SENSOR LETTERS
Vol. 3, 1–10, 2005

Modeling the Lifetime of Wireless Sensor Networks

Kewei Sha* and Weisong Shi

Department of Computer Science, Wayne State University, 5143 Cass Avenue, Detroit, MI 48202, USA

(Received: 23 February 2005. Accepted: 10 April 2005)

We propose a novel model to formally define the lifetime of a wireless sensor network based on energy by considering the relationship between individual sensors and the whole sensor network, the importance of different sensors based on their positions, the link quality, and the connectivity and coverage of the sensor network. Using the proposed model, we have compared two types of query protocols, *the direct query protocol* and *the indirect query protocol*, in terms of both mathematical analysis and comprehensive simulation. The simulation results validate the correctness of the mathematical analysis and the effectiveness of the proposed lifetime model.

Keywords: Lifetime, Wireless Sensor Networks, Energy.

RESEARCH ARTICLE

1. INTRODUCTION

The advance in the wireless communication technology makes it possible to develop more powerful micro-sensors with a lower price.[1–6] The commercialization of micro-sensors with wireless communication, such as Motes from Crossbow[7] and Intel, enables interesting applications where current wired sensor technology would fail, such as unattended environment monitoring,[8] habitat monitoring,[9] military surveillance, and others.

Most wireless sensors are battery-backed and are small. These physical constraints of sensors and the prohibitive costs to replace the failure sensors in the sensor network make energy a crucial consideration to design a long lifetime sensor network. How to extend the lifetime of sensor network has been a topic of considerable interest in the research field of sensor network. Many efforts have been made to achieve this goal by using energy efficient protocols such as Refs. [10–13]. The rationale behind these work includes taking paths with maximum available power, minimum energy, minimum hop, maximum minimum available power; however, none of them has a detailed analysis of the lifetime of sensors, and the relationship between the lifetime of individual sensor and that of the whole sensor network.

Currently, the lifetime of a sensor network is defined as the time for the first node as in Refs. [14–17] or a certain percentage of network nodes as in Refs. [18, 19] to run out of power. The former is too pessimistic since when one node fails the rest nodes still can provide appropriate functionality. While the latter does not consider the different importance of sensors in the sensor network. We argue that lifetime is an application-specific concept. To this end, we propose a novel lifetime definition based on energy by considering the relationship between the lifetime of a single sensor and that of the whole sensor network, the importance of sensors at different positions, the link quality in the communication, and the connectivity and coverage of the sensor network. Based on the proposed lifetime model, we evaluate two query protocols in the context of abstracted communication pattern. We also conduct a detailed simulation to validate the correctness of our lifetime model.

Our model of lifetime of wireless sensor networks can be used by both sensor system and protocol designers as well as sensor network practitioners (application scientists). From the perspective of sensor system designers, our model can be used as a metrics to evaluate the efficiency, effectiveness, and performance of the designed protocols and algorithms similar to how we evaluate the performance of the two query protocols. While from the view point of the application scientists, our model can be used as an indication of status of the operating sensor network.

The main contributions of this paper lie in the following aspects. First, we provide a set of formal models characterizing the remaining lifetime of each sensor, the remaining

*Corresponding author; E-mail: kewei@wayne.edu

 1546-198X/2005/3/001/010/$17.00+.25 doi:10.1166/sl.2005.017

lifetime of the whole sensor network, and the lifetime of the sensor network. To our knowledge, that is the first effort to give a general and formal analysis of the lifetime of sensor networks. Second, we compare two query protocols in terms of the proposed models by mathematical analysis. Finally, we conduct a comprehensive performance evaluation to validate the analysis results based on our model.

The rest of this paper is organized as follows. Section 2 presents the formal model of the remaining lifetime and the lifetime of the whole sensor network. An application of the lifetime model that theoretically analyzes and compares two query protocols in terms of our model is depicted in Section 3. Simulation to validate the correctness of the model is reported in Section 4. Finally, related work and conclusion are listed in Section 5 and Section 6 respectively.

2. THE LIFETIME MODEL

The lifetime of a wireless sensor network (denoted as LSN) is an application-specific, flexible concept. However, we can abstract and define a remaining lifetime of a wireless sensor network (denoted as RLSN) first, which is defined as the weighted sum of the remaining lifetime of individual sensors (denoted as RLIS) of all the sensors in the sensor network. Given that, we can define the LSN for three major application categories: *active query*, *event-driven*, and *passive monitoring*.[20]

In an active query like application, the LSN can be defined as the maximum number of queries the sensor network can handle before the sensor network terminates. For an event-driven application, the LSN can be defined as total number of events the sensor network can process before the termination of it. For passive monitoring, the LSN can be defined as the total amount of time slots before termination. The termination of the sensor network is defined as the time slot when the RLSN starts to keep stable that implies that the sensor network loses connectivity or the coverage of the sensor network below a predefined threshold, θ, which means that the sensor network becomes useless. Here, we assume the energy consumption of regular maintenance overhead is negligible, and will be considered later. Next, we in turn describe the detailed model in our proposal.

2.1. Assumptions and Definitions of Parameters

Several assumptions in our model are listed here. All the symbols used in this analysis are listed in Table I as well.

- All the sensors are homogeneous, i.e., the physical capacity like communication range of each sensor is same.
- Location information is available by physical devices such as GPS or topology discovery algorithms.[21–23]
- The location of each sensor is stationary. The data sink is fixed, which is usually true in the real deployment.
- Sensors can only communicate with the peers within communication range due to limited power. Multi-hop is required to communicate with farther ones.

2.2. Definition of RLIS

The remaining lifetime of individual sensor is defined as the normalized remaining energy of the sensor at moment N_m. In the processing of the sensor network, the energy is consumed when the sensor receives or sends a message. Because of the unstable wireless communication in WSN, the package may be retransmitted to guarantee the correct delivery. Thus, we also take link quality and lossy rate into consideration when we calculate the total energy consumption for message transmission. So the RLIS is ratio of the remaining energy to the initial energy, which can be defined as

$$L(j) = \frac{E_j - \sum_{i=1}^{N_m} \epsilon_{jiq} * N_{jiq} * Rt_{jiq} + \epsilon_{jir} * N_{jir} * Rt_{jir}}{E_j}$$
$$= 1 - \frac{\sum_{i=1}^{N_m} \epsilon_{jiq} * N_{jiq} * Rt_{jiq} + \epsilon_{jir} * N_{jir} * Rt_{jir}}{E_j}$$

We borrow the same energy model and symbols used in Ref. [15] to calculate energy consumption of each message transmission. The energy consumed when the sensor receives a message of size k is

$$\epsilon_{rcv} = \epsilon_{elec} * k$$

and the energy consumed on sending a message of size k is

$$\epsilon_{snd} = \epsilon_{elec} * k + \epsilon_{amp} * r^2 * k$$

So, we have

$$\epsilon_{jiq} = \epsilon_{jiq,\,rcv} + \epsilon_{jiq,\,snd} \quad \text{and} \quad \epsilon_{jir} = \epsilon_{jir,\,rcv} + \epsilon_{jir,\,snd}$$

To calculate the $L(j)$, we should calculate N_{jiq} and N_{jir} first, which is related to P_{ji}, the probability of the message go through the jth sensor at the ith moment.

Figure 1 shows the message propagation in the sensor network from a macro view. As we can observe from the figure, the value of P_{ji} is related to the distance between the sensor and the sink. The query messages directed to the sensors far away from the sink and reply messages from far away sensors to the sink will both go through some sensors nearer to the sink than these sensors. While the messages from or to the sensors near to the sink will not go through the farther ones. Thus if all sensors have the same probability to be the query destination, the probability that the message go through the nearby sensors of the sink is larger than that of going through the far away sensors to the sink. As shown in Figure 1, a message is

Table I. A list of variables used in this paper.

Variables	Description
N_m	The moment, which is the number of queries in active query application, the number of events in the event-driven application and the number of time slots in the passive monitoring
ϵ_{jiq}	In the ith moment the amount of energy consumed if one query message goes through the jth sensor, specifically $\epsilon_{jiq,rcv}$ for receiving and $\epsilon_{jiq,snd}$ for sending. When we assume all the query messages are same, ϵ_{jiq} can be reduced to ϵ_q
ϵ_{jir}	In the ith moment the amount of energy consumed if one reply message goes through the jth sensor, specifically $\epsilon_{jir,rcv}$ for receiving and $\epsilon_{jir,snd}$ for sending. When we assume all the reply messages are same, ϵ_{jir} can be reduced to ϵ_r
E_j	The total initial energy of jth sensor. When we assume all the sensors to be homogeneous, E_j is equal to E_0.
$1/f$	An application-specific parameter to determine the possibility of a sensor generating a reply to a query
N_{jiq}	The number of the query messages that go through the jth sensor at the ith moment
N_{jir}	The number of the reply messages that go through the jth sensor at the ith moment
r	The communication range of each sensor
S_q,S_r	The size of the query message and reply message separately
P_{jiq}	The probability that the query messages goes through the jth sensor node at the ith moment
P_{jir}	The probability that the reply messages goes through the jth sensor node at the ith moment
P_{ji}	The probability that a message will go through the jth sensor node at the ith moment
$P(B)_{ji}$	The probability that the sensor farther than the jth sensor to the sink is the destination of the query or has reply to sink at the ith moment
$P(A)_{ji}$	The probability that the message going through the jth sensor at the ith moment when that message goes through the circle area where the jth sensor node is located
N_{far}	The number of the sensor nodes that are farther than the jth sensor node to the sink
N_{nbrs}	The number of the sensor nodes in the communication range
N_{total}, N_n	The number of the total sensor nodes in the sensor network
θ	The maximum number of depleted sensors when the sensor network's functionality is correct
ρ	The density of the sensor nodes in the sensor network
d_{jis}	The distance from the jth sensor node to the sink (or delegate) at the ith moment
w_j	The weight (importance) of the jth sensor node in the sensor network
$d_{\max}$	The ratio of the maximum distance between every two sensor nodes in the sensor network to the communication range r
$\mathscr{L}$	The remaining lifetime of the whole sensor network
$L(j)$	The remaining lifetime of the jth sensor
LFT	The lifetime of the whole sensor network
Rt_{jiq}	The number of retransmission times for query message going through the jth sensor at moment i
Rt_{jir}	The number of retransmission times for reply message going through the jth sensor at moment i
Rt	The number of retransmission times
Ls_{jiq}	The lose rate of query message of jth sensor at the ith moment
Ls_{jir}	The lose rate of reply message of jth sensor at the ith moment
Ls_{avg}	The average lose rate of the sensor network
$COV(i)$	The coverage of the sensor network at moment i

routed from the sink to the jth sensor step by step, at each step the message must be in one circle area. If there are n sensors in one circle area, the probability that the message is hold by one specified sensor in that circle area is $1/n$. Let A denote the event that the message goes through the jth sensor when it goes through the circle area the node locates, and B be the event that the destination of at moment i is farther to the sink than the jth node. Let $P(B)_{ji}$ be the probability of B, and $P(A)_{ji}$ be the probability of A. Because A and B are two independent events, the probability that the message will go through the jth node at the ith moment is

$$P_{ji} = P(A|B)_{ji} = P(A)_{ji}P(B)_{ji}$$

Here,

$$P(B)_{ji} = \frac{N_{\text{far}}}{N_{\text{total}}} = \frac{N_n - \pi d_{jis}^2 \rho}{N_n}$$

and

$$P(A)_{ji} = \frac{1}{\pi\Big(\big(\big(\lfloor \frac{d_{jis}}{r} \rfloor + 1\big) * r\big)^2 - \big(\lfloor \frac{d_{jis}}{r} \rfloor * r\big)^2\Big)\rho}$$
$$= \frac{1}{\pi\big(2\lfloor \frac{d_{jis}}{r} \rfloor + 1\big) r^2 \rho}$$

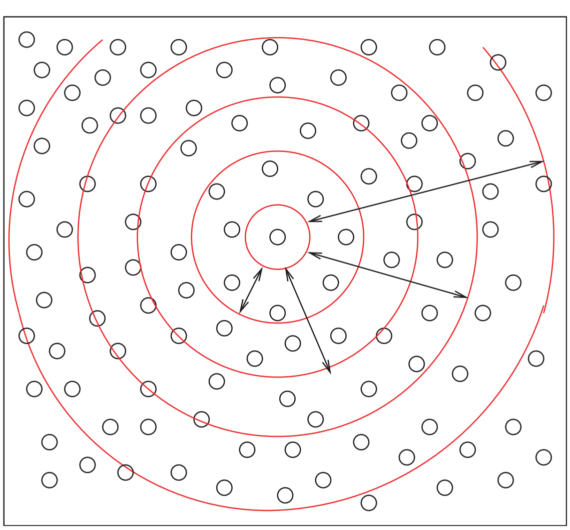


Fig. 1. Message propagation in sensor networks.

RESEARCH ARTICLE

Thus

$$P_{ji} = \frac{1}{\pi\left(2\lfloor\frac{d_{jis}}{r}\rfloor + 1\right)r^2\rho} \frac{N_n - \pi d_{jis}^2\rho}{N_n}$$

Furthermore, link quality in wireless sensor networks as advocated in Refs. [24–26] is of great importance to high level protocols. Therefore, we should also take this into consideration in our model. Here we calculate the number of the message retransmission times. Assume the lossy rate is p, then the number of message retransmission times is

$$\begin{aligned} Rt &= 1 \times p + 2p(1-p) + 3p^2(1-p) + \cdots \\ &= 1 + p + p^3 + \cdots = \frac{1}{1-p} \end{aligned}$$

Give the probability P_{ij} and the value of retransmission times, we can calculate the RLIS as:

$$L(j) = 1 - \sum_{i=1}^{N_m} \frac{\frac{\epsilon_{jiq} * N_{jiq}}{1-Lr_{jiq}} + \frac{\epsilon_{jir} * N_{jir}}{1-Lr_{jir}}}{E_j}$$

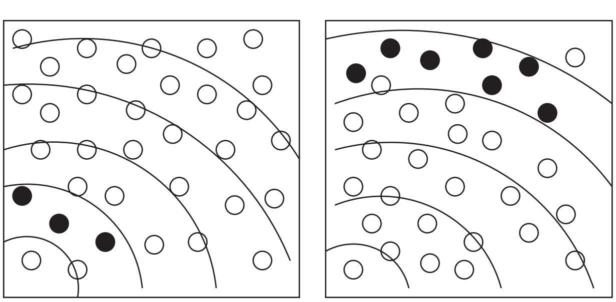
Based on RLIS we can model the RLSN, the weighted sum of RLIS of all sensors. In the following section, we will define the weight of each sensor first, which is defined based on the location of the sensor in the sensor network.

2.3. Importance of Different Sensors

The failure, which is resulted from the deplete of energy, of sensors will cause the sensor network to act improperly, but the level of the damage it causes is different, which is the reason why we think the previous definition of the termination of the lifetime without considering the location of the failed sensor is unsatisfactory. For the same number of failure sensors, the damage may be very slight at sometime and the sensor network still performs almost normal, while sometimes it may be very serious and makes the sensor network lose its most functionality. Two cases are described in the Figure 2(a) and (b) as an example. In the figure, the black nodes represent the sensors that have run out of energy and the white ones denote the ones that are still alive. In both Figure 2(a) and (b) the sensor networks cannot act as it suppose to do since in both cases the sensor network cannot gain data from some sensors, but in Figure 2(a), although there are only three failed sensors, the sink cannot get data from most of the sensors. And in Figure 2(b), there are seven dead nodes, but the sink can still get data from most of the sensors in the sensor network. So the damage to the sensor network by the failure sensors is not only related with the number of failed sensors but also related to the location of the failed sensors. To this end, sensors in the sensor network have different importance. We define a factor named *weight* for each sensor to count the importance of that sensor. Based on above analysis, the nearer the sensor to the sink, the more important it is. So we define the weight of each sensor as following:

$$w_j = c\frac{1}{d_{jis}^2}$$

Where c is a constant.



(a) The sensors near the sink are dead (b) the sensors far from the sink are dead

Fig. 2. An example of the importance of different sensors, assuming the data sink is located at the low-left corner.

2.4. Definition of RLSN

Having the RLIS and the importance of each sensor, we are in a position to examine the remaining lifetime of the whole sensor network. We consider *the remaining lifetime of the whole sensor network* as the sum of the weighted remaining lifetime of all sensors in the sensor network. Thus the remaining lifetime of the whole sensor network is

$$\mathscr{L} = \sum_{j=1}^{N_n} w_j L(j)$$

2.5. Definition of Lifetime of Sensor Network

Based on the remaining lifetime of the whole sensor network, the connection between the lifetime of individual sensors and that of the whole sensor network, taking the connectivity and coverage of the sensor network into consideration, the lifetime of the sensor network can be formally defined as:

$$LFT = \min i;$$

$$i = \begin{cases} \mathscr{L}(i-1) > \mathscr{L}(i) \ \& \ \mathscr{L}(i+1) = \mathscr{L}(i) & \text{Connectivity} \\ COV(i-1) > \theta \ \& \ COV(i) \le \theta & \text{Coverage} \end{cases}$$

In the above formula, the LSN is defined as the moment that either the sensor network loses connectivity as depicted in the first line or loses coverage to a certain threshold as indicated in the second line in the definition of i, where θ is a pre-defined threshold of the coverage of the sensor network based on the application requirements. For example, in some application, if coverage of the sensor network is less than 10%, the data gathered by the sensor network is regarded as useless. This definition reflects the two points raised in Section 1, which denotes that our lifetime model can be used by both sensor system designers and application scientists. For example, protocol designers can use the notion of lifetime to evaluate the effectiveness and energy efficiency of their designs, while sensor network practitioners will benefit from this high level metrics to monitor the operation of the sensor network, especially when this notion has been integrated into a GUI-based interface, such as QueryAgent.[20]

2.6. Discussion

Our lifetime model is a novel definition, which is formalized from the view of energy, the most important consideration in the design of the sensor network. In the lifetime model, we combine several significant factors such as the relationship between the individual sensor and the whole sensor network, the importance of the sensor based on their location, the link quality of the wireless communication, the connectivity and coverage of the sensor network. Considering the real usage of the proposed lifetime model, we can have either centralized or localized algorithm to calculate the value of LSN. In the centralized approach, we assume that there is a program running at the sink, which collects the corresponding information from the sensor network periodically, so the powerful sink can easily detect the coverage of the sensor network and calculate the RLSN based on the formula to make a decision on the time when the lifetime of the sensor network is over. In the real implementation, we do not want this operation consumes extra energy. Thus, this piece of information can be piggybacked by regular data messages. On the other hand, it is also easy to develop a decentralized algorithm to calculate the lifetime because the RLIS and the link quality are properties of the individual sensors, which is always available to each sensor, the importance of a sensor is decided by the location of the sensor, which is easy to get by proposed localization approaches like in Refs. [21–23], the local coverage is easy to compute by using the approach described in Ref. [27]. After calculating the RLIS of each individual sensor, it will send it back (using aggregation) to the sink so that the LSN of the sensor network can be calculated. In our current Capricorn simulator,[28] we have implemented both algorithms.

Having these formal models, in the following section, we will use these models to mathematically analyze two query protocol, *the indirect query protocol* proposed in Ref. [29] and *the traditional query protocol*.

3. APPLICATION OF THE LIFETIME MODEL

Our model aims to be used to evaluate the performance of the proposed protocols. In this section, we use it as an example to show how to apply our model to evaluate two different query protocols. First we will describe these two query protocols, then we will build and compare the lifetime model for the two protocols.

3.1. Two Different Query Protocols

In this section, we briefly describe the two query protocols, the indirect query protocol (denoted as *IQ*) in Ref. [29] and the direct query protocol (denoted as *Traditional*). Previously, query is processed using *Traditional*, where queries are always routed from the data sink to its destination (one or more sensors) using an energy efficient path or an alternative path based on other performance metrics. While the *IQ* protocol consists of three steps. First, the data sink randomly selects a sensor as the query delegate and forwards the query to the delegate. Second, the delegate gets the query and conducts the query processing on behalf of the data sink, and then aggregates the replies. Third, the delegate sends the aggregated reply back to the data sink. Comparing with *Traditional*, two extra steps, *query forwarding* and *query replying*, are added in the *IQ* protocol, but compared with *Traditional*, *IQ* can balance the load to the whole sensor thus to extend the lifetime of the sensor network. From observation, we can abstract the communication pattern to four typical types, unicast, area-multicast, area-anycast and broadcast. As argued in Ref. [29], *IQ* performs the same as *Traditional* in the point to point routing, but it performs perfect for other types. Additionally, because we can look area-multicast as a combination of unicast and broadcast, so in this paper we will just compare these two query protocols in the context of the broadcast communication pattern.

3.2. Lifetime Model for *Traditional*

Based on the model we proposed in the previous section, we deduce the lifetime model of in *Traditional* here. In *Traditional* using broadcast communication, the query floods to all the sensor while the reply will go through only the sensor on its path to the sink. For each query if we assume the probability that one sensor will generate a reply is $1/f$, RLIS using *Traditional* is

$$L(j) = 1 - \sum_{i=1}^{N_m} \frac{\frac{\epsilon_{jiq} * N_{jiq}}{1-Lr_{jiq}} + \frac{\epsilon_{jir} * N_{jir}}{1-Lr_{jir}}}{E_j}$$

$$= 1 - \sum_{i=1}^{N_m} \frac{\frac{\epsilon_{jiq} * N_{nbrs}}{1-Lr_{jiq}} + \epsilon_{jir} P_{jir} N_n \frac{1}{f(1-Ls_{jir})}}{E_j}$$

Where N_{nbrs} is the number of the sensors within its communication range r, and Lr_{jiq} and Lr_{jir} are environment sensitive lost rates. To simplify the analysis, here we use Lr_{avg} which can be pre-calculated from Lr_{jiq} and Lr_{jir} to replace Lr_{jiq} and Lr_{jir}, since from the long term view the link quality should keep at one level. We also assume each sensor has the same initial energy, E_0. So RLIS in the broadcast traditional query is

$$L(j) = 1 - \frac{N_m \epsilon_q N_{nbrs}}{E_0(1-Lr_{\text{avg}})} - \frac{\epsilon_r N_m}{E_0(1-Lr_{\text{avg}}) f \pi r^2 \rho} \frac{N_n - \pi d_{jis}^2 \rho}{2\lfloor \frac{d_{jis}}{r} \rfloor + 1}$$

RESEARCH ARTICLE

Based on the model defined for RLIS, the RLSN in *Traditional* can be calculated as

$$\mathscr{L} = \sum_{j=1}^{N_n} \frac{c}{d_{jis}^2}\left(1 - \frac{N_m \epsilon_q N_{nbrs}}{E_0(1-Lr_{\text{avg}})} - \frac{\epsilon_r N_m}{E_0(1-Lr_{\text{avg}}) f \pi r^2 \rho} \frac{N_n - \pi d_{jis}^2 \rho}{2\lfloor \frac{d_{jis}}{r} \rfloor + 1}\right)$$

$$\approx c\pi\rho\left(2lnd_{\max} + \frac{\pi^2}{6}\right)\left(1 - \frac{\epsilon_q N_m N_{nbrs}}{E_0(1-Lr_{\text{avg}})}\right) - \frac{c\pi^2 \epsilon_r N_m N_n}{6E_0(1-Lr_{\text{avg}}) f r^2} + \frac{c\epsilon_r N_m \pi\rho d_{\max}}{E_0(1-Lr_{\text{avg}}) f}$$

3.3. Lifetime Model for *IQ*

In the indirect query, a query is directed to a randomly selected delegate, then the delegate acts as the sink to take care of query forwarding, data collection, and data transmitting back to the sink. If the probability of each sensor to be a delegate is the same, when there are totally N_m queries been processed, and there are totally N_n sensors in the sensor network, for each sensor the possible times it is selected as a delegate is N_m/N_n. The number of times that the sensor is located in the area with $d_{jis} = kr$ to the delegate is the number of the sensors located in the circle area between kr and $kr+1$ to that sensor which is $\pi(2k+1)r^2\rho$. Thus RLIS in *IQ* can be defined as

$$L(j) = 1 - \sum_{i=1}^{N_m} \frac{\frac{\epsilon_{jiq} * N_{jiq}}{1-Lr_{jiq}} + \frac{\epsilon_{jir} * N_{jir}}{1-Lr_{jir}}}{E_j}$$

$$= 1 - \frac{\epsilon_q N_m N_{nbrs}}{E_0(1-Lr_{\text{avg}})} - \frac{1}{E_0(1-Lr_{\text{avg}})}\frac{N_n}{f} * \sum_{i=1}^{N_m} \epsilon_r N_{jir} - \frac{N_m}{N_n}\frac{N_n}{E_0(1-Lr_{\text{avg}}) f}\epsilon_{r,rcv}$$

$$= 1 - \frac{\epsilon_q N_m N_{nbrs}}{E_0(1-Lr_{\text{avg}})} - \frac{1}{E_0(1-Lr_{\text{avg}})}\frac{N_n}{f} * \sum_{k=1}^{d_{\max}} \epsilon_r \frac{N_m}{N_n} P_{jis}(\pi\rho(((k+1)r)^2 - (kr)^2)) - \frac{N_m}{E_0(1-Lr_{\text{avg}}) f}\epsilon_{r,rcv}$$

$$= 1 - \frac{\epsilon_q N_m N_{nbrs}}{E_0(1-Lr_{\text{avg}})} - \frac{\epsilon_r N_m d_{\max}}{E_0(1-Lr_{\text{avg}}) f} + \frac{\epsilon_r N_m}{E_0(1-Lr_{\text{avg}}) f}\frac{\pi\rho r^2}{2N_n} d_{\max}(d_{\max}+1) - \frac{N_m}{E_0(1-Lr_{\text{avg}}) f}\epsilon_{r,rcv}$$

Similarly in *IQ*, the RLSN is defined as:

$$\mathscr{L} = \sum_{j=1}^{N_n} c\frac{1}{d_{jis}^2}\left(1 - \frac{\epsilon_q N_m N_{nbrs}}{E_0(1-Lr_{\text{avg}})} - \frac{\epsilon_r N_m d_{\max}}{E_0(1-Lr_{\text{avg}}) f} + \frac{\epsilon_r N_n}{E_0(1-Lr_{\text{avg}}) f}\frac{\pi\rho r^2}{2N_n} d_{\max}(d_{\max}+1) - \frac{N_m}{E_0(1-Lr_{\text{avg}}) f}\epsilon_{r,rcv}\right)$$

$$= c\pi\rho\left(1 - \frac{\epsilon_q N_m N_{nbrs}}{E_0(1-Lr_{\text{avg}})} - \frac{\epsilon_r N_m d_{\max}}{E_0(1-Lr_{\text{avg}}) f} + \frac{\epsilon_r N_m}{E_0(1-Lr_{\text{avg}}) f}\frac{\pi\rho r^2}{2N_n} d_{\max}(d_{\max}+1) - \frac{N_m}{E_0(1-Lr_{\text{avg}})}\epsilon_{r,rcv}\right)\left(2lnd_{\max} + \frac{\pi^2}{6}\right)$$

3.4. Comparison of *Traditional* and *IQ*

One of the goals of modeling is to evaluate the performance of different protocols. Now we are ready to compare *Traditional* with *IQ* in terms of RLIS and RLSN.

To quantitatively compare these two query protocols, we adopt the practical values of sensor parameters obtained from Berkeley motes,[30] including the initial energy and the energy consumption rate. In Ref. [30] two 1.5 V batteries rated at 575 mAh are used for each sensor, so the initial total energy of each sensor is 1.725 J. The energy to transmit and receive a single bit is 1 μJ and 0.5 μJ respectively. We assume the size of query message and reply message to be 240 bits and 1200 bits separately. Thus it takes 240 μJ to transmit a query message and 120 μJ to receive a query message; and it takes 1200 μJ and 600 μJ to transmit and receive a reply message. If we assume the total number of sensors is 1500 and the density of the sensor network is 1 per 1000 m^2, the maximum distance between every two sensors is $14r$, where r is the communication range equal to 50 m. We also assume the probability of one sensor generating a corresponding reply message is 1/30, thus $f = 30$.

First we compare the sensors located at different areas in the sensor network based on RLIS, which is depicted in Table II, in the context of two query protocols. Here we select the sensors with its distance to the sink as within one communication range, with seven times communication

Table II. Comparison of the remaining lifetime of different nodes in different locations.

Query types	*Traditional*	*IQ*
$0 < d < r$	$1 - \frac{10310N_m}{10^6}$	$1 - \frac{2647N_m}{10^6}$
$d = 7r$	$1 - \frac{2460N_m}{10^6}$	$1 - \frac{2647N_m}{10^6}$
$d = 14r$	$1 - \frac{2048N_m}{10^6}$	$1 - \frac{2647N_m}{10^6}$

range, and with 14 times communication range respectively. The last range is the $d_{\max}$ here. From the figures in the table, we can find that RLIS increases along with the increase of the distance between the sensor to the sink in *Traditional*, thus the sensors near to the sink will consume lots of energy and fail very quickly, which results in the earlier termination of LSN. From this observation, we find that the unbalanced load results in the short RLN. On the contrary, as we expected, RLIS of sensors located in different regions using *IQ* is almost same, which denotes that *IQ* indeed does a good job on balancing the load among all sensors.

We also compare RLSN here. The results of comparison of the two query protocol are listed in Table III. From these results, we can find that the sensor network using *IQ* has larger RLSN than that of using *Traditional* by providing a global optimization to balance the load to the whole sensor network. Furthermore, considering these two tables together, in *Traditional*, when $N_m = 121$, although there are still a large amount of energy ($13384/10^6$ from Table III) remaining in the sensor network, it will never be used because the sensor network is down when all sensors within the sink's communication range are down (see in Table II when $N_m = 121$). While in *IQ*, because the load is balanced, no sensors will run out of energy much earlier than others. So most energy of each sensor will be effectively used in *IQ*. To this end, we think that *IQ* is more energy efficient than *Traditional*, i.e, the energy utilization is much higher in *IQ* than that in *Traditional*.

From above analysis, we conclude that *IQ* extends LSN a lot because it balances the load to all the sensors in the sensor network, which again validates that the load balance plays a very important role in LSN.

4. PERFORMANCE EVALUATION

To verify the analytical results, we conduct a detailed simulation using the Capricorn,[28] a large-scale discrete-event driven simulator. In our simulation, 400 nodes are scattered to a 600 m × 600 m square field. We use the GPSR routing protocol implemented in routing layer of the simulator to deliver message. All simulation parameters are listed in Table IV. In this section, we evaluate *Traditional* and *IQ* in terms of energy consumption, RLSN and LSN.

4.1. Energy Consumption

First we compare energy consumption of each sensor in *Traditional* and *IQ*, which can reflect RLIS. Figure 3 reports the energy consumption of sensors in two protocols

Table III. Comparison of the remaining lifetime of the whole sensor network.

Query types	*Traditional*	*IQ*
Remaining lifetime	$\frac{21733-87N_m}{10^6}$	$\frac{21733-66N_m}{10^6}$

Table IV. Simulation parameters.

Variables	Values
Communication range	30 m
Number of nodes	400
Total energy of each sensor	3 Joules
Packet size	240, 1200 bits
Energy dissipated for receiving	50 nJ/bit
Energy dissipated for transmission	50 nJ/bit
Energy dissipated for transmit amplifier	100 pJ/bit/m^2
Bandwidth	40 kbps

after 400 queries have been processed, where x-axis and y-axis together decide the location of each sensor node and z-axis depicts the value of energy consumption. Figure 3(a) shows that some sensors in *Traditional* consume a lot of energy, especially those located along the two edges and the diagonal line of the sensor field to which the data sink belongs. So these sensors are energy hungry which consume all 3 Joules, while sensors located outside this region just consume as little as 0.1 Joules after 400 queries. Obviously, the energy consumption in *Traditional* is very unbalanced. On the contrary, the load in *IQ* balances very well, as shown in Figure 3(b), where there are

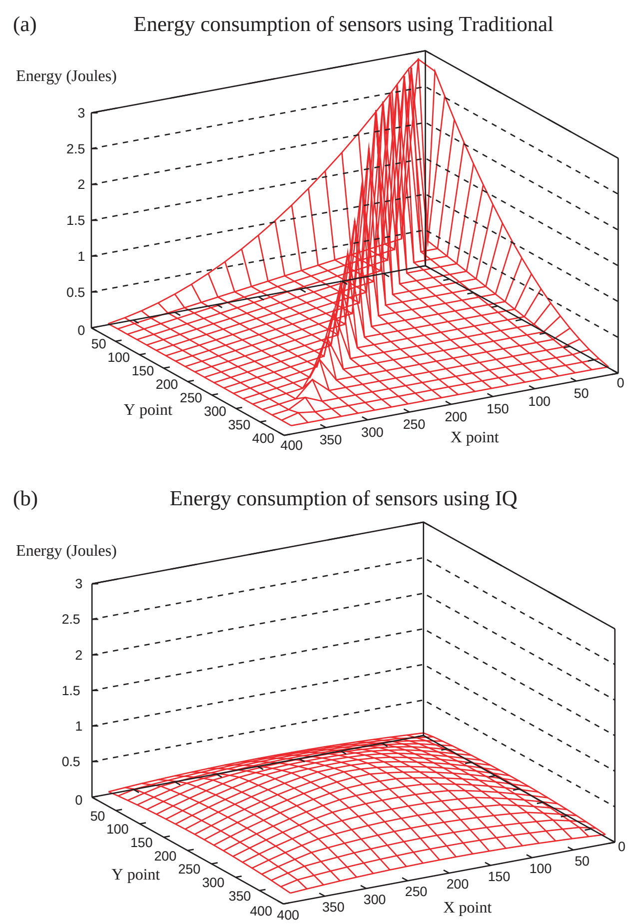


Fig. 3. Comparison of energy consumption in *Traditional* and in *IQ* using GPSR.

no energy intensive nodes. In *IQ* the maximum energy consumption is 0.6 J and the minimum energy consumption is 0.04 J. In other words, in the *IQ* protocol, by running out of the total 3 J, the sensor network can process at least 2000 queries.

4.2. Simulation Results of RLSN

In this section, we compare the two protocols in terms of RLSN. According to the default values of simulation parameters, the initial RLSN is 21, which is calculated from the formula in Section 2. Figure 4 shows the simulation results, where x-axis is the number of queries, and y-axis represents RLSN.

From the figure, we find that RLSN decreases along with the increase of the number of processed queries. In *Traditional*, RLSN drops very quickly from 21 to less than 10 during 300 queries have been processed. After 300 queries have been processed, RLSN using *Traditional* keeps stable. This is because the sensor network is already dead after 300 queries. In other words, no more query message can be sent from the sink to other alive sensors, thus no more energy is consumed. This does not mean that the remaining energy in the alive sensors is saved. On the contrary, these energy is just wasted, which can never be used in the future. In case of *IQ*, RLSN drops very slowly and smoothly. After 400 queries have been processed, RLSN is still good using *IQ*. A lot of energy is saved to be used for the future queries. From this point of view, we argue that *IQ* is more energy efficient than *Traditional*.

4.3. Simulation Results of LSN

Finally, we compare LSN, which is more interesting to application scientists and system designers. We set the value of θ (the threshold to determine the aliveness of the sensor network) to 90%. The comparison between *Traditional* and *IQ* protocols is reported in Figure 5, where

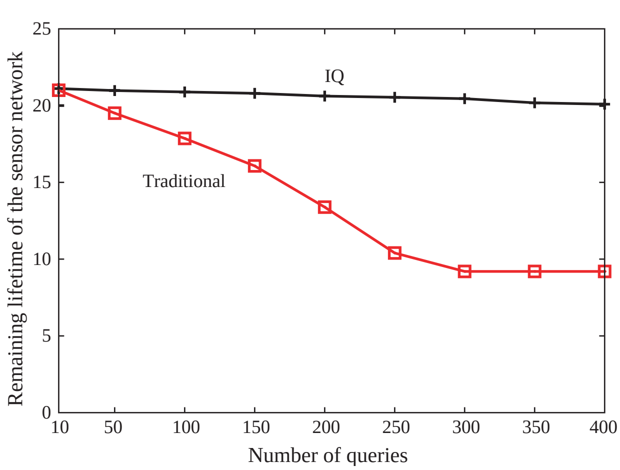


Fig. 4. Comparison of RLSN by using *Traditional* and *IQ*.

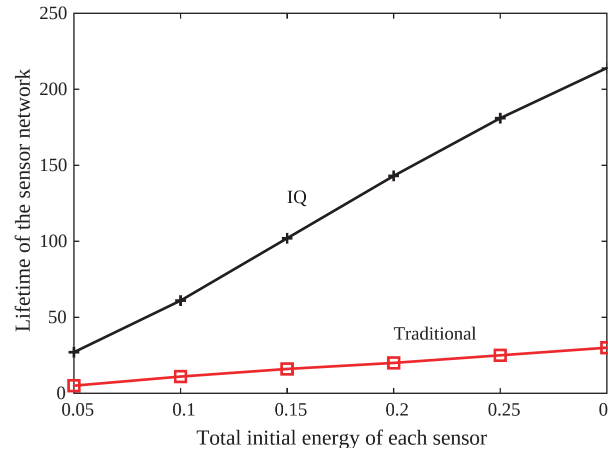


Fig. 5. Comparison of LSN by using *Traditional* and *IQ*.

x-axis is the initial energy of each sensor and y-axis is the value of LSN. From the figure, it can be easily seen that LSN increases almost linearly with the increase of initial energy in both *Traditional* and *IQ*. However, LSN increases much faster in *IQ* than that in *Traditional*, where the lifetime in the *Traditional* model is about $1/6$ of that in *IQ*. Additionally, if we decrease the value of θ, the gap between the *Traditional* and *IQ* will become much larger. Thus we conclude that *IQ* indeed extends LSN several times as that of *Traditional*.

5. RELATED WORK AND DISCUSSIONS

Energy-efficient routing protocols and optimizations to maximize the lifetime of sensor networks have been widely studied in the literature;[31] however, few of the previous efforts have been done to formally model the lifetime of the sensor network. To this end, our work is the first step towards this direction. Next, we compare our work with them respectively.

In several previous work, the lifetime of the sensor network is defined as the time for the first node to run out of power such as in Refs. [14–17] or a certain percentage of network nodes to run out of power as in Refs. [18, 19]. We think that these definitions of the lifetime of the sensor network are not satisfactory. The former is too pessimistic since when only one node fails the rest of nodes can still provide the whole sensor network appropriate functionality. While the latter does not consider the different importance of the sensors in the sensor network.

In the work of Refs. [32–35], the lifetime of the sensor network is defined as the time when the sensor network first losts connectivity or coverage. The rationale of their definition is based on the functionality of the sensor network, which is similar to our definition. However the way to detect the termination of the sensor network is different. Blough and Santi[34] define it by checking the connectivity of a graph; Mhatre et al. use a connectivity and coverage model to describe it; while we define it as the time when the remaining lifetime of the whole sensor network starts to keep constant as losing connectivity or the sensor network loses coverage.

Xue and Ganz study the lifetime of a large scale sensor network in Ref. [36]. They explore the relationship between the lifetime of a sensor network with the network density, transmission schemes and maximum transmission range. Their work is based on a general cluster-based model, and does not consider the importance of different sensors. They also aim to explore the fundamental limits of network lifetime. Compared with their work, our model is more general which can be used not only for cluster-based model. Furthermore, because we take more factors into consideration in our model, our model is more useful and flexible, in which the lifetime is calculated according to the really energy consumption.

Bhardwaj et al. define upper bounds on the lifetime of the sensor network in Refs. [32, 33]. They explore the fundamental limits of data gathering lifetime that previous strategies strive to increase. One of their motivations is to calibrate the performance of collaborative strategies and protocols, but they just give out an upper bound of the lifetime rather than the actually lifetime model for different strategies. Besides, our model can also guide the design of the low-level protocols.

Another recent work has been done also aim to derive the upper bound of the lifetime of a sensor network in Ref. [37]. The author want to explore the fundamental limits of sensor network lifetime that all algorithms can possibly achieve. Compared with their work, our model is aiming to develop models for both sensor system designer and application scientist, and we focus on calculating the more accurate lifetime of a sensor network according to different underlayer routing or query protocols. In both our work, we consider the coverage and connectivity of the sensor network.

Duarte-Melo and Liu provide a mathematical formulation to estimate the average lifetime of a sensor network in Ref. [18]. Their work aims to estimate the average lifetime of the sensor network rather than to provide a general model that can be used to measure different protocols. Our model can be used to model the lifetime of the sensor network using different communication patterns, which is more general. In their later work, they give a modeling frame for computing lifetime of a sensor network. Their approach is to maximize the functional lifetime of a sensor network and get the value of it based on the solution of a fluid-flow model. While our goal of this paper is to provide a general model for lifetime of sensor network. Besides, the calculation of their lifetime still need a lot of calculation. While our model can be easily calculated based on the centralized algorithms.

Similar to Ref. [18], several other efforts such as in Refs. [12, 13, 16, 17, 35] have been done to maximize the lifetime of sensor network. Whereas almost all their work take it as an optimization problem and build a linear programming model, then find an algorithm or a protocol to achieve the maximum lifetime, so that these approaches are always closely related to the routing protocols, rather than giving a general model for the lifetime of sensor network. Besides, most of them ignore the load imbalance problem. Even though some of them do notice the problem, they only balance the load at the routing level.

Energy-aware routing[13] is proposed by Shah et al. using a set of sub-optimal paths to increase the lifetime of the network. This approach uses one of multiple paths with a certain probability to increase the lifetime of the whole network. Another similar approach is proposed in Ref. [38], which constructs a load balance tree in the sensor networks with load balance to different branches. Their work balances the load of each data path so that extend the lifetime of sensor networks. They do not provide a formal model for the lifetime.

6. CONCLUSION AND FUTURE WORK

In this paper, we formally define RLIS, RLSN and LSN. Based on these models we compare two query protocols; both theoretical and simulation results show that *IQ* balances the load so that extends the lifetime of the sensor network.

Given the model of the lifetime of the sensor network, we will extend our work for two folds. On one hand, we will extend the model of the lifetime to make it more usable by considering more types of energy consumption such as take the sleep/active dynamics into consideration in our model. On the other hand, we will use this model to guide the development of more suitable protocols and evaluate the efficiency of more proposed protocols.

RESEARCH ARTICLE

References and Notes

1. I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, *IEEE Communications Magazine* 40, 102 (**2002**).
2. D. Estrin, D. Culler, K. Pister, and G. Sukhatme, *IEEE Pervasive Computing* 1, 59 (**2002**).
3. D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, *Proc. of ACM/IEEE MobiCom'99* 263 (**1999**).
4. D. Estrin, W. Michener, and G. Bonito, A Report From a National Science Foundation Sponsored Workshop, August (**2003**). http://www.lternet.edu/sensor_report/cyberRforWeb.pdf
5. R. Min, M. Bhardwaj, S. Cho, E. Shih, A. Sinha, A. Wang, and A. Chandrakasan, *Proc. of International Conference on VLSI Design* 205 (**2001**).
6. G. Pottie and W. Kaiser, *Communications of the ACM* 43, 51 (**2000**).
7. Crossbow technology Inc., San Jose, CA USA, http://www.xbow.com/
8. W. Shi and C. Miller, Technical Report MIST-TR-2004-009, Wayne State University, March (**2004**). http://www.cs.wayne.edu/~weisong/papers/mist-tr-2004-009.pdf
9. H. Wang, D. Estrin, and L. Girod, *EURASIP Journal on Applied Signal Processing* 4, 392 (**2003**).
10. T. He, J. A. Stankovic, C. Lu, and T. Abdelzaher, *Proc. of IEEE ICDCS'03* 46 (**2003**).

11. C. Intanagonwiwat, R. Govindan, and D. Estrin, *Proc. of ACM/IEEE MobiCom'00* 56 **(2000)**.

12. N. Sadagopan, B. Krishnamachari, and A. Helmy, *Proc. of the First International Workshop on Sensor Network Protocol and Applications* 149 **(2003)**.

13. R. Shah and J. Rabaey, *Proc. of the IEEE Wireless Communications and Networking Conference (WCNC'02)* 1, 812 **(2002)**.

14. J. H. Chang and L. Tassiulas, *IEEE/ACM Transactions on Networking* 12, 609 **(2004)**.

15. W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, *IEEE Transactions on Wireless Communications* 1, 660 **(2002)**.

16. K. Kalpakis, K. Dasgupta, and P. Namjoshi, *Proc. of IEEE Networks Conference* 685 **(2002)**.

17. I. Kang and R. Poovendran, *Proc. of IEEE ICC 2003* 3, 2256 **(2003)**.

18. E. Duarte-Melo and M. Liu, *Proc. of IEEE Globecom'02* 1, 21 **(2002)**.

19. Y. Xu, J. Heidemann, and D. Estrin, *Proc. of ACM/IEEE MobiCom'01* 70 **(2001)**.

20. W. Shi, S. Sellamuthu, K. Sha, and L. Schwiebert, *Proc. of International Workshop on Ad Hoc and Sensor Networks, in conjunction with ICPP 2004*, 488 **(2004)**.

21. N. Bulusu, J. Heidemann, and D. Estrin, *IEEE Personal Communications Magazine* 7, 28 **(2000)**.

22. D. Niculescu and B. Nath, *Proc. of IEEE GlobeCom'01* 5, 2926 **(2001)**.

23. A. Savvides, C. Han, and M. B. Srivastava, *Proc. of ACM/IEEE MobiCom'01* 166 **(2001)**.

24. A. Woo, T. Tong, and D. Culler, *Proc. of the First ACM SenSys'03* 14 **(2003)**.

25. J. Zhao and R. Govindan, *Proc. of the First ACM SenSys'03* 1 **(2003)**.

26. G. Zhou, T. He, S. Krishnamurthy, and J. A. Stankovic, *Proc. of ACM MobiSys'04* 125 **(2004)**.

27. Q. Fang, J. Gao, and L. J. Guibas, *Proc. of IEEE INFOCOM 2004* 4, 2458 **(2004)**.

28. K. Sha, Z. Zhu, and W. Shi, Technical Report MIST-TR-2004-001, Wayne State University, January 2004. http://www.cs.wayne.edu/~weisong/papers/mist-tr-2004-001.pdf

29. K. Sha, W. Shi, and S. Sellamuthu, *Sensor Network Operations*, edited by S. Phoha and T. F. La Porta, IEEE Press **(2005)**, to appear.

30. J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, *ACM SIGOPS Operating Systems Review* 34, 93 **(2000)**.

31. K. Akkaya and M. Younis, *Proc. of the IEEE Workshop on Mobile and Wireless Networks (MWN'03)* 710 **(2003)**.

32. M. Bhardwaj and A. P. Chandrakasan, *Proc. of IEEE INFOCOM 2002* 3, 1587 **(2002)**.

33. M. Bhardwaj, T. Garnett, and A. P. Chandrakasan, *Proc. of IEEE ICC 2001* 3, 785 **(2001)**.

34. D. M. Blough and P. Santi, *Proc. of ACM/IEEE MobiCom'02* 183 **(2002)**.

35. V. Mhatre, C. Rosenberg, D. Kofman, R. Mazumdar, and N. Shroff, *IEEE Transaction on Mobile Computing* 4, 4 **(2004)**.

36. Q. Xue and A. Ganz, *Elsevier Computer Communications* **(2005)**, to appear.

37. H. Zhang and J. Hou, *Proc. of the Fifth ACM International Symposium on Mobile Ad Hoc Networking and Computing* 121 **(2004)**.

38. H. Dai and R. Han, *Proc. of IEEE Globecom'03* 1, 548 **(2003)**.