Telecommun Syst
DOI 10.1007/s11235-009-9228-z

# Modelling the energy cost of a fully operational wireless sensor network

**Waltenegus Dargie · Xiaojuan Chao · Mieso K. Denko**



**Abstract** Several applications have been proposed for wireless sensor networks, including habitat monitoring, structural health monitoring, pipeline monitoring, precision agriculture, active volcano monitoring, and many more. The energy consumption of these applications is a critical feasibility metric that defines the scope and usefulness of wireless sensor networks. This paper provides a comprehensive energy model for a fully functional wireless sensor network. While the model uses toxic gas detection in oil refineries as an example application, it can easily be generalized. The model provides a sufficient insight about the energy demand of the existing or proposed communication protocols.

**Keywords** Wireless sensor networks · Energy-model · Energy-efficient protocols · Lifetime of a wireless sensor network

## 1 Introduction

Several applications have been proposed for wireless sensor networks. The application of Mainwaring et al. [12] gathers data from humidity, temperature, barometric pressure, and light sensors for monitoring the activities of seabirds. Kim et al. [10] use wireless sensor networks for structural health monitoring, in which the structural integrity of bridges and buildings is inspected using accelerometer sensors. The networks are tasked with measuring the response of a structure to an ambient excitation (heavy wind or passing vehicles) or a forced shake (using shakers or impact hammers). The application of Werner-Allan et al. [17] monitors active volcano using seismic and infrasonic sensors. The underlying network was able to capture 230 volcano events just over three weeks. The application of Stoianov et al. [15] uses hydraulic and acoustic/vibration sensors for monitoring large diameter, bulk-water transmission pipelines.

The most prevalent concern in wireless sensor networks is the limited lifetime. The nodes operate with exhaustible batteries; and recharging or replacing these batteries, given the sheer size of the network and the deployment settings, is a significant hurdle. For example, because of the energy constraint, Kim et al. [10] suggest that wireless sensor networks can only be used during occasional inspection of bridges and buildings, thereby limiting their scope as well as usefulness. Subsequently, almost all types of communication protocols and data processing algorithms target efficient use of energy and optimization of network lifetime as their design goal.

In this paper, we carefully analyse the energy cost of a fully operational wireless sensor network. The application we use for our analysis will be toxic gas detection in oil refineries. We will consider highly referenced, energy-aware protocols for establishing and running the network. We shall give particular consideration to the link and network layer as well as to the self-organization (neighbor discovery and interest dissemination) aspects, as these claim a significant portion of the energy budget. Finally, we shall provide comprehensive analytic and simulation models based on which

W. Dargie (✉)
Chair of Computer Networks, Technical University of Dresden, 01187 Dresden, Germany
e-mail: waltenegus.dargie@tu-dresden.de

X. Chao
9th Floor Tower B, CEC Plaza, NO. 3 Dan Ling Street, Hai Dian District, Beijing 100080, China
e-mail: chaoxj@gmail.com

M.K. Denko
Department of Computing and Information Science, University of Guelph, Guelph, Ontario, Canada, N1G 2W1
e-mail: denko@cis.uoguelph.ca

Published online: 05 November 2009

the lifetime of the network can be estimated. The models take into account node density, distributed sleeping schedules, multi-hop communication and time synchronization.

The rest of this paper is organized as follows: In Sect. 2, we discuss related work; in Sect. 3 we will briefly discuss toxic gas detection in refineries; in Sect. 4, we will establish basic assumptions for the network model; in Sects. 5 and 6, we will provide a comprehensive analysis and simulation of the energy cost. Finally in Sect. 7, we will discuss our experiences and observations and provide concluding remarks and outline for future work.

## 2 Related work

Tseng [16] provide an analytic energy model for estimating the energy consumption of a wireless sensor network that employs the S-MAC medium access control protocol [18]. The model takes the cost of control messages (RTS/CTS/ACK/DIFS) and the duty cycle of the sleeping schedule of individual nodes into account. The model attempts to define and estimate the energy consumption of various operation modes. In [19], an analytic, integrated data-link layer model is presented. The model enables to estimate the energy cost of link layer protocols. The strength of the model is in its capability to give insight about the effect of a link layer decision on other layer concerns, including channel assignment, rate of transmission, power and management. However, the framework does not offer a comprehensive understanding of the energy cost of the entire network.

Feeney [6] propose an analytical model for examining the energy cost of routing in a mobile ad hoc networks. The work attempts to demonstrate the trade-off between energy consumption and reliability. Two popular routing protocols are chosen for the analysis: Dynamic Source Routing (DSR) [13] and Ad hoc On-demand Distance Vector (AODV) [11]. These two protocols support routing in flat topology networks, with all nodes participating equally in the routing process. Moreover, both protocols are on-demand protocols, in which nodes discover and maintain routes as needed. DSR heavily depends on the cache of network wide topology information extracted from source routing headers, while AODV is a destination-oriented protocol based on the distributed Bellman-Ford algorithm. Both protocols are adaptive for dynamic topology. Similar to other energy models, the network interface has four possible energy consumption states: transmitting, receiving, idle, and sleeping. The idle mode is the default mode for ad hoc environment. The energy cost is calculated as a function of packet size. The unit energy of a packet is decided by the sender, the intended receiver(s), and the nodes overhearing the message. Chao et al. [4] report an initial result of this work, but its mathematical model was not fully developed.

## 3 Toxic gas detection

The application we use to analyse the energy cost of the communication protocols in wireless sensor networks is a toxic gas detection application in oil refineries. There are two reasons for choosing toxic gas detection: (1) Oil refineries cover extensive areas, requiring large scale sensing to detect oil and gas leakages in pipelines. This fits into the basic assumption that a wireless sensor network is made up of hundreds and thousands of wireless sensing nodes. (2) Presently, a good portion of the oil industry is replacing cable based sensing systems by portable and wireless devices which can easily be deployed and maintained. The next evolution in toxic gas detection will be towards wireless sensor networks. For the detail description of the various toxic gases that should be sensed, we refer our readers to [4] and [5].

## 4 Network model

Our analysis and simulation of the network's energy consumption and lifetime is based on a network model. The network model establishes the basic assumptions concerning the network's topology, the distribution and density of nodes, and the way the network is connected. Moreover, it defines the network's sensing task.

Deployment refers to the way wireless sensor nodes are placed in areas where the sensing task should be carried out. This decision directly affects the quality of sensing as well as the overall energy consumption of the entire network. While there can be three basic monitoring strategies—spot, area and fence—for toxic gas monitoring, spot monitoring is the most suitable strategy [4].

Coverage is another significant performance metric. In [2], it indicates how well a given area can be monitored by the network. Even though there are some existing models for estimating the number of sensors required to cover the entire sensing field with a probability, $p$, of detection an event, coverage is deployment dependent. For a spot monitoring scenario, even though the whole area is not necessarily covered, all potential leakage sources are monitored.

As shown in Fig. 1, $N$ nodes are distributed randomly on a rectangular area $A$ of size $A = a \times b$. Without loss of generality, we assume that $a \leq b$. The node distribution can be modelled as a two-dimensional Poisson distribution with average density, $\lambda$. The probability of finding $k$ nodes in $A$ is given by:

$$P(k\ nodes \in A) = e^{\lambda A} \frac{(\lambda A)^k}{k!} \tag{1}$$

The connectivity figure speaks about the existence of a communication link between a source anywhere in the network and a single sink. In multi-hop communication, there

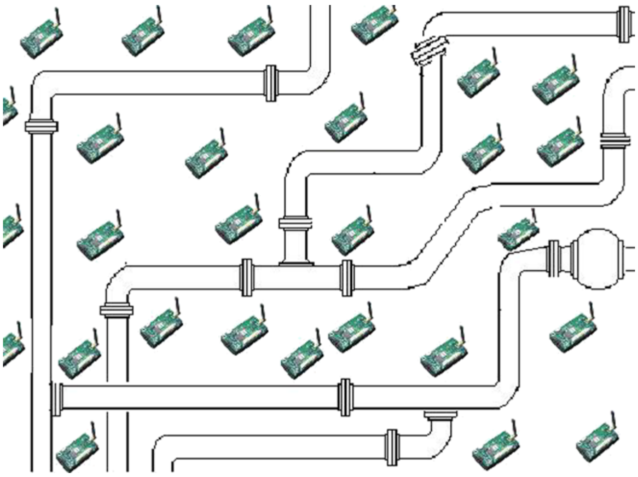

**Fig. 1** 2D Poisson distributed node deployment

is at least one multi-hop path between a node and the sink (base station). The probability that a network is connected, i.e., all nodes can communicate with the sink either directly or with the support of intermediate nodes, mainly depends on the node density and the transmission range of individual nodes. If the nodes are assumed to be homogeneous, the relationship between connectivity probability, transmission range and node density is estimated by[1] [3]:

$$p(conn) \cong (1 - e^{\lambda \pi r_0^2})^n \tag{2}$$

where $p(conn)$ is the probability that the network is connected; $\lambda$ is the density of the network, $n/A$; $r_o$ is the threshold transmission rage; and $n \gg 1$ is the number of deployed nodes. The deployment scenario for our case is depicted in Fig. 1. The spot monitoring strategy is complemented by additional randomly deployed nodes for improved connectivity. Each node has the same radio transmission range $R$, and two nodes can communicate via a wireless link if their Euclidean distance is less that the transmission range, i.e., $d \leq R$. For simplification, fading and path efficiency are not taken into account; we do not consider also the presence of obstacles in the path of propagation.

Finally, the sensing task for which the network is deployed determines the data traffic size in the network. For toxic gas detection, there are two essential concerns: the long and short term impact of toxic gases release. Hence, every sensor node should periodically (a tunable parameter) report the concentration of $H_2S$ and $NH_3$ to a sink. This is define as a normal case with a normal priority. In case of a leakage that surpasses a threshold defined by the safety board of the refinery (this is usually a concentration between 10 and 15 ppm, an alarm should be fired off within 30 seconds. This is characterized as an abnormal condition with high priority.

[1]This is without taking the border effect into account.

## 5 Energy model

In Sect. 4, we presented a number of factors that affect the quality of sensing and the lifetime of a wireless sensor network. In this section, we shall translate those factors into quantifiable terms so that we can estimate the energy cost. The model together with the sensing task description, and the specification of the hardware devices and the communication protocols will be sufficient to estimate the lifetime.

The communication protocols we employ to establish the wireless sensor network are the S-MAC [18], for medium access control, and the Directed Diffusion [8], for supporting self-organization and routing. The justification for these protocols is given in more detail elsewhere [4]. A more technical assessment of these protocols can be found in [20] and [21].

Hop count is an essential performance parameter and indicates how many hops a packet is relayed in average for a given distance in a network. For a deterministic topology, this hop-count estimation is a simple geometry problem. For a random network model, however, a combination of statistics and probability theory is required. Fortunately, there are many existing models already. To calculate the minimum hop count, we determine the distance $S$ between two random nodes and divide it by the transmission range $R$. In the literature, the random distance formula [1] is widely adopted. It is based on the calculation of the random distance distribution within a rectangular area:

$$\begin{aligned} E\{S\} = {} & \frac{1}{15}\left[\frac{a^3}{b^2} + \frac{b^3}{a^2} + \sqrt{a^2+b^2}\left(3 - \frac{a^2}{b^2} - \frac{b^2}{a^2}\right)\right] \\ & + \frac{1}{6}\left[\frac{b^2}{a}\ln\left(\frac{a+\sqrt{a^2+b^2}}{b}\right)\right. \\ & \left. + \frac{a^2}{b}\ln\left(\frac{b+\sqrt{a^2+b^2}}{a}\right)\right] \end{aligned} \tag{3}$$

Taking $E(S)$ as the expected distance between the source and the destination in our random network, the lower bound of the expected value of hop count can be expressed as:

$$H_{\min} = E\{S\}/R \tag{4}$$

To achieve a more realistic analysis, transmission error due to packet loss and collision should be included in the energy model. Because the listening time in S-MAC is fixed, a fixed contention window is better for coordination and synchronization than an exponential back-off. However, a fixed contention window can cause significant packet loss. We used Bianchi's model [7] to estimate packet loss due to collision at the link layer. According to the model, the prob-

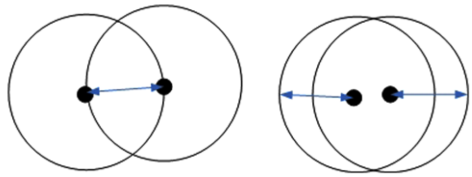

**Fig. 2** Possible intersections of two neighbor nodes

ability of successful transmission, $p_{succ}$, can be calculated as:

$$p_{succ} = \frac{(\lambda - 1)\tau(1-\tau)^{\lambda-2}}{1-(1-\tau)^{\lambda-1}} \tag{5}$$

where $\lambda$ refers to the network's node density and $\tau = \frac{2}{(CW+1)}$ and $CW$ is the carrier sensing contention window.

All data packets in S-MAC, except interest dissemination, are unicast and will cause $RTS/CTS/ACK$ control overhead. To estimate the energy cost of adaptive listening, it is useful to estimate the number of neighbors which potentially overhear the $RTS/CTS$ message, i.e., the average neighbors in enclosure of the sender and the receiver. This can be calculated by first getting the overlaps of communication coverage between two random neighbors. When two nodes becomes neighbors, their transmission circles intersect, in which case Fig. 2 shows the two extreme scenarios. The intersection area can be described as [14]:

$$2R^2\cos^{-1}\left(\frac{d}{2R}\right) - \frac{1}{2}d\sqrt{4R^2-d^2} \tag{6}$$

where $0 \le d \le R$ and $d$ is the Euclidean distance between two nodes. Taking the assumption of the random Poisson distribution of the nodes into account, then $d$ is bound in $(0, R)$ with a uniform probability distribution. Accordingly, $d$ will be:

$$d = \frac{R}{2} \tag{7}$$

Then the average overlap of two circles can be described by:

$$A_{\text{intersect}} \approx 2.152R^2 \tag{8}$$

The enclosure area for neighbors of a sender or a receiver is

$$A_1 = 2\pi R^2 - 2.152R^2 \approx 4.131R^2 \tag{9}$$

with

$$\lambda = \frac{N \times \pi R^2}{a \times b} \tag{10}$$

And,

$$N_{neighb} = \frac{A_1}{a \times b} \times N \approx \frac{A_1}{\pi R^2} \times \lambda = 1.314\lambda \tag{11}$$

## 5.1 Energy consumption analytic model

For a thorough analysis of the energy model (from (12) to (65)), the variables (parameters) listed in Table 1 and their corresponding descriptions should be referred to. Additional variables will be explained according to their context of use.

We propose two analytic models to estimate the energy consumption of a toxic gas detection network. We call the first model Pure Synchronization Energy Model (PSE) and the other Full Application Energy Model (FAE). In PSE, there will not be data transmission in the network; nodes communicate with each other to perform synchronization (i.e., exchanging sleeping schedules). Most existing S-MAC based energy models assume that the whole network is synchronized without actually considering the energy consumed by the synchronization process. We present the PSE model to provide a realistic picture of the contribution of time synchronization on the overall energy consumption. In the simulation section, we shall demonstrate that synchronization and periodical neighbor discovery cost more energy than data transmission. FAE models a fully functional network in which both periodical and incidental data transmission and time synchronization are taking place.

### *5.1.1 Pure synchronization energy model*

S-MAC carries out time synchronization in 4 steps: In the first step, every node is initially active for $sync_p$ cycles, waiting for the arrival of SYNC packet from other nodes. The energy consumption of this phase is expressed as:

$$E_{listen\_sync} = N \times sync_p \times T_{frame} \times P_{idle} \tag{12}$$

In the second step, nodes periodically resynchronize to avoid clock drift. During this time, $t$, the number of attempts every node sends SYNC packet is expressed as,

$$N_{sync\_sent\_try} = \left(t / \left(sync_p \times T_{frame}\right) - sync_p\right) \tag{13}$$

Due to packet collision and loss, only a portion of these packets will be successfully received:

$$N_{sync\_sent} = N_{sync\_sent\_try} \times p_{succ} \tag{14}$$

The energy consumed during sending SYNC packets at this stage is given by:

$$E_{sync\_per\_node\_sent} = E_{trans} + E_{idle} + E_{sleep} \tag{15}$$

where

$$E_{trans} = M_{sync}/R_{data\_rate} \times P_{trans} \tag{16}$$

**Table 1** Variables definition

| Variables | Definition |
|---|---|
| $P_{trans}$, $P_{sleep}$, $P_{recv}$, $P_{idle}$ | Energy consumption per time unit of four modes |
| $M_{sync}$, $M_{RTS}$, $M_{CTS}$, $M_{ACK}$, $M_{interest}$, $M_{data}$ | Size of SYNC, RTS, CTS, ACK, Interest, and data Message |
| $t_{sync\_cw}$, $t_{data\_cw}$ | Size of contention window of SYNC and Data Message |
| $t_{backoff}$, $t_{DIFS}$, $t_{SIFS}$ | Size of backoff, DCF, Inter Frame Space and Short Inter Frame Space |
| $t_{idle}$ | Idle period of every transmission/reception pair of one data packet |
| $t_{adapt}$ | Adaptation time, frame length dependent |
| $I_{normal}$, $I_{abnormal}$ | Interest propagation frequency for Normal and Abnormal case |
| $i_{normal}$, $i_{abnormal}$ | Normal and Abnormal event report interval |
| $d_{report\_abnormal}$ | Report duration after a leak is detected |
| $R_{data\_rate}$ | Data rate |
| $duty\_cycle$ | Duty cycle |
| $T_{frame}$ | Frame length |
| $p_{succ}$ | Probability of successful packet transmission/reception |
| $R_{retry}$ | Max retry times |
| $f_{srch\_cycle}$ | Frequency of neighbor Discovery |
| $sync_p$ | The initial Synchronization period |
| $N_{neigh}$ | Average neighbors in enclosure of sender and receiver |
| $\lambda$ | Network density |
| $N$ | Total number of nodes in an area, $A$ |
| $N_{leak}$ | The number of nodes that detected leakage |
| $H_{min}$ | Minimum hop count (Topology dependent) |

And,

$$E_{idle} = P_{idle} \times (T_{frame} \times duty\_cycle - M_{sync}/R_{data\_rate}) \tag{17}$$

$$E_{sleep} = P_{sleep} \times T_{frame} \times (1 - duty\_cycle) \tag{18}$$

$$E_{sync\_per\_node\_recv} = (1 - \lambda) \times (E_{recv} + E_{idle\_recv} + E_{sleep\_recv}) \tag{19}$$

The energy consumed during receiving the SYNC packets is expressed as,

$$E_{recv} = (M_{sync}/R_{data\_rate} \times P_{recv}) \tag{20}$$

When a SYNC packet arrives at a receiving node, it either succeeds or fails due to collision or channel error. Since a failure reception consumes the same amount of energy as a successfully received packet, we merge both scenarios together. In other words, all $(\lambda - 1)$ neighbor nodes will receive the SYNC packet regardless of its usefulness. Then total amount of energy consumed during periodical SYNC packet sending and receiving is therefore calculated as:

$$E_{period\_sync\_pure} = N_{sync\_sent\_try} \times N \times E_{sync\_per\_node\_sent} + N_{sync\_sent} \times N \times E_{sync\_per\_node\_recv} \tag{21}$$

In the third step, every node periodically performs neighbor discovery by listening for the whole $sync_p$ cycles as described in the first stage, i.e., for every $sync_p \times f_{srch\_cycle}$ cycles. Note, however, that not all node enter into neighbor discover phase at the same time since those nodes that lose during contention for channel access will compete only in the next contention cycle, after sending a SYNC packet. Thus the periodical neighbor discovery will be delayed due to collision.

$$E_{nb\_srch} = N \times sync_p \times p_{succ} \times T_{frame} \times P_{idle} \times \frac{t}{T_{frame} \times sync_p \times f_{srch\_cycle}} \tag{22}$$

Finally, to calculate the energy consumption of transmitting empty frames,[2] first, we find out the number of empty frames.

$$N_{empty} = (t/T_{frame} - sync_p - (t \times p_{succ}) /(T_{frame} \times f_{srch\_cycle}) - N_{sync\_sent\_try} + N_{sync\_sent} \times (\lambda - 1)) \times N \tag{23}$$

Here $t/T_{frame}$ gives the total number of frames per node for the period, $t$. $sync_p$ is the probable duration of the initial synchronization time in which a node waits for SYNC packet from other nodes. $\frac{t}{(T_{frame} \times f_{srch\_cycle})}$ is the neighbor discovery frames; and $N_{sync\_sent} \times \lambda$ expresses the number of frames for sending and transmitting periodical SYNC packets. Subsequently, the expected energy consumption for synchronization is expressed as:

$$E_{empty} = N_{empty} \times T_{frame} \times (P_{idle} \times duty\_cycle + P_{sleep} \times (1 - duty\_cycle)) \tag{24}$$

The energy consumption for SYNC overhead without data transmission is given by:

$$E_{sync\_pure} = E_{listen\_sync} + E_{period\_sync\_pure} + E_{nb\_srch} + E_{empty} \tag{25}$$

### 5.2 Full application energy model

In this model, the energy model contains two parts: the energy consumption due to synchronization and the energy consumption due to data transmission. The four stages of synchronization discussed in Sect. 5.1.1 apply for the Full Application Energy Model as well. The amount of energy consumed during listening for Sync packets and neighbor discovery is the same as in the previous case. However, even though both SYNC and data packets can be processed in the same frame, in stage 2 of the synchronization stage, we calculated only the energy for sending/receiving SYNC packets. The energy consumed during the remaining time can be accounted for data transmission or receiving; or for idly listening. Suppose $N_{sync\_sent}$ is the number of times every node sends SYNC packets successfully and $N_{sync\_sent\_try}$ is the number of times a node broadcasts SYNC packets. The energy consumption during sending and receiving every SYNC packet is given as follows:

$$E_{sync\_per\_node\_sent} = \frac{M_{sync}}{R_{data\_rate}} \times P_{trans} + P_{idle} \times (t_{sync\_cw} + t_{backoff} + t_{DIFS}) \tag{26}$$

$$E_{sync\_per\_node\_recv} = (\lambda - 1) \times M_{sync}/R_{data\_rate} \times P_{recv} \tag{27}$$

Because of the reason stated in step 2 of the PSE model, we merge both scenarios together.

$$E_{period\_sync} = N_{sync\_sent\_try} \times N \times E_{sync\_per\_node\_sent} + N_{sync\_sent} \times N \times E_{sync\_per\_node\_recv} \tag{28}$$

Unlike stage 4 in Pure Synchronization Model, the empty frames in this scenario are both data and SYNC packets dependent; thus the total energy consumed during synchronization is given as follows:

$$E_{sync} = E_{listen\_sync} + E_{period\_sync} + E_{nb\_srch} \tag{29}$$

Given the average neighbors in enclosure of two nodes, the number of neighbors that overhear an RTS/CTS message can be known.[3] We use the $p_{succ} \times N_{neigh}$ to denote the number of nodes that will join adaptive listening. The energy consumption due to nodes participating in an Adaptive Listening is given by:

$$E_{adapt} = (p_{succ} \times N_{neigh} \times t_{adapt} + t_{idle}) \times P_{idle} \tag{30}$$

In One-phase pull of the Directed Diffusion routing protocol, there are no exploratory and reinforcement overheads. One only needs to calculate the cost of flooding interest and data transmission. In the Interest propagation phase, the sink periodically sends interest to all nodes. The duration of a period is relatively large. The successful transmission and retransmission rate are described in Table 2.

$$\alpha = p_{succ} + (1 - p_{succ}) \times p_{succ} \times 2 + (1 - p_{succ})^2 \times p_{succ} \times 3 + \cdots + (1 - p_{succ})^{R_{retry}-1} \times p_{succ} \times R_{retry} \tag{31}$$

Based on the back off behavior of S-MAC, for every transmission/reception pair of one data packet, the idle period can be described as follows.

$$t_{idle} = t_{data\_cw} + t_{backoff} + t_{DIFS} + 3t_{SIFS} \tag{32}$$

[2] Here we define empty frames as frames that contain scheduled idle time only. In these frames, we need only calculate the energy consumed during idle time.

[3] There are also nodes that may not be able to hear an RTS/CTS message.

**Table 2** Transmission times of RTS

| Number of transmission times | Send/Resend Possibility | Send Times of $M_{RTS}$ |
|---|---|---|
| 1 | $p_{succ}$ | 1 |
| 2 | $(1 - p_{succ}) \times p_{succ}$ | 2 |
| 3 | $(1 - p_{succ})^2 \times p_{succ}$ | 3 |
| $R_{retry}$ | $(1 - p_{succ})^{(R_{retry}-1)} \times p_{succ}$ | $R_{retry}$ |

When a node finishes transmitting/receiving a packet, the remaining time may not always fit to the scheduled active and sleep time of the node, in which case the node has to keep idle until the next active or sleep time arrives. Since we already take the active period in one frame into account, the extra idle time can be estimated by: $(1 - duty\_cycle) \times \frac{T_{frame}}{2}$. Accordingly, the energy consumption of interest propagation can be expressed as:

$$E_{interest\_per\_node} = E_{useful} + E_{waste} \tag{33}$$

where

$$E_{useful} = M_{interest}/R_{data\_rate} \times (P_{trans} + p_{succ} \times (\lambda - 1) \times P_{recv}) \tag{34}$$

And,

$$E_{waste} = (t_{DIFS} + t_{data\_cw} + t_{backoff}) \times P_{idle} + (1 + (\lambda - 1) \times p_{succ}) \times (P_{idle} + P_{sleep}) \times 1 - duty\_cycle) \times \frac{T_{frame}}{2} \tag{35}$$

Every interest packet is successfully transmitted with a probability of $p_{succ}$. This holds true for both normal and abnormal conditions.

$$E_{normal\_set} = E_{abnormal\_set} = N \times E_{interest\_per\_node} \tag{36}$$

During a reporting phase, we have either a normal event or an abnormal event. During a normal report, the $H_2S$ concentration is below 10 ppm. We first calculate the energy consumption of a single event delivery path. Every packet along a single path will be received $H_{min}$ times. It will be forwarded to the next hop if the concentration is larger than the max value in memory of the current node. The possibility of every intermediate packet being successfully forwarded is assumed to be 0.5. Thus,

$$E_{normal\_report\_one\_path} = H_{min} \times (E_{OH} + E_{trans} + E_{waste}) \tag{37}$$

where

$$E_{OH} = (M_{data} + M_{RTS} + M_{CTS} + M_{ACK}) / R_{data\_rate} \times P_{recv} \tag{38}$$

And,

$$E_{tran} = 0.5 \times (M_{data} + M_{RTS} \times \alpha + M_{CTS} + M_{ACK})/R_{data\_rate} \times P_{trans} \tag{39}$$

$$E_{waste} = 0.5 \times E_{adapt} \times \alpha + (P_{idle} + P_{sleep}) \times (1 - duty\_cycle) \times \frac{T_{frame}}{2} \tag{40}$$

The energy consumption during an abnormal case can be calculated in a similar way. Based on the result of the two phases above, we derive the energy consumption by $N$ nodes during time, $t$. This includes the energy consumption of interest propagation phase and reporting phase:

$$E_{routing\_normal} = \frac{t}{I_{normal}} \times E_{normal\_set} + \frac{t}{i_{normal}} \times E_{normal\_report\_one\_path} \times N \tag{41}$$

Similarly, the energy consumption for the abnormal case is expressed as follows:

$$E_{routing\_abnormal} = \frac{t}{I_{abnormal}} \times E_{abnormal\_set} + \frac{d_{report\_abnormal}}{i_{abnormal}} \times E_{abnormal\_report\_one\_path} \times N_{leak} \tag{42}$$

Here $\frac{d_{report\_abnormal}}{i_{abnormal}}$ refers to the number of messages that a leakage event keeps on reporting.

### 5.2.1 Energy for empty frames and missed part

As mentioned before, we now express the energy consumed in idle time of empty frames that are not used during synchronization or data transmission/reception during time $t$. To estimate the number of empty frames, we calculate the number of frames occupied in routing and synchronization based on the analysis above. From the interest propagation phase, we get the number of frames for both cases:

$$N_{normal\_set} = N_{abnormal\_set} = N \times \lambda \tag{43}$$

The reporting frame is defined as

$$N_{normal\_one\_path} = (1 + 0.5 \times \alpha) \times H_{\min} \tag{44}$$

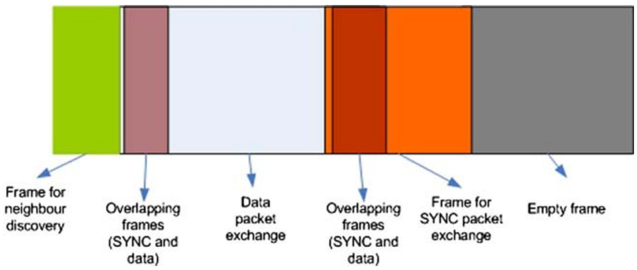


**Fig. 3** Ratio among number of frames

$$N_{abnormal\_one\_path} = (1+\alpha) \times H_{\min} \quad (45)$$

Now with the above intermediate calculation, we derive the number of frames required for data exchange:

$$N_{data} = F_{comb} + F_{normal} + F_{ab} \quad (46)$$

where,

$$F_{comb} = \frac{t}{I_{normal}} \times N_{normal\_set} + \frac{t}{I_{abnormal}} \times N_{abnormal\_set} \quad (47)$$

And,

$$F_{normal} = \frac{t}{i_{normal}} \times (N-1) \times N_{normal\_one\_path} \quad (48)$$

$$F_{abnormal} = \frac{d_{report\_abnormal}}{i_{abnormal}} \times N_{leak} \times N_{abnormal\_one\_path} \quad (49)$$

Then with the number of frames for neighbor discovery and SYNC packets exchanged, we compute at the $n$ synchronization stage, the number of frames for synchronization:

$$N_{sync\_neighbor\_discovery} = \left( sync_p + \frac{t}{(T_{frame} \times f_{srch\_cycle})} \times p_{succ} \right) \times N \quad (50)$$

$$N_{sync\_exchange} = (N_{sync\_sent\_try} + N_{sync\_sent} \times (\lambda - 1)) \times N \quad (51)$$

$$N_{sync} = N_{sync\_neighbor\_discovery} + N_{sync\_exchange} \quad (52)$$

Figure 3 shows the ratio among a number of frames used for synchronization, data, or empty $\varphi_1$ and $\varphi_2$ are the number of frames that handle both SYNC and data packets. While $\varphi_1$ represents the overlap between frames of data and neighbor discovery (the whole frame is in idle state), and $\varphi_2$ denotes the overlap between frames of data and common SYNC packets exchange.

It is difficult to precisely determine how many frames a node uses for both SYNC and data transmitting/receiving. We divide the intersection between data and synchronization into sub periods as $\varphi_1$ and $\varphi_2$ to decrease the uncertainty. By adding frames on both payload and synchronization, we can estimate the total frames produced by a node. This is expressed as follows:

$$N_{work} = N_{data} + N_{sync} - \varphi_1 - \varphi_2 \quad (53)$$

The total number of frames communicated during $t$ is:

$$N_{total} = t / T_{frame} \times N \quad (54)$$

So the number of empty frames can be calculated by subtracting $N_{work}$ from $N_{total}$.

$$N_{empty} = N_{total} - N_{data} - N_{sync\_neighbor\_discovery} - N_{sync\_exchange} + \varphi_1 + \varphi_2 \quad (55)$$

Accordingly ,we get the energy consumed by empty frames:

$$E_{empty} = N_{empty} \times T_{frame} \times (duty\_cycle \times P_{idle} + (1 - duty\_cycle) \times P_{sleep}) \quad (56)$$

As we mentioned before, we only calculate the energy for SYNC packets exchange till now, we need to add the missing part here. From the Fig. 3, the $N_{sync\_miss}$ can be calculated as follows:

$$N_{sync\_miss} = N_{sync\_exchange} - \varphi_2 \quad (57)$$

Thus

$$E_{sync\_miss} = (N_{sync\_exchange} - \varphi_2) \times (P_{idle} \times (T_{frame} \times duty\_cycle - M_{sync}/R_{data\_rate}) + P_{sleep} \times T_{frame} \times (1 - duty\_cycle)) \quad (58)$$

If we add $E_{empty}$ and $E_{sync\_miss}$, we can get

$$E_{empty} + E_{sync\_miss} = N + \times T_{frame} \times N_{sleep\_idle} - (N_{sync\_exchange} - \varphi_2) \times (P_{idle} \times M_{sync}/R_{data\_rate}) \quad (59)$$

With

$$N = N_{total} - N_{data} - N_{sync\_neighbor\_discovery} + \varphi_1 \quad (60)$$

And,

$$N_{sleep_idle} = duty\_cycle \times P_{idle} + (1 - duty\_cycle) \times P_{sleep} \quad (61)$$

$$\varphi_1 \in [0, \min(N_{data}, N_{sync\_neighbor\_discovery})] \quad (62)$$

$$\varphi_2 \in [0, \min(N_{data}, N_{sync\_exchange})] \quad (63)$$

Based on $\varphi_1$ and $\varphi_2$'s range, we could derive the upper bound and lower bound of the sum of $E_empty$ and $E_{sync\_miss}$. The distribution of $\varphi_1$ and $\varphi_2$ can be assumed by a Binomial distribution with probability of 0.5, and with a mean value of:

$$\varphi_1 = 0.5 \times \min(N_{data}, N_{sync\_neighbor\_discovery}) \tag{64}$$

$$\varphi_2 = 0.5 \times \min(N_{data}, N_{sync\_exchange}) \tag{65}$$

And finally, taking all intermediate results into consideration, the overall energy consumption of the network can be summed up as follows:

$$E_{total} = E_{routing\_normal} + E_{routing\_abnormal} + E_{sync} + E_{empty} + E_{sync\_miss} \tag{66}$$

## 6 Energy analysis

The simulation environment we use is the NS-2 simulator, version 2.31 [9]. Our simulation model combines S-MAC and the one-phase-pull algorithm in Directed Diffusion. In the S-MAC protocol, we enable the adaptive listening and global schedule functionalities. The default duty cycle is set at 10 percent, and the data rate is 2 Mbps for the message sizes we proposed in Sect. 5, the S-MAC frame length will be 1.31 seconds with 10% duty cycle. Error encoding ratio is set at 2, as specified by the default setting in S-MAC. The data message size is 136 bytes and interest size is 96 bytes. We set the interest refresh time as 300 seconds and changed the ping application to report normal data once in 600 seconds, the event generation time is randomly selected. For every abnormal event, it generates 6 abnormal messages repeatedly within 10 s.

We use the topology of randomly distributed nodes in an area of 100 m × 70 m. One of these nodes is specified as the sink node. The simulation duration is 600 seconds. All the other parameter values are described in Table 3.

We change the network density and compare the energy consumption for both the PSE and FAE models. There is a linear relationship between the density and energy consumption (Fig. 4). The analytic result for both PSE and FAE models is remarkably similar to the simulation results, for density below 45. The small deviation in the energy consumption of the two scenarios illustrates that the synchronization cost is high when S-MAC is used. There are two reasons for this: (1) S-MAC repeatedly uses SYNC packets to synchronize the local timer and discover new neighbors during the entire lifetime; and (2) A node relentlessly attempts to send out a broadcast SYNC packet even if it loses a contention. For a high density networks, efficient packet transmission can be achieved by tuning parameters such as the event generation interval and the interest propagation duration.

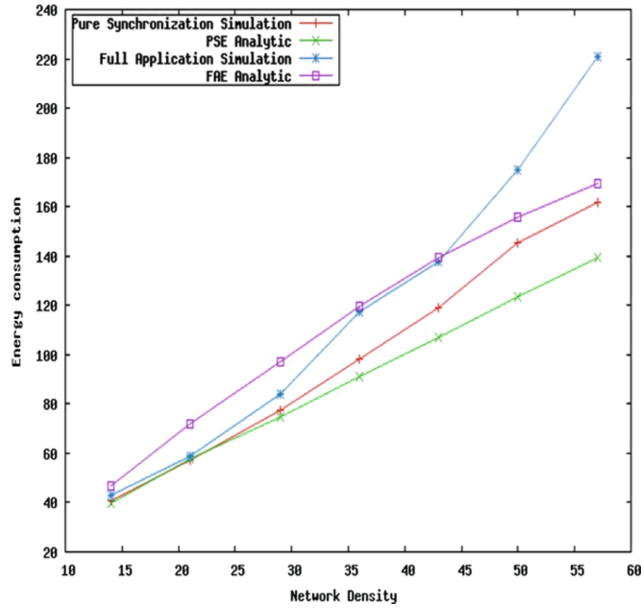


**Fig. 4** Energy consumption increases when network density becomes higher

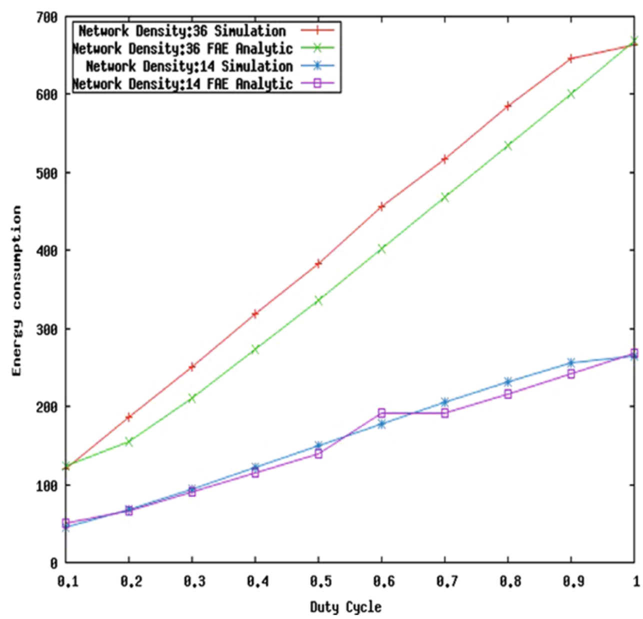


**Fig. 5** Energy consumption as duty cycle changes

### 6.1 Model validation and duty cycle

Figure 5 shows how energy consumption can be affected by the duty cycle of the MAC protocol. We varied the duty cycle of two different network densities: 14 and 35. The analytic results of the FAE model are similar to the simulation results with the deviation of less than 10% for both densities. The energy consumption increases due to the additional active time as well as collision and synchronization overhead. Because in S-MAC the listen time is fixed, when duty cycle

**Table 3** Simulation parameter list

| Basic parameter | Default value |
|---|---|
| Control message RTS/CTS/ACK | 10 bytes |
| SYNC message | 9 bytes |
| Interest message | 96 bytes |
| Data message | 136 bytes |
| Interest propagation frequency | 300 seconds |
| Normal event report interval | 300 seconds |
| Abnormal event report interval | 10 seconds |
| Abnormal event report period | 60 seconds |
| Abnormal event occurrence ratio | 1% |
| Duty cycle | 10% |
| Bandwidth | 2 Mbps |
| Network density | λ |
| Minimum hop counts | Topology dependent |
| S-MAC Frame length | Message size, duty cycle and Backoff Window |
| Adaptation time | Frame length dependent |
| Max retry times | 5 |
| Frequency of neighbor Discovery | 22 |
| Synchronization period | 10 |
| Nominal transmission Range | 40 m |
| Sensing field | 7000 $m^2$ (70 m × 100 m) |
| Transmission power | 31.2 mW |
| Receive/idle power | 22.2 mW |
| Radio@sleep status | 3 μW |

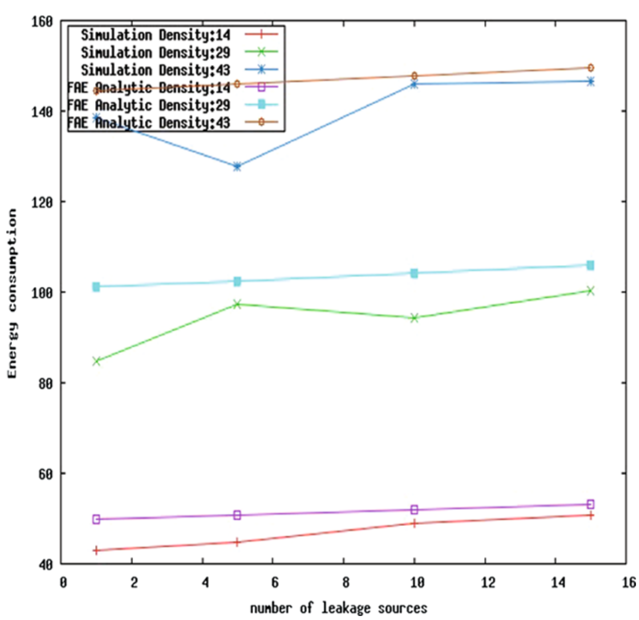


**Fig. 6** Energy consumptions varies along with the leakage sources

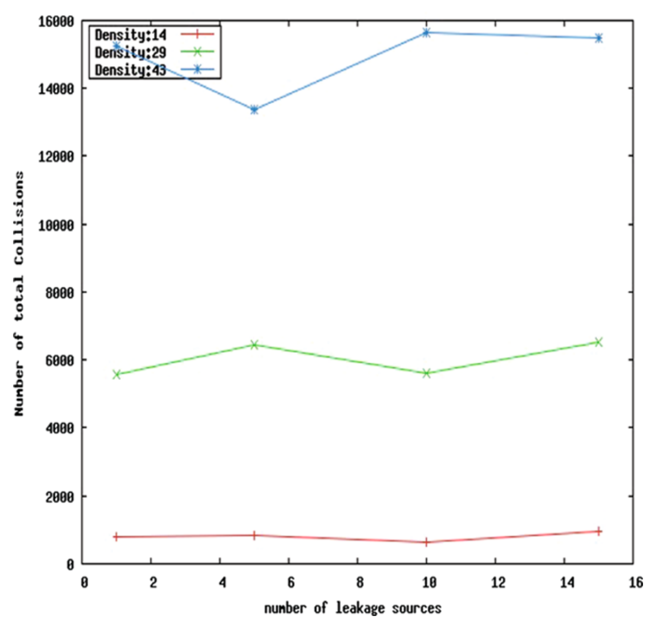


**Fig. 7** Collision varies as number of leakage sources increases

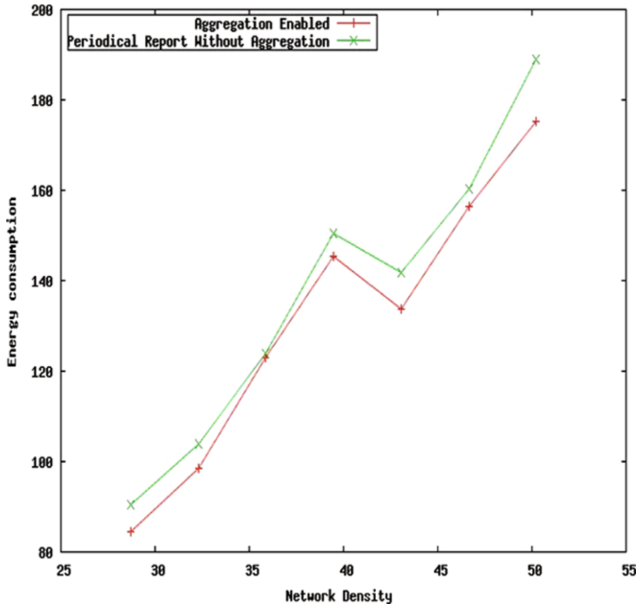


**Fig. 8** Data aggregation impact on energy consumption

varies, the frame length will vary adversely. Therefore when the duty cycle increases, the whole time will be divided into more frames, which will result in SYNC packet overhead increase. This in turn affects SYNC packets broadcasting interval and the neighbor discovery, both of which are frame size dependent.

Figure 6 denotes the relationship between energy consumption and the number of leakage sources. We simulated with three different densities: 14, 29, 43. When the leakage source increases from 1 to 15 with an increasing step of 5, the energy consumption rises in steps, but there is anomalous reduction in the simulation curves. The anomalous reduction becomes more obvious when network density increases. When abnormal events dominate data transmission for a certain period of time, the synchronization as well as neighbor discovery will be delayed, and SYNC packet collisions will be reduced temporarily and eventually results in a transient energy decrease. The relationship between collision and the number of leakage sources is shown in Fig. 7. In Fig. 6 and Fig. 7, these curves reveal similar behavior. In Fig. 6, though our analytical result approaches the simulation result, the analytical energy consumption raises only slightly in a liner fashion, without any anomalous point. This is because energy consumption in the analytical model is more ideally calculated. Though it considers the collision possibility in a statistical way, the collisions with other network behaviors such as synchronization and message queuing were difficult combine.

One way of reducing the data traffic in the network is by forwarding a report from a node only if the maximum $H_2S$ and $NH_3$ concentrations it reports is greater than all the other nodes in its neighborhood. This requires data aggregation, but it reduces the traffic in the entire network significantly. Figure 8 depicts the considerable energy saving under all network densities.

## 7 Conclusions

Both in the analysis and simulation case, as the density of the network increases, the energy utilization of the network increases also. One reason for this is that in a large density networks, the power consumption of each node at the link layer is significantly high due to collision. S-MAC begins applying the sleep schedule for each node only once the nodes have exchanged their schedule. Synchronization claims a significant amount of energy. The disproportional energy distribution even during normal sensing makes S-MAC unsuitable for toxic gas detection. Moreover, during simulation, we have observed that S-MAC's performance deteriorates considerably when the number of nodes in the sensing field exceeded 40. The Bianchi model for computing the energy cost during contention assumes saturation traffic, in which all the nodes have data to send at all times. While this is plausible for normal, periodic reports, it is unsuitable for irregular and bursty traffics. The energy cost of normal and abnormal events propagation decreases exponentially as the interest propagation interval increases. Interest has to be disseminated in the network to update routing paths and to define a new sensing task. Interest dissemination prompts gradient computation and reinforcement. The longer the interval, the lower the energy cost. On the other hand, choosing a long interest propagation interval implies a potential increase in latency of event propagation, since old paths might be broken for a number of reasons, as such is the case when some nodes exhaust their energy more quickly than others. There is a trade-off between latency and energy cost.

In our energy model, we have not considered the energy required for local signal processing, such as the energy consumed by the analog-to-digital (ADC) converter to produce a high resolution senor data. In reality, however, the ADC consumes a significant amount of power. In the future, we will accommodate this fact to assess the feasibility of using existing off-the-shelf hardware for building wireless sensor networks.

## References

1. Ghosh, B. (1951). Random distances within a rectangle and between two rectangles. *Bulletin of the Calcutta Mathematical Society*, *4*.
2. Boukerche, A., Fei, X., & Araujo, R. B. (2006). An energy-efficient sensing coverage protocol for surveillance and monitoring applications using wireless sensors. In *Performance, computing, and communications conference, 2006. IPCCC 2006. 25th IEEE international* (pp. 611–616). April 2006.

3. Bettstetter, C. (2002). On the connectivity of wireless multihop networks with homogeneous and inhomogeneous range assignment. In *IEEE vehicular technology conference, VTC 2002* (pp. 1706–1710).
4. Chao, X., Dargie, W., & Lin, G. (2008). Energy model for h2s monitoring wireless sensor network. In *CSE '08: proceedings of the 2008 11th IEEE international conference on computational science and engineering* (pp. 402–409). Washington: IEEE Computer Society.
5. Dargie, W., Schill, A., Mochaourab, R., & Guan, L. (2009). A topology control protocol for 2d Poisson distributed wireless sensor networks. In *The third international workshop on telecommunication networking, applications and systems*.
6. Feeney, L. M. (2001). An energy consumption model for performance analysis of routing protocols for mobile ad hoc networks. *Mobile Networks and Applications*, *6*(3), 239–249.
7. Bianchi, G. (2000). Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE Selected Areas in Communications*, *18*, 535–547.
8. Intanagonwiwat, C., Govindan, R., Estrin, D., Heidemann, J., & Silva, F. (2003). Directed diffusion for wireless sensor networking. *IEEE/ACM Transactions on Networking*, *11*(1), 2–16.
9. Issariyakul, T., & Hossain, E. (2008). Signal & communication. In *Introduction to network simulator NS2*. Berlin: Springer.
10. Kim, S., Pakzad, S., Culler, D., Demmel, J., Fenves, G., Glaser, S., & Turon, M. (2007). Health monitoring of civil infrastructures using wireless sensor networks. In *IPSN '07: proceedings of the 6th international conference on information processing in sensor networks* (pp. 254–263). New York: ACM.
11. Lee, S.-J., Belding-Royer, E. M., & Perkins, C. E. (2003). Scalability study of the ad hoc on-demand distance vector routing protocol. *International Journal of Network Management*, *13*(2), 97–114.
12. Mainwaring, A., Culler, D., Polastre, J., Szewczyk, R., & Anderson, J. (2002). Wireless sensor networks for habitat monitoring. In *ACM international workshop on wireless sensor networks and applications (WSNA 2002)* (pp. 88–97).
13. Nasipuri, A., Casta, N. R., & Das, S. R. (2001). Performance of multipath routing for on-demand protocols in mobile ad hoc networks. *Mobile Networks and Applications*, *6*(4), 339–349.
14. Spivak, M. (2006). *Calculus* (3rd ed.). Cambridge: Cambridge University Press.
15. Stoianov, I., Nachman, L., Madden, S., & Tokmouline, T. (2007). Pipenet: a wireless sensor network for pipeline monitoring. In *IPSN '07: Proceedings of the 6th international conference on information processing in sensor networks* (pp. 264–273). New York: ACM.
16. Wei Tseng, H., Yang, S.-H., Chuang, P.-Y., Wu, H.-K., & Chen, G.-H. (2004). An energy consumption analytic model for a wireless sensor mac protocol. In *Vehicular technology conference* (pp. 4533–4537).
17. Werner-Allen, G., Lorincz, K., Welsh, M., Marcillo, O., Johnson, J., Ruiz, M., & Lees, J. (2006). Deploying a wireless sensor network on an active volcano. *IEEE Internet Computing*, *10*(2), 18–25.
18. Ye, W., Heidemann, J., & Estrin, D. (2002). An energy-efficient mac protocol for wireless sensor networks. In *Infocom* (pp. 1567–1576).
19. Zhong, L. C. (2004). *A unified data-link energy model for wireless sensor networks*. PhD thesis, Chair-Jan M. Rabaey.
20. Zimmerling, M., Dargie, W., & Reason, J. (2007). Energy-efficient routing in linear wireless sensor networks. In *The fourth IEEE international conference on mobile ad-hoc and sensor systems*.
21. Zimmerling, M., Dargie, W., & Reason, J. M. (2008). Localized power-aware routing in linear wireless sensor networks. In *CASEMANS '08: Proceedings of the 2nd ACM international conference on context-awareness for self-managing systems* (pp. 24–33). New York: ACM.

**Waltenegus Dargie** obtained a Ph.D. in Computer Engineering from the Technical University of Dresden, Germany (2006), a M.Sc. degree in Electrical Engineering from the Technical University of Kaiserslautern, Germany (2002) and a B.Sc. in Electrical and Electronics Technology from the Nazareth Technical College, Ethiopia (1997). Prior to his current position, he has been working as a researcher at the University of Kassel, Germany (2002–2005) and at the Fraunhofer Institute of Experimental Software Engineering, Germany (2002–2003). His research interests include digital signal processing, wireless and mobile networks, wireless sensor networks, and pervasive computing. Presently, Dr. Dargie servers as a Guest Editor to the International Journal of Autonomous and Adaptive Communication Systems (IJAACS) and to the Journal of Computer and System Science (JCSS). He is also a chair or co-chair of a number of IEEE and ACM workshops, including the CASEMANS 2009 (ACM) and PMECT 2009 (IEEE). Dr. Dargie is the member of IEEE.

**Xiaojuan Chao** Xiaojuan Chao is a graduate of the Technical University of Dresden, from which she obtained a M.Sc. degree in Computational Engineering in 2007. Her research interests include distributed systems, web software architecture, wireless communications and wireless sensor networks. She is now focusing on Internet technologies, Project Management and Architecture. Chao has more than 9 years of experience in software design and system architecture.

**Mieso K. Denko** received his M.Sc. degree from the University of Wales, UK, and his Ph.D. degree from the University of Natal, South Africa, both in Computer Science. Currently, he is with the Department of Computing and Information Science, University of Guelph, Guelph, Ontario, Canada. His current research interests include wireless networks, mobile and pervasive computing, wireless mesh networks, body sensor networks and network security. Dr. Denko is a founder/co-founder of a number of ongoing international workshops and served as program chair/co-chair of a number of IEEE/ACM international conferences. Currently he is serving as guest co-editor of Special Issues for a number of journals including the ACM/Springer Mobile Networks and Applications (MONET) and IEEE Systems Journal. Dr. Denko has co-edited two books in the areas of pervasive computing and wireless networking, and currently co-editing two forthcoming books, autonomic computing and networking with Springer and Pervasive Computing and Networking with Wiley. He is editorial board member of international journals including, the International Journal of Smart Homes (IJSH), the Journal of Ubiquitous Computing and Communications, (UBICC), and Associate Editor of the International Journal of Communication Systems (IJCS), Wiley, Security & Communications Network (SCN), Wiley and the Journal of Ambient Intelligence and Humanized Computing, Springer. He is a senior member of the ACM and IEEE.