Optimizing Sensor Networks in the Energy-Latency-Density Design Space

Curt Schurgers, *Student Member*, *IEEE*, Vlasios Tsiatsis, *Student Member*, *IEEE*, Saurabh Ganeriwal, and Mani Srivastava, *Senior Member*, *IEEE*

Abstract—In wireless sensor networks, energy efficiency is crucial to achieving satisfactory network lifetime. To reduce the energy consumption significantly, a node should turn off its radio most of the time, except when it has to participate in data forwarding. We propose a new technique, called Sparse Topology and Energy Management (STEM), which efficiently wakes up nodes from a deep sleep state without the need for an ultra low-power radio. The designer can trade the energy efficiency of this sleep state for the latency associated with waking up the node. In addition, we integrate STEM with approaches that also leverage excess network density. We show that our hybrid wakeup scheme results in energy savings of over two orders of magnitude compared to sensor networks without topology management. Furthermore, the network designer is offered full flexibility in exploiting the energy-latency-density design space by selecting the appropriate parameter settings of our protocol.

Index Terms—Sensor networks, energy efficiency, wakeup, topology.

1 Introduction

Wireless sensor networks are autonomous ad hoc networks designed for monitoring tasks, such as battlefield surveillance, equipment supervision, intruder detection, and wildlife observation, among many others [1], [2], [3]. Sensor networks are made up of a large number of tiny devices, called sensor nodes, which contain integrated sensors, processors, and radios. The nodes gather various sensor readings, process them, coordinate among each other, and forward the processed information to a data sink. This forwarding typically occurs wirelessly via other nodes using a multihop path [3], [9]. The crucial design challenge in sensor networks is energy efficiency as the individual nodes have only a small battery as a power source. To achieve satisfactory network lifetime, energy efficiency is really a problem that needs to be tackled on the level of the entire network. One of the key aspects is the organization of network communications as the radio is the main energy consumer in a sensor node [1], [3], [4]. The only way to reduce this energy is to completely turn the radio off [4]. However, besides sensing their environment, nodes also form the ad hoc network needed to forward the data to the data sink. Topology management schemes coordinate which nodes turn their radio off and when, such the traffic forwarding remains satisfactory while minimizing the network energy consumption.

Consider, for example, a sensor network that is designed to detect brush fires. It has to remain operational for months or years while only sensing if a fire has started. Once a fire is detected, this information should be forwarded to the user quickly. Even when we want to track how the fire spreads, it probably suffices for the network to remain up only for an additional week or so. Similar observations hold for applications such as surveillance of battlefields, machine failures, room occupancy, or other reactive scenarios, where the user needs to be informed once a condition is satisfied. The majority of the time, the network is only sensing its environment, which we refer to as the network being in the *monitoring state*. Once an event happens, data needs to be forwarded to the data sink and the network transitions to the more active *transfer state*.

We propose a new topology management scheme, called ***STEM (Sparse Topology and Energy Management)***. It reduces the energy consumption in the monitoring state to a bare minimum while ensuring satisfactory latency for transitioning to the transfer state. In fact, STEM allows us to efficiently trade one design constraint (energy) for the other (latency). We also combine it with techniques that leverage increased network density to obtain energy savings. In essence, our scheme thus offers the designers full flexibility in trading latency, density, and energy versus each other.

Furthermore, we derive the mathematical model that governs these tradeoffs. For example, for a specific desired network lifetime and acceptable notification latency, we can calculate the network density that is required. When STEM is running in the network, this density will assure that both the latency and the network lifetime are as desired. This model is therefore a tool for the network designer to choose the optimal parameter settings given the deployment requirements. At design time, it allows the selection of the desired operating point in the latency-density-lifetime design space.

2 Related Work

For routing in sensor networks, two alternative approaches have been considered: flat multihop and clustering. Although

• *The authors are with the Networked and Embedded Systems Lab (NESL), Electrical Engineering Department, University of Calfornia at Los Angeles, 56-125B Eng. IV, UCLA-EE Dept., Los Angeles, CA 90095. E-mail: {curts, tsiatsis, saurabh, mbs}@ee.ucla.edu.*

Manuscript received 7 Mar. 2002; revised 13 May 2002; accepted 13 May 2002.
For information on obtaining reprints of this article, please send e-mail to: tmc@computer.org, and reference IEEECS Log Number 7-032002.

1536-1233/02/$17.00 © 2002 IEEE

TABLE 1
Radio Power Characterization

Radio mode	Power consumption (mW)
Transmit (T_x)	14.88
Receive (R_x)	12.50
Idle	12.36
Off	0.016

STEM is applicable to both of them, we mainly focus on flat multihop routing [3], [7], [8]. For clustered approaches [9], which are possibly hierarchical, our scheme can be used to reduce the energy of the cluster heads. Recently, topology management techniques, called SPAN [5] and GAF [6], have been proposed for flat multihop routing. They trade network density for energy savings while preserving the data forwarding capacity of the network. However, the absence of traffic in the monitoring state is not exploited at all. By integrating these schemes into STEM, as we describe in Section 6, we can combine their benefits with those of STEM to achieve compounded energy savings.

STEM is essentially a technique to quickly transition to the transfer state, while making the monitoring state as energy efficient as possible. Other authors have proposed to do this wake up using a separate paging channel [10]. This approach critically assumes that the listen mode of this paging radio is ultra low power. However, the difference in the transmission range between the data and wakeup radio presents a major difficulty. STEM offers an alternative by trading energy for latency. Furthermore, if such a low-power radio is available, the energy savings are further improved by using it in a low duty cycle, as STEM does. The work in [14] describes an algorithm that also uses a low duty cycle radio. This algorithm is designed for a different goal, namely, to discover the network topology some time after its deployment. It is less aggressive than STEM and, therefore, would result in much higher latencies to transition to the transfer state. The same principle of duty cycling the radio is also adapted in the Medium Access Control (MAC) protocol presented in [16], called S-MAC. However, channel access and node wakeup are integrated together. In the monitoring state, where there is no data to forward, STEM is therefore more energy efficient than S-MAC while assuring timely transitioning to the transfer state. In this transfer state, our approach allows for any MAC protocol, including S-MAC.

3 Sparse Topology and Energy Management

3.1 Basic Concept

In the monitoring state, where there is no traffic to forward, only the node's sensors and some preprocessing circuitry are on. For simplicity, we refer to this state as the node being asleep or in the sleep state. When a possible event is detected, the main processor is awakened to analyze the data in more detail. The radio is only turned on if the processor decides that the information needs to be communicated to other nodes. The reason for this can be understood from the radio mode power numbers in Table 1. These are for the TR1000 radio from RF Monolithics [15],

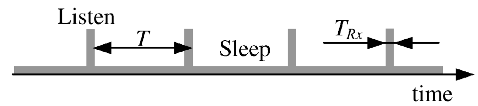


Fig. 1. Low-power listen mode.

where the transmit range is set to approximately 20 meters [4]. This low-power radio has a data rate of 2.4 Kbps and uses On-Off Keying (OOK [18]) modulation. As can be observed from this table, the radio consumes considerable power except when completely turned off.

To forward traffic, nodes on the multihop path need to be awakened, or, equivalently, transition from the monitoring to the transfer state. The problem is that these nodes have no way of knowing when to transition if they did not detect that same event. Thus, the dilemma we are faced with is the following: For energy reasons, the nodes should turn off their radio when in the monitoring state, but still need to be told somehow if they should turn it back on. As a solution, each node periodically turns on its radio for a short time to listen if someone wants to communicate with it. In the monitoring state, instead of complete being asleep, a node goes into this low-power listen mode, as shown in Fig. 1. The period of the listen-sleep cycle is denoted as T. The node that wants to communicate, the *initiator node*, polls the node it is trying to wake up, called the *target node*. We will detail the nature of these polls in Section 3.3. As soon as the target node, which is in the low-power listen mode of Fig. 1, hears the poll, the link between the two nodes is activated. If the packet needs to be relayed further, the target node will become an initiator for the next hop and the process is repeated.

Once the link between nodes is activated, data is transferred using a MAC protocol. This MAC protocol is only used in the transfer state as even the most efficient one would consume a lot more energy than our low-power listen mode. The reason is that MAC protocols are designed to organize access to the shared medium, in addition to contacting nodes. Our strategy is thus to decouple the transfer and wakeup functionalities: In the monitoring state, we consume as little energy as possible and only in the transfer state is the MAC protocol started.

3.2 Alternative Setups

Without special protocol provisions, nodes are not synchronized and, therefore, do not know the phase of each other's wakeup-sleep cycles in the listen mode. To avoid missing the short time the target node has its radio on, the initiator has to poll continuously. As the data arrivals are uncorrelated with the sleep cycles, it will take about half a cycle for the target to hear the poll. However, this aggressive polling causes problems, as shown in Fig. 2. A regular data transmission is going on between nodes A and

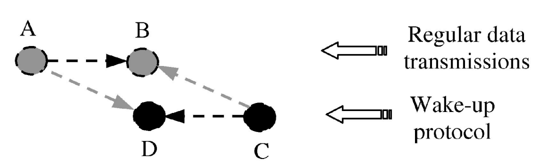


Fig. 2. Interference problem of aggressive wakeup.

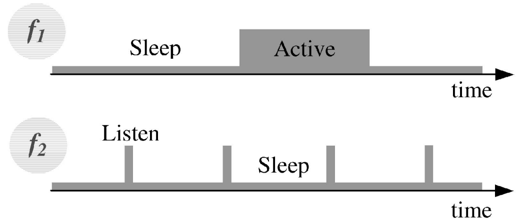


Fig. 3. Separate data and wakeup using two radios.

B. When node C wants to wake up D, its aggressive polls will collide with the ongoing data transmission, essentially acting as a jammer to B. Despite possible recovery action from the MAC, the data communication between A and B suffers from extra delays. As we will illustrate in Section 4, more energy can be saved if we allow more time to set up a link between two nodes. Therefore, we might desire this setup procedure to be relatively long, but its impact is only felt at the start of a communication epoch. However, it is typically undesirable as this setup would also cause equally long disruptions of ongoing transmissions.

Since this aggressive nature is needed to limit the wakeup latency, the solution is to completely separate data transfer from wakeup. A natural choice is to use two radios operating in separate frequency bands. As shown in Fig. 3, the radio in band f_1 is only turned on in the transfer state, and the wakeup band f_2 can be viewed as a separate paging channel. Unlike [10], we are not limited by the availability of an ultra low power radio for this paging channel. Instead, we can use the most efficient radio available and further reduce the energy consumption by putting it in our low-power listen mode. This allows us to trade energy savings versus latency, beyond the capabilities of the radio alone, as we will detail later on.

In principle, we could use one radio and let it switch between frequencies. However, if a target node is already transferring data and also has to wake up another node, it has to interrupt its data transmission or postpone the wakeup. Both are undesirable and, therefore, we choose to use two radios. The penalty of making the node more expensive is minimal, as the radio typically accounts for less than 15 percent of the cost of a sensor node (e.g., an extra TR1000 [15] radio on the MICA motes [17]).

An alternative way to separate data transmissions from wakeup is to assign them to separate time slots, as shown in Fig. 4. In this case, the initiator only sends the poll in the slot to which the target is listening. Unfortunately, this option requires time synchronization as all nodes have to remain synchronized at all times, which brings about considerable overhead. The monitoring state would thus be much more energy hungry than in the two radio option of Fig. 3. As we will explain in Section 6, the same energy savings as in the setup of Fig. 3 could be achieved by deploying more nodes, but the total network deployment cost would exceed that of using slightly more expensive nodes. We therefore opt for the solution with two radios. Sensor nodes developed by Sensoria Corporation [11], for example, are already equipped with a dual radio.

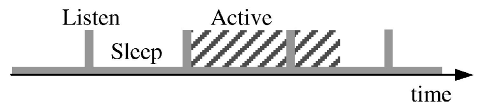


Fig. 4. Separate data and wakeup in time.

3.3 Operation of STEM-B and STEM-T

To poll the target node, the initiator sends a stream of beacon packets in band f_2. Each beacon contains the MAC address of both the target and initiator node. As soon as the target receives a beacon, it turns on its data radio in band f_1 and also sends back an acknowledgment in band f_2. This way, the initiator knows when it can stop polling, resulting in a reduction in setup latency, as we will detail in Section 4. The length of the interbeacon interval T_B is such that there is sufficient time to send the beacon and receive the acknowledgment. The time during which the radio is turned on in the listen mode is denoted as T_{Rx} (see Fig. 1). In order to guarantee that the target node receives at least one beacon, T_{Rx} needs to be at least as long as the transmit time of a beacon plus the interbeacon interval T_B.

It is still possible that collisions between beacons occur. To handle this issue, a node also turns on its data radio if it hears a collision during the listen interval T_{Rx}. A collision can be detected by monitoring the RSSI (received signal strength indicator) of the radio. In this case, the node does not send back an acknowledgment as it would likely collide with that of other nodes that are also awakened this way. After transmitting the beacon stream for a sufficient amount of time (approximately equal to T; we derive the exact expression in Section 4), the initiator node can be sure that the target node has turned on its data radio in band f_1. Indeed, if the target node has surely listened once, it has received the beacon correctly or seen a collided packet and turned on its data radio in either case. It is possible that nodes that were not the intended target node decided to wake up due to beacon collisions. If they do not receive any traffic in band f_1 after some time, they time out and return to the monitoring state. Eventually, only the desired target nodes keep their data radio on for the duration of the data transfer. The regular MAC layer handles collisions on the data plane.

As an alternative to the beacon-based approach described above, the initiator node could simply send a wakeup tone. In this case, nodes wake up when they detect the presence of signal energy in their listen interval. This is exactly the same situation as when beacons collide. However, with the tone-based approach, a target node never sends back an acknowledgment. As before, the initiator has to send the tone for a sufficiently long time, such that the target surely has awoken once. Note that all nodes in the neighborhood of the initiator always wake up in this case. In this paper, we analyze both the beacon-based and the tone-based scheme. We refer to them as ***STEM-B*** (beacon) and ***STEM-T*** (tone), respectively.

4 Theoretical Analysis of STEM

4.1 Setup Latency

Before simulating our protocol, we first develop a theoretical model of the system performance. We define the *setup latency* T_S *of a link* as the interval from the time the initiator starts contacting the target to the time both nodes have turned on their data radio.

In STEM-B, the average setup latency is approximately given by (1) for the case where there is no beacon collision. In this equation, B_1 and B_2 are the transmit duration of the beacon and acknowledgment packet, respectively. To conserve the flow of this paper, we have moved the derivation of (1) to Appendix A.

$$\bar{T_S} = \frac{T + T_B}{2} + 2 \cdot B_1 + B_2 - T_{Rx}. \tag{1}$$

In Appendix A, we also show that the setup latency in case of a beacon collision is given by (2). This expression, at the same time, specifies the maximum time during which the initiator needs to send beacons and is, in essence, the worst-case latency of STEM-B.

$$T_S = T + T_B + 2 \cdot B_1 + B_2 - T_{Rx}. \tag{2}$$

The setup latency of the tone-based variant, STEM-T, is constant and given by (3), as we derive in Appendix A. In this equation, T_I is the time interval over which channel sensing needs to be performed to achieve a satisfactory low false alarm probability.

$$T_S = T - T_{Rx} + 2 \cdot T_I. \tag{3}$$

4.2 Energy Savings

For typical short-range radios used in sensor networks, the transmit, receive, and idle power are almost identical (see also Table 1). For the data radio, we approximate all of them by just one value: P_1. For the wakeup radio, which is not necessarily of the same type as the data radio, this value is P_2. In our analysis, P_1 and P_2 are chosen equal to the idle power. The sleep power for the data and wakeup radio are denoted as $P_{sleep,1}$ and $P_{sleep,2}$, respectively. To simplify our subsequent expressions, we define two new parameters:

$$\rho = \frac{P_2}{P_1}, \tag{4}$$

$$\phi = \frac{P_{sleep,1} + P_{sleep,2}}{P_1}. \tag{5}$$

Parameter ρ represents the relative power of the data and wakeup radio. We have introduced parameter ϕ for readability reasons, but it also corresponds to the lower bound on the power savings, as we will explain in Section 6.3.

After being awakened using STEM, a node turns on its data radio in band f_1. When the data communication phase is over, the node returns to the low-power sleep state. The average time the radio is on during one such data communication phase is denoted as t_{burst}. If the MAC protocol is such that the node turns off its radio for some time, this sleep time is not included in t_{burst}. Each communication phase requires one transition from the monitoring to the transfer state. The number of such transitions per second is called the wakeup frequency, f_W. The fraction of time the data radio is turned on is thus given by α, defined as (6). This also corresponds to the relative importance of the transfer state. The inverse of the duty cycle of the wakeup radio is called β.

$$\alpha = f_W \cdot t_{burst}, \tag{6}$$

$$\beta = \frac{T}{T_{Rx}}. \tag{7}$$

We evaluate the energy savings of running STEM relative to the situation where there is no wakeup radio and the data radio is never turned off. The relative energy for both STEM-B and STEM-T is approximately given by (8). In this equation, f_S is the average number of times per second the node sets up a link as the initiator or, equivalently, the setup frequency. The derivation of this expression is included in Appendix B.

$$\frac{E}{E_0} = \frac{\rho}{\beta} + \alpha + f_S \cdot T_S \cdot \rho + \phi. \tag{8}$$

Although (8) is valid for both versions, α is likely to be larger in STEM-T than STEM-B since more nodes are awakened when they are not the intended target. In any case, the first two terms are typically dominant. The energy savings are larger when β increases, by extending the period *T*. This results in larger setup latencies, as can be seen from (1)-(3). The energy savings are also larger when the monitoring state becomes more dominant and when fewer setups are needed. If the wakeup radio can be designed to be lower power than the data radio ($\rho < 1$), the savings also increase. The last term in (8) presents a floor to the energy as the best we can do is to have the two radios sleeping all the time. Since the node has a finite battery capacity, the energy savings directly correspond to the same relative increase in the node's lifetime, which ultimately results in a prolonged lifetime of the sensor network.

5 STEM Performance Evaluation

5.1 Simulation Setup

In this section, we verify our algorithm and theoretical analysis through simulations which were written on the Parsec platform, an event-driven parallel simulation language [12]. We distribute *N* nodes in a uniformly random fashion over a field of size *L* x *L*. Each node has a transmission range *R*. For a uniformly random deployment, the network connectivity is only a function of the average number of neighbors of a node, denoted by parameter λ:

$$\lambda = \frac{N}{L^2} \cdot \pi R^2. \tag{9}$$

Since traffic communication patterns depend solely on the network connectivity, we only have to consider λ and not *N*, *R*, and *L* separately. This statement was verified through simulations and we therefore can characterize a uniform network density by the single parameter λ.

In principle, data and wakeup radios can be different. This is especially true for STEM-T, where the wakeup radio only needs to be able to send out a tone and detect it, possibly resulting in a simplified implementation. In our simulations,

TABLE 2
Simulation Settings

General settings	
R	20 m
L	79.27 m
R_b	2.4 Kbps

STEM-B	
T_{Rx}	225 ms
T_B	150 ms
L_{beacon}	144 bits
L_{ack}	144 bits

STEM-T	
T_{Rx}	10 ms
T_I	9.5 ms

we have chosen the same TR1000 radio of Table 1 for both data and wakeup such that $\rho = 1$. Table 2 lists the other simulation settings. The area of the sensor network is such that, for N = 100, we have λ = 20. Furthermore, our setup includes a CSMA-type MAC, similar to the DCF (distributed coordination function) of 802.11. The node closest to the top left corner detects an event and sends 20 information packets of 1,040 bits (including all headers) to the data sink with an interpacket spacing of 16 seconds. The node turns its data radio back off if it has not received any traffic for 20 seconds. The time for the data transfer, t_{burst}, is thus about 340 seconds. The data sink is the sensor node located closest to the bottom right corner of the field. We have observed that the average path length is between six and seven hops. All reported results are averaged over 100 simulation runs.

Clearly, the nodes that are on the path consume more energy than the ones that are not. For those on the path, f_W is equal to the inverse of the total simulation time since there is only one communication phase. The value of α is thus the same for STEM-B and STEM-T. In addition, f_S is equal to f_W, except for the final destination, where it is zero. Besides the nodes on the path, there are those that are awakened accidentally in STEM-T or due to collisions in STEM-B. They have the same value of f_W as above, but f_S is equal to zero. Therefore, they consume less energy. Finally, all other nodes always remain in the low-power listen state and have both f_W and f_S equal to zero. In our subsequent simulations, we only report the average energy for the nodes that are on the path. This is essentially the worst case as all other nodes will consume less energy and are therefore less critical in the considered scenario. If we had averaged over all nodes, the energy savings would depend on the size of the overall network. The effect of waking up nodes that are not on the path is addressed in Section 6.

5.2 Simulation Results

Fig. 5 shows the average setup latency per hop as a function of the wakeup period T. The simulation results of STEM-B without collisions agree well with the theoretical analysis of (1). It also confirms that the approximations used in Appendix A are indeed appropriate for the chosen settings. In addition, we have verified that, if the maximum beacon train duration is equal to (2), at least one beacon is received. The worst-case latency for STEM-B in the case of collisions is thus given by (2), which is also plotted in Fig. 5. For STEM-T, the setup latency is equal to the chosen tone duration, given by (3). We have verified that the short listen time is indeed never missed.

For the same average setup latency, the period T of STEM-B can be approximately twice as long as that of STEM-T, in case there are no collisions. The reason is the feedback provided by the acknowledgments in STEM-B. Note that we have used a relatively slow radio with a bitrate of just 2.4 Kbps. By choosing a radio that is 10 times faster, the absolute latency also decreases by this factor.

Figs. 6 and 7 show the relative energy of STEM-B and STEM-T as a function of the period T for different values of α. As defined in the previous section, α represents the fraction of time in the transfer state. The solid theoretical curves are obtained from (8) and we observe again the close correspondence to simulated values. As α decreases, the monitoring state becomes more predominant. STEM already results in energy savings when the network is in the monitoring state half of the time. For STEM-T, the two top curves flatten out when the energy reaches α. This is indeed the best gain possible in (8) for a certain amount of data traffic.

When comparing Figs. 6 and 7, we see that STEM-T results in substantially more energy savings than STEM-B. The reason is the following: As discussed in Section 3.3, the values of T_{Rx} have to satisfy (10) and (11) for STEM-B and STEM-T, respectively. Since T_I is typically less than both B_1 and B_2, T_{Rx} is at least three times shorter for STEM-T than STEM-B. Typically, as in our settings, T_{Rx} is much shorter for STEM-T. From (7), we therefore notice that β is larger for STEM-T, which makes it superior in terms of energy savings.

$$\textbf{STEM-B} \qquad T_{Rx} \geq T_B + B_1 \geq 2 \cdot B_1 + B_2, \tag{10}$$

$$\textbf{STEM-T} \qquad T_{Rx} \geq T_I. \tag{11}$$

For the same setup latency, STEM-T needs to have a period T that is about twice as large as that of STEM-B, but

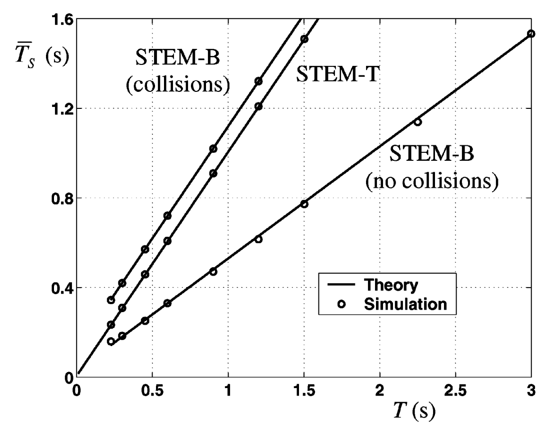


Fig. 5. Average setup latency per hop.

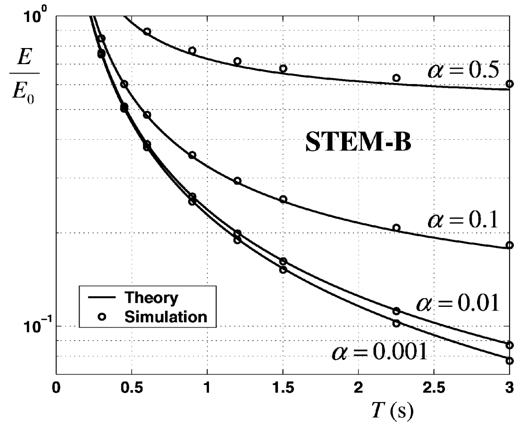


Fig. 6. Energy versus period for STEM-B.

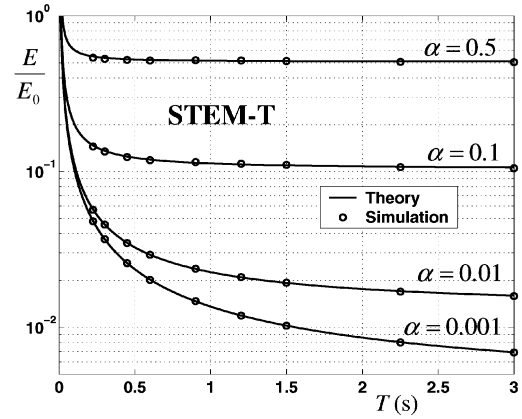


Fig. 7. Energy versus period for STEM-T.

β is still larger, as argued above. This makes STEM-T preferable in trading energy versus setup latency. However, the disadvantage is that not only the intended node wakes up. This was not an issue in the particular scenario considered here, but can be significant in general. This aspect may eventually negate the edge STEM-T has over STEM-B and make it less efficient. In the next section, where we combine STEM with density-based topology management schemes, this effect is indeed observed to be important.

6 STEM AND NETWORK DENSITY

As mentioned in the introduction, existing topology management schemes, such as GAF and SPAN, coordinate the radio sleep and wakeup cycles while ensuring adequate communication capacity. The resulting energy savings increase with the network density. STEM, on the other hand, leverages the setup latency. Moreover, it can be integrated with schemes, such as GAF or SPAN, to achieve additional gains by also exploiting the density dimension in topology management. We specifically focus on combining STEM with GAF.

6.1 Behavior of GAF

In this section, we discuss plain GAF, i.e., without STEM. Furthermore, we also analyze its behavior theoretically as this is an essential building block in the analysis of STEM combined with GAF. Such an analysis was not provided in the original paper [6]. The GAF algorithm is based on a division of the sensor network in a number of virtual grids of size r by r. The value of r is chosen such that all nodes in a grid are equivalent from a routing perspective. In [6], it was derived that r has to satisfy:

$$r \leq \frac{R}{\sqrt{5}}. \tag{12}$$

As before, R denotes the radio transmission range. The average number of nodes in a grid, M, is given by (13). By combining this with (12), we see that M is related to the network density λ by satisfing (14). In the remainder of this paper, we choose (12) and (14) to hold with equality.

$$M = \frac{N}{L^2} \cdot r^2, \tag{13}$$

$$M \leq \frac{\lambda}{5\pi}. \tag{14}$$

Since all nodes in a grid are equivalent from a routing perspective, we can use this redundancy to increase the network lifetime. GAF only keeps one node awake in each grid, while the other nodes turn their radio off. To balance out the energy consumption, the burden of traffic forwarding is rotated between nodes. In the theoretical analysis, we ignore the unavoidable time overlap of this process associated with handoff. If there are m nodes in a grid, the node will (ideally) only turn its radio on $1/m$th of the time and, therefore, last m times longer. The relative energy compared to a scenario without GAF for a node in a grid with m nodes is therefore given by:

$$\left.\frac{E}{E_0}\right|_{node} = \frac{1}{m}. \tag{15}$$

We can essentially view a grid as being a "virtual node," composed of m actual nodes. For the special case of a uniformly random node deployment, the probability of having m nodes in a grid is:[1]

$$Q(m) = \frac{M^m}{m!} \cdot e^{-M}. \tag{16}$$

The derivation of this equation is similar to that of the degree of a node in [4]. However, some grids will not contain any nodes at all and the probability of having m nodes in a used grid is:

$$Q(m|m \geq 1) = \frac{Q(m)}{Q(m \geq 1)} = \frac{M^m}{m!} \cdot \frac{e^{-M}}{1 - e^{-M}}. \tag{17}$$

By combining (15) and (17), we derive that the average relative energy of a node when running GAF is expressed as (18).

$$\left.\frac{E}{E_0}\right|_{node} = \frac{1 - e^{-M}}{M}. \tag{18}$$

6.2 Analysis of STEM combined with GAF

As discussed in the previous subsection, GAF leverages the network density to conserve energy while leaving the data

1. We use the symbol Q in this paper for probabilities to avoid confusion with power (denoted by P).

forwarding capacity intact. STEM, on the other hand, saves energy by trading it for path setup latency. We anticipate better results by combining both approaches in an effort to exploit both latency and density dimensions.

In GAF, a grid can be viewed as having one virtual node and the physical nodes alternatively perform the functionality of that virtual node. From this perspective, STEM can be introduced in a straightforward manner by letting it run on the virtual node. In real life, nodes alternate between sleep and active states, as governed by GAF. The one active node in the grid runs STEM in the same way as described in Section 3. The routing protocol only needs to be modified to address virtual nodes (or grids) instead of real nodes.

However, we need to change the mechanism by which the functionality of being active in a grid is rotated between nodes, which is referred to as "leader election." In the original election scheme of GAF, described in [6], nodes that are asleep decide to become the leader after some time interval. To resolve the inconsistency of having multiple leaders, these nodes send periodic broadcasts and listen to similar messages from the other leaders in their grid. Upon receiving such broadcasts, each leader decides to go to sleep or remain a leader based on the expected remaining time to live of both nodes, which is included in the broadcasts. Note that this procedure requires the leader to have its radio on continuously.

If leaders run STEM, as we propose in our hybrid scheme, they have their data radio turned off and will not receive the broadcast messages. We therefore need another election scheme to avoid the persistent occurrence of multiple leaders in one grid. As a solution, a node that wants to become the leader first sets up a link to the current leader using regular STEM. It does not need to know the exact node to address as it can simply wake up "whoever is the current leader." Once the link is set up, the necessary information to decide the election process is exchanged on the data plane. If a node cannot contact the current leader, it assumes that it died (e.g., due to physical destruction) and takes over its role.

With this modification, STEM and GAF can be integrated effectively. As they are orthogonal in our hybrid scheme, we can directly obtain (19) for the relative energy gain of a node in a grid with m nodes. This is based on expanding (8), where the statistics of m are given by (17). The extra term Δ represents the overhead of the leader election process (which we ignored previously in our analysis of GAF). As it is based on STEM, we could model this election overhead using (8), at least in principle. However, quantifying the associated values of wakeup frequency is hard and we chose not to model the overhead in detail.

$$\left.\frac{E}{E_0}\right|_{node} = \frac{1}{m}\left[\frac{\rho}{\beta} + \alpha + f_S \cdot T_S \cdot \rho\right] + \phi + \Delta. \qquad (19)$$

From (19), the average relative energy over all nodes can be derived as being equal to (20), the same way as was done in Section 6.1.

$$\frac{E}{E_0} = \frac{1 - e^{-M}}{M}\left[\frac{\rho}{\beta} + \alpha + f_S \cdot T_S \cdot \rho\right] + \phi + \Delta. \qquad (20)$$

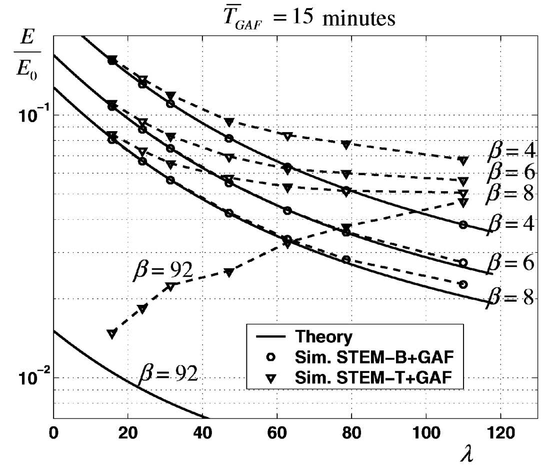


Fig. 8. Relative energy saving versus density for $T_{GAF} = 15$ minutes.

For the link setup latency of regular data traffic, the expressions are exactly the same as the ones for STEM, given in Section 4.1. The reason is that the leader appears simply as a virtual node that is using STEM as long as there is no interference from the leader election process. As this election process occurs at a timescale that is much larger than the link setup time, such interference is negligible.

6.3 Evaluation of STEM Combined with GAF

We now verify our hybrid scheme of STEM combined with GAF through simulations, again with the settings of Tables 1 and 2. To limit the dimensionality of the graphs, we have chosen $\alpha = f_S = 0$. This corresponds to a network that is always in the monitoring state, but we have verified that the algorithm and analysis also work fine when there is data traffic. All reported results are averaged over 1,000 simulation runs. In Fig. 8, the relative energy is plotted versus the network density λ for our hybrid scheme of STEM + GAF. We can also deduce the behavior of pure STEM without GAF from this figure. By comparing (8) and (20), we see that the behavior of STEM alone is mathematically equivalent to that of STEM+GAF with $M = \lambda = 0$ and $\Delta = 0$. Although, $\lambda = 0$ has no physical significance by itself, it allows us to visualize the behavior of STEM on the same graph as that of STEM + GAF.

A node tries to become the leader after a random time T_{GAF} in the range of 15 ± 3 minutes. For the theoretical values, we have set $\Delta = 0$ due to the complexity of modeling the leader-election overhead. This causes the discrepancies in Fig. 8 between the theoretical analysis and the simulated results, denoted by circles for STEM-B and by triangles for STEM-T. For the same value of the inverse duty cycle β, STEM-B outperforms STEM-T, although both are given by (19). The reason is that, in STEM-T, the leader wakes up each time one of its neighbors initiates the election process, even if it is not part of the same virtual grid. In STEM-B, the leader only wakes up as a response to other nodes in its grid. However, for the same β, the setup latency of STEM-T is smaller than that of STEM-B.

To compare both schemes for similar latency, we have included the curve for STEM-T with $\beta = 92$. This results in a setup latency of 0.93 seconds, which is the same as that of

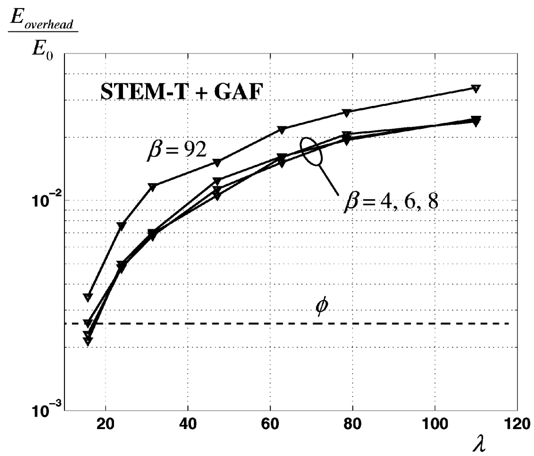


Fig. 9. Leader election overhead for STEM-T.

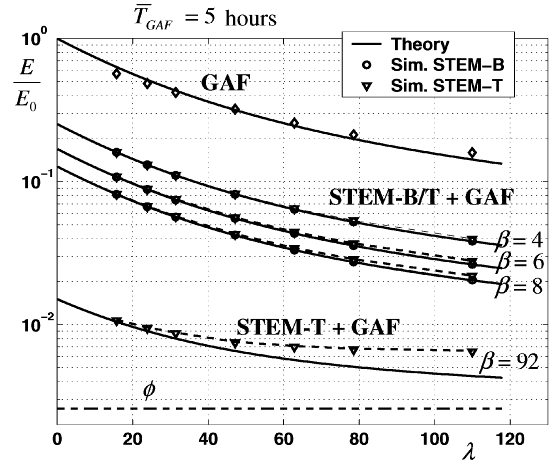


Fig. 10. Relative energy saving versus density for $T_{GAF} = 5$ hours.

STEM-B with β = 8. The unexpected behavior of the simulated curve with β = 92 is due to the GAF leader-election overhead. As we will illustrate shortly, this overhead does not vary much with β and increases with λ. As the energy savings due to STEM are large for large β, the overall energy consumption of STEM+GAF is heavily dominated by the leader-election overhead, in this case, as it becomes the major cause of energy consumption.

For the same setup latency, one of the two variants (STEM-T+GAF with β = 92 or STEM-B+GAF with β = 8) is more energy efficient, depending on the network density, with a crossover point at λ = 62. Note that, for these particular settings, it is undesirable to leverage the density using GAF. Instead, running STEM-T without GAF, thereby avoiding the leader election overhead, is most energy efficient.

Fig. 9 plots the overhead of the leader election for STEM-T. When the network density increases, leaders are awakened more frequently. In addition, the effect of the overhead becomes relatively more pronounced in Fig. 8 when the absolute energy decreases. For STEM-B, the overhead is considerably below the value of ϕ and is not visible in Fig. 9.

In the previous simulations, a node tries to become a leader relatively often (about every 15 minutes). In more realistic scenarios, the election process is likely to operate at a much larger timescale such that the overhead decreases. Indeed, a node is expected to live longer when its energy consumption is reduced (ignoring physical destruction). It can therefore remain the leader for a longer time period. Fig. 10 shows the relative energy savings of the different schemes for T_{GAF} in the range of 5 ± 1 hour. The effect of the overhead is heavily reduced in this case, except when the absolute energy is extremely low. From (19), we also see that the absolute best we can do is have all nodes sleeping all the time such that the relative energy is given by ϕ. Although not shown here, we also verified that the link setup latency is similar to that of STEM alone.

Fig. 10 also shows the results of GAF alone and, as explained before, the behavior of pure STEM corresponds to the point on the curves of STEM + GAF where λ = 0. By combining STEM and GAF as in our hybrid scheme, both network density and path setup latency are leveraged to achieve considerable energy savings. Even at low densities or low latencies, the other dimension can be traded off for energy savings. The gains are compounded when both dimensions can be exploited together. For a network density of λ = 80 and a setup latency per hop of 0.93 seconds (STEM-T with β = 92), the energy consumption of a node is 150 times lower than without topology management. In Fig. 10, we see that this is already close to the lower bound of ϕ.

Our hybrid scheme provides the sensor network designer with full flexibility to trade energy, latency, and density for each other. The equations we derived in the previous section allow him/her to predict, at design-time, the exact relationships between these three parameters when our hybrid scheme is deployed in the network. For example, given a bound on the maximum allowable latency and the desired energy consumption per node, we can calculate the required density and listen period of STEM. With these settings, the network will satisfy the aforementioned constraints when running our hybrid STEM+GAF topology management protocol.

7 Conclusions

In this paper, we have introduced STEM, a topology management technique that trades power savings for path setup latency in sensor networks. It emulates a paging channel by having a separate radio operating at a lower duty cycle. Upon receiving a wakeup message, it turns on the primary radio, which takes care of the regular data transmissions. This wakeup message can take the form of a beacon packet or simply a tone, resulting in two variants of STEM. Our topology management is specifically geared toward those scenarios where the network spends most of its time waiting for events to happen, without forwarding traffic.

Whether the beacon-based or the tone-based variant is superior depends on the application scenario. We observed this when combining STEM with a topology management scheme that leverages the network density. The resulting hybrid scheme exploits both setup latency and network

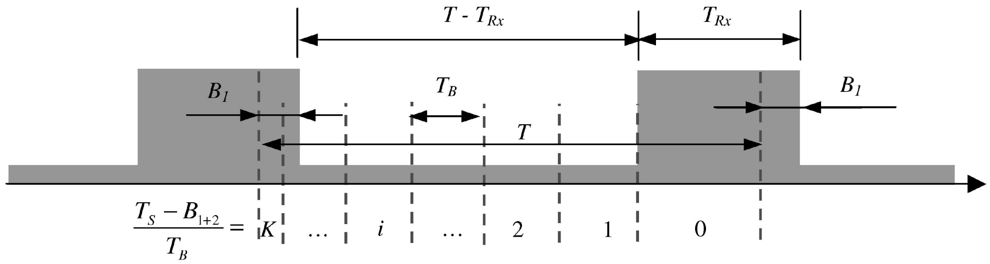


Fig. 11. Analysis of the setup latency of STEM-B without collisions.

density. However, whether density should be leveraged and which variant of STEM is most energy efficient, depends on the impact of the protocol overhead. For practical settings, the combination of STEM and GAF can reduce the energy to 1 percent or less of that of a network without topology management. Alternatively, this results in an increase of the average node lifetime of a factor 100.

At design time, the settings of our protocol can be derived, positioning the network at the desired operating point in the density-latency-energy design space. In essence, this is just part of a more general and design tradeoff, which is impacted by the specific application, the layout of the network, the cost of the nodes, the desired network lifetime, and many other factors.

Appendix A

Derivation of the Setup Latency

First, we derive the setup latency (as defined in Section 4.1) for STEM-B. If there are no collisions, the target sends back an acknowledgment after it receives a beacon packet. As soon as the initiator receives this acknowledgment, the link is set up. The setup latency is thus a number of beacon periods plus the time to transmit the beacon that is received and get back an acknowledgment. Therefore, T_S is equal to B_{1+2} plus an integer multiple of the interbeacon spacing T_B, where we use the shorthand notation $B_{1+2} = B_1 + B_2$.

As the target and originator node are not synchronized, the beacon sending process starts at a random point in the cycle T of the target node. Fig. 11 shows the normalized values of T_S for different start times of the beacon sending process. In the region that is labeled i (i = 1..K), the setup latency is equal to $i \cdot T_B + B_{1+2}$. The reason is that beacon $i+1$ is the first one to fall entirely within the interval of length T_{Rx} when the target node's radio is on. The probability of being in region i is equal to the length of that region divided by T. As a result, for $T > T_{Rx}$, the statistics of T_S are derived from Fig. 11 as:

$$\begin{cases} P(T_S = B_{1+2}) = \frac{T_{Rx} - B_1}{T} \\ P(T_S = k \cdot T_B + B_{1+2}) = \frac{T_B}{T} \qquad k = 1...K \\ P(T_S = (K+1) \cdot T_B + B_{1+2}) = \frac{T - (T_{Rx} - B_1) - K \cdot T_B}{T} \end{cases} \tag{A1}$$

$$K = \left\lfloor \frac{T - (T_{Rx} - B_1)}{T_B} \right\rfloor.$$

Based on this equation and after some algebra, the average setup latency per hop can be calculated as being equal to:

$$\bar{T}_S = B_{1+2} + \frac{T - T_B}{2} - \varepsilon \cdot \left(1 - \frac{T_B + \varepsilon}{2 \cdot T}\right) + \delta \cdot (1 - \delta) \cdot \frac{T_B^2}{2 \cdot T}. \tag{A2}$$

The variables δ and ε, which we introduced to simplify the notation of (A2), are defined as:

$$\delta = \frac{T - (T_{Rx} - B_1)}{T_B} - K, \tag{A3}$$

$$\varepsilon = T_{Rx} - T_B - B_1. \tag{A4}$$

We have verified that, in practical scenarios, the last term in (A2) is negligible, resulting in:

$$\bar{T}_S = B_{1+2} + \frac{T - T_B}{2} - \varepsilon \cdot \left(1 - \frac{T_B + \varepsilon}{2 \cdot T}\right). \tag{A5}$$

In addition, T is typically substantially larger than T_{Rx} such that we can further simplify this expression to:

$$\begin{aligned} \bar{T}_S &= B_{1+2} + \frac{T - T_B}{2} - \varepsilon \\ &= \frac{T + T_B}{2} + 2 \cdot B_1 + B_2 - T_{Rx}. \end{aligned} \tag{A6}$$

From (A1) and Fig. 10, we also learn that the maximum setup latency is equal to:

$$T_S^{\max} = (K+1) \cdot T_B + B_{1+2}. \tag{A7}$$

Furthermore, K is maximal when (A3) is equal to 0. Therefore, we can bound the setup latency by:

$$T_S^{\max} \leq T - \varepsilon + B_{1+2}. \tag{A8}$$

If the initiator sends out beacons for this time period, the target is guaranteed to have received it unless there was a collision or the packet got corrupted. However, in the case of a collision or corrupted packet, the receiver has awakened as well, as we explained in Section 3.3. In any case, the initiator can stop the beacon sending process after T_S^{max}, given by (A8). When a beacon collision occurs, the setup latency is also given by this equation.

The analysis of STEM-T is similar to the collision scenario in STEM-B. For sufficient noise tolerance, we propose an integrative detector. The target detects the tone if it is integrated for at least an interval T_I. Fig. 12 shows the worst-case scenario, where the tone starts too late to be

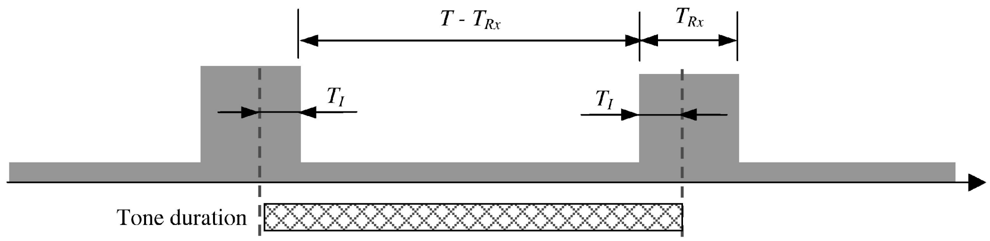


Fig. 12. Analysis of the worst-case setup latency of STEM-T.

detected in the first wakeup period. From this figure, we see that the minimum tone duration to guarantee detection is equal to (A9), which is therefore also the setup latency.

$$T_S = T - T_{Rx} + 2 \cdot T_I. \tag{A9}$$

Appendix B

Derivation of the Energy Consumption

When running STEM, the total energy consumed by a node during a time interval t can be broken up into two components, one for each frequency band.

$$E_{node} = E_{wakeup} + E_{data}. \tag{A10}$$

Equation (A11) details the energy consumption in the wakeup plane. The first term accounts for the listening cycle, where $P_{node,2}$ is given by (A12). $P_{listen,2}$ denotes the power during the periodic listen interval, which contains contributions of idle and receive power. The second term in (A11) represents the energy of sending beacon packets and listening for the acknowledgment (STEM-B) or the energy of sending a tone (STEM-T). It therefore corresponds to the energy spend as an initiator, where the average power $P_{setup,2}$ has contributions of transmission, reception, and idle power. $P_{sleep,2}$ is the sleep power of the wakeup radio.

$$E_{wakeup} = P_{node,2} \cdot (t - t_{setup}) + P_{setup,2} \cdot t_{setup}, \tag{A11}$$

$$P_{node,2} = \frac{P_{sleep,2} \cdot (T - T_{Rx}) + P_{listen,2} \cdot T_{Rx}}{T}. \tag{A12}$$

The energy consumption in the data plane is given by (A13). In this equation, t_{data} is the total time the radio is turned on in the data plane. As a result, $P_{data,1}$ contains contributions of packet transmission, packet reception, and idle power.

$$E_{data} = P_{sleep,1} \cdot (t - t_{data}) + P_{data,1} \cdot t_{data}. \tag{A13}$$

For ease of comparison, we normalize the energy consumption of STEM to a scenario where there is only one radio that is never in the sleep state, see (A14) and (A15). In (A15), P_1 is as defined in Section 4.2.

$$\frac{E}{E_0} = \frac{E_{node}}{E_{node}^{original}}, \tag{A14}$$

$$E_{node}^{original} = P_1 \cdot t. \tag{A15}$$

As explained in Section 4.2, we approximate $P_{data,1} \approx P_1$ and $P_{setup,2} \approx P_{listen,2} \approx P_2$. Furthermore, we note that $P_{sleep,i} << P_i \ (i = 1, 2)$, which allows us to write the relative energy of (A14) as (A16), after appropriate simplifications. Variables ρ and ϕ are defined in (4) and (5).

$$\frac{E}{E_0} = \frac{T_{Rx}}{T} \cdot \rho + \frac{t_{data}}{t} + \frac{t_{setup}}{t} \cdot \left(1 - \frac{T_{Rx}}{T}\right) \cdot \rho + \phi. \tag{A16}$$

In this equation, t_{setup} is the total time spent setting up the link in the wakeup plane as an initiator. The ratio of t_{setup} and the total time t can be rewritten as the setup frequency, f_S, times the duration of one setup T_S. When T is not too small, we can also make the following simplification:

$$\left(1 - \frac{T_{Rx}}{T}\right) \approx 1. \tag{A17}$$

Similarly, t_{data} can be split up in bursts of average duration t_{burst}, where a burst of data transfer requires one link setup. Consequently, the fraction of time the data-plane radio is turned on, which we define as α, is written as (A18), corresponding to (6). Here, f_W is the wakeup frequency. It can be higher than the setup frequency as a node wakes up when it is the initiator, the target, or an unintended receiver that hears a collision (STEM-B) or a tone (STEM-T).

$$\alpha = \frac{t_{data}}{t} = f_W \cdot t_{burst}. \tag{A18}$$

Finally, by defining β as in (7), (A16) becomes:

$$\frac{E}{E_0} = \frac{\rho}{\beta} + \alpha + f_S \cdot T_S \cdot \rho + \phi. \tag{A19}$$

If the wakeup radio would be turned off once the data radio is turned on, the derivation is similar to the one above. With the same simplifications, the final equation is exactly the same as (A19). This situation corresponds to STEM-T, where there is no need to leave the wakeup radio on once the data radio has been activated.

Acknowledgments

This paper is based in part on research funded by the US Office of Naval Research and the US Defense Advanced Research Projects Agency PAC/C and SenseIT programs through US Air Force Research Laboratory contracts F30602-00-C-0154 and F30602-99-1-0529. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the ONR, DARPA, Air Force Rome Laboratory, or the US Government.

REFERENCES

[1] K. Sohrabi, J. Gao, V. Ailawadhi, and G. Pottie, "Protocols for Self-Organization of a Wireless Sensor Network," *IEEE Personal Comm. Magazine*, vol. 7, no. 5, pp. 16-27, Oct. 2000.

[2] L. Clare, G. Pottie, and J. Agre, "Self-Organizing Distributed Sensor Networks," *SPIE—The Int'l Soc. Optical Eng.*, pp. 229-237, Apr. 1999.

[3] D. Estrin and R. Govindan, "Next Century Challenges: Scalable Coordination in Sensor Networks," *Proc. MobiCom 1999*, pp. 263-270, Aug. 1999.

[4] V. Raghunathan, C. Schurgers, S. Park, and M. Srivastava, "Energy-Aware Wireless Sensor Networks," *IEEE Signal Processing*, vol. 19, no. 2, pp. 40-50, Mar. 2002.

[5] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks," *Proc. MobiCom 2001*, pp. 70-84, July 2001.

[6] Y. Xu, J. Heidemann, and D. Estrin, "Geography-Informed Energy Conservation for Ad Hoc Routing," *Proc. MobiCom 2001*, pp. 70-84, July 2001.

[7] J.-H. Chang and L. Tassiulas, "Energy Conserving Routing in Wireless Ad-Hoc Networks," *Proc. INFOCOM 2000*, pp. 22-31, Mar. 2000.

[8] J. Rabaey, J. Ammer, J.L. da Silva, and D. Patel, "PicoRadio: Ad-Hoc Wireless Networking of Ubiquitous Low-Energy Sensor/Monitor Nodes," *Proc. IEEE CS Workshop VLSI 2000*, pp. 9-12, Apr. 2000.

[9] W. Rabiner Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," *Proc. Hawaii Int'l Conf. System Sciences 2000*, Jan. 2000.

[10] C. Guo, L. Zhong, and J. Rabaey, "Low-Power Distributed MAC for Ad Hoc Sensor Radio Networks," *Proc. Internet Performance Symp. (Globecom '01)*, Nov. 2001.

[11] Sensoria Corporation, http://www.sensoria.com/, May 2002.

[12] R. Bagrodia, R. Meyer, M. Takai, Y.A. Chan, X. Zeng, J. Marting, and H.Y. Song, "Parsec: A Parallel Simulation Environment for Complex Systems," *Computer*, vol. 31, no. 10, pp. 77-85, Oct. 1998.

[13] M. Yacoub, *Foundations of Mobile Radio Engineering*. CRC Press, 1993.

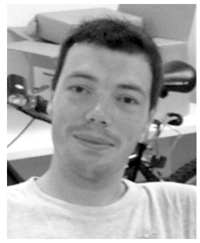
[14] M. McGlynn and S. Borbash, "Birthday Protocols for Low Energy Deployment and Flexible Neighbor Discovery in Ad Hoc Wireless Networks," *Proc. MobiHoc 2001*, pp. 137-145, Oct. 2001.

[15] "ASH Transceiver Designer's Guide," http://www.rfm.com, May 2002.

[16] W. Ye, J. Heidemann, and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," *IEEE Infocom '02*, June 2002.

[17] "MICA motes at Crossbow," http://www.xbow.com/Products/Wireless_Sensor_Networks.htm, May 2002.

[18] J. Proakis, *Digital Communications*, third ed. McGraw-Hill Series in Electrical and Computer Engineering, 1995.



Curt Schurgers received the MSEE degree, summa cum laude, from the Katholieke Universiteit Leuven (KUL), Belgium, in 1997. From 1997 to 1999, he worked at IMEC (Interuniversity Micro Electronics Center, Leuven, Belgium) on memory optimization techniques for turbo codes. Currently, he is pursuing a PhD degree at the University of Califirnia at Los Angeles, focusing on energy efficient communication and networking systems. He has received the F.W.O., the B.A.E.F., and the UCLA fellowships in 1997, 1999, and 2000, respectively. He is a student member of the IEEE.

Vlasios Tsiatsis received the BS degree from the Technical University of Crete, Chania, Greece, in 1998, and the MS degree in electrical engineering from the University of California, Los Angeles (UCLA), in 2001. He is currently pursuing a PhD degree at UCLA, researching low-power protocol architectures. In 1998 and 2000, he was honored with two UCLA fellowships. He is a student member of the IEEE.

Saurabh Ganeriwal received the Btech and Mtech degrees in the field of communication and signal processing from the Indian Institute of Technology, Bombay, India, in 2001. Since 2001, he has been pursuing an MS degree at the University of California, Los Angeles. His research interests are in wireless communications and networking. Currently, his research focuses on sensor networks.

Mani Srivastava received the BTech degree in electrical engineering from IIT Kanpur in India and the MS and PhD degrees from the University of California at Berkeley. He is an associate professor of electrical engineering at the University California at Los Angeles. From 1992 through 1996, he was a member of the technical staff at Bell Laboratories in networked computing research. His current research interests are in mobile and wireless networked computing systems, low-power systems, and sensor networks. He received the US National Science Foundation CAREER award in 1997 and the President of India Gold Medal in 1985. He is a senior member of the IEEE.

▷ **For more information on this or any computing topic, please visit our Digital Library at** http://computer.org/publications/dilb.