will affect (and hence degrade) communication in a nearby cluster. For example, Figure 13 shows the range of communication for a radio. Node A's transmission, while intended for Node B, corrupts any transmission to Node C. To reduce this type of interference, each cluster communicates using different CDMA codes. Thus, when a node decides to become a cluster-head, it chooses randomly from a list of spreading codes. It informs all the nodes in the cluster to transmit using this spreading code. The cluster-head then filters all received energy using the given spreading code. Thus neighboring clusters' radio signals will be filtered out and not corrupt the transmission of nodes in the cluster.

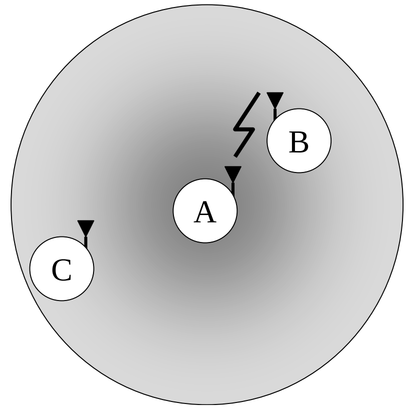

Figure 13. Radio interference. Node A's transmission to node B corrupts any transmission to node C.

Efficient channel assignment is a difficult problem[3], even when there is a central control center that can perform the necessary algorithms. Using CDMA codes, while not necessarily the most bandwidth efficient solution, does solves the problem of multiple-access in a distributed manner.

5.6. Hierarchical Clustering

The version of LEACH described in this paper can be extended to form hierarchical clusters. In this scenario, the cluster-head nodes would communicate with "super-cluster-head" nodes and so on until the top layer of the hierarchy, at which point the data would be sent to the base station. For larger networks, this hierarchy could save a tremendous amount of energy. In future studies, we will explore the details of implementing this protocol without using any support from the base station, and determine, via simulation, exactly how much energy can be saved.

[3]In fact, the authors in [13] assert that it is NP-complete.

6. Conclusions

In this paper, we described LEACH, a clustering-based routing protocol that minimizes global energy usage by distributing the load to all the nodes at different points in time. LEACH outperforms static clustering algorithms by requiring nodes to volunteer to be high-energy cluster-heads and adapting the corresponding clusters based on the nodes that choose to be cluster-heads at a given time. At different times, each node has the burden of acquiring data from the nodes in the cluster, fusing the data to obtain an aggregate signal, and transmitting this aggregate signal to the base station. LEACH is completely distributed, requiring no control information from the base station, and the nodes do not require knowledge of the global network in order for LEACH to operate.

Distributing the energy among the nodes in the network is effective in reducing energy dissipation from a global perspective and enhancing system lifetime. Specifically, our simulations show that:

- LEACH reduces communication energy by as much as 8x compared with direct transmission and minimum-transmission-energy routing.
- The first node death in LEACH occurs over 8 times later than the first node death in direct transmission, minimum-transmission-energy routing, and a static clustering protocol, and the last node death in LEACH occurs over 3 times later than the last node death in the other protocols.

In order to verify our assumptions about LEACH, we are currently extending the network simulator **ns** [11] to simulate LEACH, direct communication, and minimum-transmission-energy routing. This will verify our assumptions and give us a more accurate picture of the advantages and disadvantages of the different protocols. Based on our MATLAB simulations described above, we are confident that LEACH will outperform conventional communication protocols, in terms of energy dissipation, ease of configuration, and system lifetime/quality of the network. Providing such a low-energy, ad hoc, distributed protocol will help pave the way for future microsensor networks.

Acknowledgments

The authors would like to thank the anonymous reviewers for the helpful comments and suggestions. W. Heinzelman is supported by a Kodak Fellowship. This work was funded in part by DARPA.

References

[1] Bluetooth Project. http://www.bluetooth.com, 1999.

[2] Chandrakasan, Amirtharajah, Cho, Goodman, Konduri, Kulik, Rabiner, and Wang. Design Considerations for Distributed Microsensor Systems. In *IEEE 1999 Custom Integrated Circuits Conference (CICC)*, pages 279–286, May 1999.

[3] Clare, Pottie, and Agre. Self-Organizing Distributed Sensor Networks. In *SPIE Conference on Unattended Ground Sensor Technologies and Applications*, pages 229–237, Apr. 1999.

[4] M. Dong, K. Yung, and W. Kaiser. Low Power Signal Processing Architectures for Network Microsensors. In *Proceedings 1997 International Symposium on Low Power Electronics and Design*, pages 173–177, Aug. 1997.

[5] D. Dudgeon and R. Mersereau. *Multidimensional Digital Signal Processing*, chapter 6. Prentice-Hall, Inc., 1984.

[6] M. Ettus. System Capacity, Latency, and Power Consumption in Multihop-routed SS-CDMA Wireless Networks. In *Radio and Wireless Conference (RAWCON '98)*, pages 55–58, Aug. 1998.

[7] D. Hall. *Mathematical Techniques in Multisensor Data Fusion*. Artech House, Boston, MA, 1992.

[8] L. Klein. *Sensor and Data Fusion Concepts and Applications*. SPIE Optical Engr Press, WA, 1993.

[9] X. Lin and I. Stojmenovic. Power-Aware Routing in Ad Hoc Wireless Networks. In *SITE, University of Ottawa, TR-98-11*, Dec. 1998.

[10] T. Meng and R. Volkan. Distributed Network Protocols for Wireless Communication. In *Proc. IEEEE ISCAS*, May 1998.

[11] UCB/LBNL/VINT Network Simulator - ns (Version 2). http://www-mash.cs.berkeley.edu/ns/, 1998.

[12] R. Ruppe, S. Griswald, P. Walsh, and R. Martin. Near Term Digital Radio (NTDR) System. In *Proceedings MILCOM '97*, pages 1282–1287, Nov. 1997.

[13] K. Scott and N. Bambos. Routing and Channel Assignment for Low Power Transmission in PCS. In *5th IEEE Int. Conf. on Universal Personal Communications*, volume 2, pages 498–502, Sept. 1996.

[14] T. Shepard. A Channel Access Scheme for Large Dense Packet Radio Networks. In *Proc. ACM SIGCOMM*, pages 219–230, Aug. 1996.

[15] S. Singh, M. Woo, and C. Raghavendra. Power-Aware Routing in Mobile Ad Hoc Networks. In *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '98)*, Oct. 1998.

[16] L. Williams and L. Emergy. Near Term Digital Radio- a First Look. In *Proceedings of the 1996 Tactical Communications Conference*, pages 423–425, Apr. 1996.

[17] K. Yao, R. Hudson, C. Reed, D. Chen, and F. Lorenzelli. Blind Beamforming on a Randomly Distributed Sensor Array System. *Proceedings of SiPS*, Oct. 1998.