LIFETIME IMPROVEMENT IN WIRELESS SENSOR NETWORKS

by

Nooshin Eghbal

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

in

Communications

Department of Electrical and Computer Engineering
University of Alberta

Abstract

We study lifetime of linear and two-dimensional Wireless Sensor Networks (WSNs) for the case where wireless sensors are not able to change their transmission power. In this case, unlike majority of existing works, power control cannot be used to increase network's lifetime. Instead, other approaches such as adding extra nodes and careful node placement and routing can be employed to improve network's lifetime. For linear WSNs, we consider two scenarios. In the first scenario, every node is required to periodically send packets to the base station while, in the second scenario, only a subset of nodes (e.g., a subset that covers the whole network) are required to do so. For the first scenario, we derive bounds on the maximum lifetime achievable. For the second scenario, we establish a lower bound on the minimum number of nodes required to achieve any specific lifetime, as in this scenario, lifetime can be arbitrarily increased by adding enough number of nodes in the network.

Next, we extend our work to two-dimensional WSNs. As before, we do not use any power control mechanism. Similar to the first application in linear networks, we assume that all nodes send reports to the base station, periodically. However, rather than node placement, we focus on routing. Also, we assume that each node can aggregate all the received packets into a single packet. Our primary objective here is to improve network's lifetime. As a secondary objective, we aim at reducing network's delay while preserving network's lifetime.

Acknowledgements

Several persons helped me throughout this journey. First of all, I wish to acknowledge my supervisor, Dr. Majid Khabbazian, for all his support and guidance, that made it possible for me to fulfill this research project. I would also like to thank my examining committee, Dr. Masoud Ardakani and Dr. Hai Jiang, for all their guidance and insightful comments, which improved my thesis a lot.

I desire to thank my kind family whose support and kindness make me strong to overcome obstacles through this way. Specially, my lovely parents, who have always encouraged me to learn and improve myself from the first day of school until now. I would also like to thank my sisters and brother, who patiently take care of their little sister in various aspects of life including education.

My special thanks is to my best friend, my dearest person, my beloved husband, Pooria, without whom I could not finish my M.Sc. program. He not only helped me to improve my skills, but also is a strong supporter near whom I feel safe and calm, which are the most priceless feelings in this world.

Table of Contents

List of Figures

Chapter 1

Introduction

A wireless sensor is a typically small and inexpensive hardware deployed in various applications to measure, for example, temperature, light, humidity, sound, vibration, pressure, motion, and leakage.

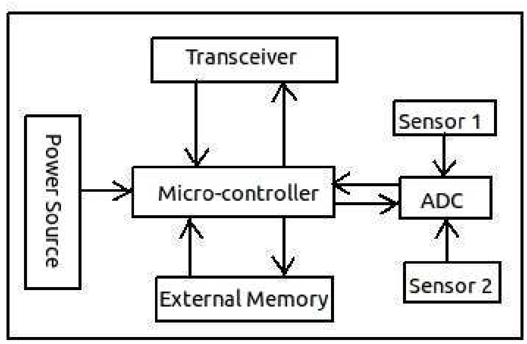


Figure 1.1: Basic components of a wireless sensor node

A Wireless Sensor Network (WSN) is a collection of wireless sensor nodes placed in an area of interest for a particular application (in WSNs, sensors sense the environment and send their measurements to one or more base stations directly or through other nodes). For example, applications of WSNs in military include monitoring friend and enemy forces, targeting, and nuclear, biological, and chemical attack detection. Environmental applications include habitat monitoring, animal tracking, forest-fire detection, and precision farming. In healthcare, WSNs can be

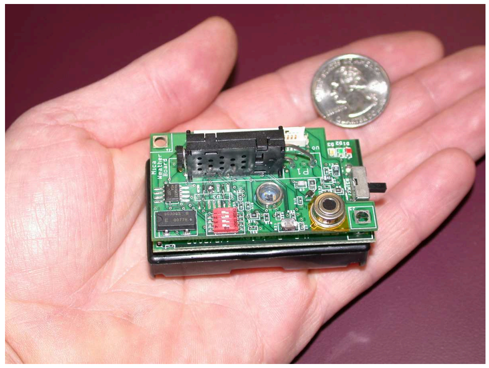


Figure 1.2: The Mica wireless sensor node used for environmental monitoring applications [24].

used for monitoring blood sugar levels of persons with diabetes and the patient's location and condition, and for improving life quality of the elderly [26]. Furthermore, WSNs have applications in industrial automation such as energy control systems, security, and wind turbine health monitoring [31].

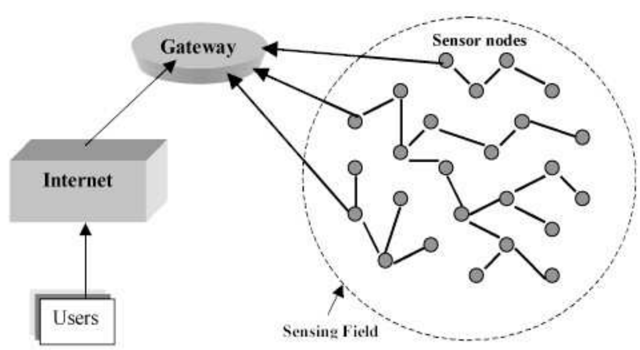


Figure 1.3: General view of a wireless sensor network

Linear Wireless Sensor Networks (LWSNs) in which nodes are placed along a straight line, have many applications in the modern industry [34, 16], including:

- Oil, gas, and water pipeline monitoring;
- Road (e.g., driver-alert), bridge, river, and tunnel monitoring;
- Railroad and subway monitoring;
- Monitoring of AC Powerlines;

○ Border monitoring.

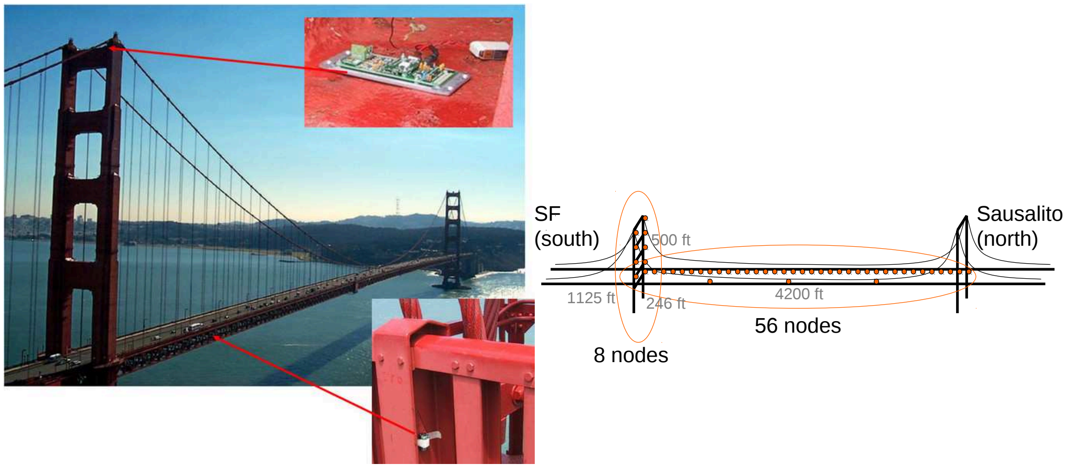


Figure 1.4: Structural health monitoring of the Golden Gate Bridge for vibration measurement [18].

Figure 1.5: Using ultrasonic sensors for corrosion monitoring of the Trans-Alaska oil pipeline.

1.1 Motivation

WSNs have certain constraints that necessitate careful strategies and designing protocols for managing them. One of the major challenges in deploying WSNs is related to the network's lifetime. A wireless sensor device typically has a limited energy supply, which is normally in the form of a small battery. When sensors die (e.g., when they run out of battery), the network may become disconnected, or stop satisfying the application requirements (such as area coverage). Therefore, to keep the network alive for as long as possible, it is important to reduce the energy consumptions of individual sensors. Limited bandwidth, storage capacity, and processing capabilities are other constraints of sensor nodes.

In this thesis, we study applications in which nodes send their own packets to a special node called base station using multi-hop routing. Thus, a node not only sends its own packet, but also has to participate in relaying (reception and sending) other nodes' packets. As a result, the traffic of packets in the area near the base station is more, and nodes located in this area run out of energy faster than nodes far from the base station. This problem is called energy hole problem in the literature [20].

To improve lifetime of the network, previous works tried to solve energy hole problem by deploying various methods such as node placement [5, 13, 10, 12, 3], careful routing, for example Mixed-routing in [38, 39], sleeping [6, 1, 37], clustering [11, 22, 4, 19, 33, 29, 17], compression and aggregation [21, 8]. Majority of existing works use power control to increase network's lifetime. However, this may not be an option in all types of wireless sensors as they are typically simple devices. In this thesis, we study lifetime of linear and two-dimensional WSNs for the case where wireless sensors are not able to change their transmission power.

In Chapter 2 of this thesis, we study lifetime improvement in LWSNs. We consider two different scenarios. In the first scenario, all nodes periodically send packets to the base station, while in the second scenario, only a subset of nodes (e.g., a subset that covers the whole network area) does so. We investigate the effect of node placement on the lifetime when the transmission power of sensors is fixed and identical. For the first scenario, we derive bounds on the maximum lifetime achievable. For the second scenario, we establish a lower bound on the minimum number of nodes required to achieve any specific lifetime.

In Chapter 3, we extend our study of the lifetime improvement problem to two-dimensional wireless sensor networks. We consider the case where all nodes are required to periodically send report packets to the base station, and every node aggregates the packets it receives from the other nodes into one packet, instead of relaying the individual packets. For a fixed node placement, we propose a routing algorithm that prolongs lifetime while trying to reduce the total delay.

Finally, Chapter 4 summarizes our work and concludes the thesis.

Chapter 2

Linear Wireless Sensor Networks

In this chapter, we study the lifetime of linear wireless sensor networks for two different scenarios. In the first scenario, every node is required to periodically send packets to the base station while, in the second scenario, only a subset of nodes (e.g., a subset that covers the whole network) are required to do so. First, we explain the related background in Section 2.1. Then, we review the existing works related to our problem in Section 2.2. In Section 2.3, we specify our assumptions and define the problem studied in this chapter. Next, the first and second scenarios are explained and analyzed in Section 2.4 and 2.5, respectively. We conclude this chapter by presenting our simulation results in Section 2.6.

2.1 Background

2.1.1 Sources of Energy Consumption

Every wireless sensor node can consume energy for three different functions: transmitting packets, receiving packets, and sensing data from the environment. An energy model describes how energy is consumed for each of theses functions. The most widely used energy model in the literature makes the following assumptions: the energy consumed for sending a bit is $\alpha_1 + \alpha_2 d^n$, where α_1 is the energy per bit consumed in the transmitter circuit, α_2 shows the energy dissipated in the transmit op-amp circuit, d is the distance between transmitter and receiver, and n is the path loss exponent; the energy consumed for receiving or sensing a bit is assumed to be fixed values α_3 and α_4, respectively.

Often, it is assumed that $2 \leq n \leq 4$, although some others consider 6 as the upper bound for n [19]. A typical value of α_1, α_3, and α_4 is 50×10^{-9} J/bit. Also, α_2 is typically set to 100 pJ/bit/m^2 when $n = 2$, or 1000 nJ/bit/m^4 when $n = 4$.

2.1.2 Complexity

In this section, we explain some mathematical notations used in the thesis. Big O ($\mathcal{O}$), Omega (Ω), and Theta (Θ) notations are mostly used to describe how fast a function grows asymptotically.

Big O ($\mathcal{O}$) notation is used for upper bounding the growth rate of a function. Formally, $B(t) = \mathcal{O}\left(A(t)\right)$ if and only if there exist constant C and t_0 such that $|B(t)| \leq C|A(t)|$ for all $t > t_0$. For example, if $B(t) = 3t^2 + 5t + 2$, we will have $B(t) = \mathcal{O}\left(t^3\right)$ or $B(t) = \mathcal{O}\left(t^2\right)$.

Omega (Ω) notation is used for lower bounding the growth rate of a function. Formally, $B(t) = \Omega\left(A(t)\right)$ if and only if there exist constant C and t_0 such that $|B(t)| \geq C|A(t)|$ for all $t > t_0$. For example, if $B(t) = \sqrt{n}$, we will have $B(t) = \Omega\left(\log n\right)$.

Theta (Θ) notation is used for tight bounding the growth rate of a function. Formally, $B(t) = \Theta\left(A(t)\right)$ if $B(t) = \mathcal{O}\left(A(t)\right)$ and $B(t) = \Omega\left(A(t)\right)$. In other words, $B(t) = \Theta\left(A(t)\right)$ if and only if there are constant C_1, C_2, and t_0 so that $C_1 A(t) \leq B(t) \leq C_2 A(t)$ for all $t > t_0$. As an example, if $B(t) = 3t^2 + 5t + 2$, we will have $B(t) = \Theta\left(t^2\right)$.

2.2 Related Work

The lifetime improvement problem for linear wireless sensor networks has been long studied in the literature. Existing strategies for solving this problem are typically based on careful node placement, transmission power control, routing, clustering and compression at intermediate nodes. The majority of the existing solutions use a combination of the aforementioned strategies. Following, we highlight some of the existing works on linear networks that use node placement as part of their strategies, since, in this work, we study the effect of node placement on the lifetime of this kind of networks.

Node placement plays an important role not only in satisfying the application requirements such as coverage and connectivity, but also in load balancing and energy consumption of nodes. The aim of node placement could be lifetime maximization or only minimizing the energy required to deliver a packet to the base station.

In [2], the authors showed that the minimum energy consumption for relaying packets is achieved when nodes are placed at an equal distance, named characteristics distance, that is equal to $\sqrt[n]{\frac{\alpha_1+\alpha_3}{\alpha_2(n-1)}}$, where α_1, α_2, α_3, and n are the parameters of the energy model, as defined in Section 2.1.1. The transmission power of all nodes is set to the value required for sending packets to this distance. The same idea was used in other works [41, 13, 36] to increase lifetime. Using a placement according to characteristic distance and power control, the authors of [41] proposed two routing algorithms, MERR (Minimum Energy Relay Routing) and AMERR (Adaptive Minimum Energy Relay Routing), which achieve a lower bound on the optimal power consumption of routing in a linear WSN.

It is noteworthy that minimizing the energy consumption to deliver packets can have positive effect on network's lifetime but it is not sufficient, as it cannot solve the energy hole problem. To deal with this, existing works typically use load balancing, i.e., they try to balance the energy consumption of nodes so all nodes run out of battery around the same time. In such setting, it is natural to assume that lifetime nearly ends when a node runs out of battery.

Typical strategies used for load balancing mostly include node placement. The authors of [36] proposed two node placement strategies for solving the energy hole problem. The first strategy uses node placement according to characteristics distance, followed by putting extra nodes as additional batteries and using data compression. The authors provided simulation results for this strategy when the number of additional nodes grows linearly as the nodes get closer to the base station. The second strategy is node placement with variable hop distance such that all nodes consume the same amount of energy. Both strategies need the ability to control the transmission power.

All the aforementioned work use transmission power control on top of other methods such as node placement. In this chapter, we consider a fixed transmission power for all nodes and aim at maximizing network's lifetime through adding extra nodes in the network and careful node placement. The closest existing work to our settings is [27]. In [27], all nodes periodically generate and send packets to the base station. This is the same setting we use in the first scenario we study in this chapter. The authors in [27] used a linearly increasing distribution to place nodes in the network. Using this distribution, the density of nodes linearly increases as they become closer to the base station. The analysis of our first scenario generalizes this.

2.3 Problem Statement

In this chapter, we consider two scenarios with different requirements. In both scenarios, the sensing ranges of the set of alive nodes must cover the whole network. We define a round as the time during which the base station receives the packet from all required nodes. We assume that sensors are not capable of adjusting their transmission power. More specifically, we assume that the transmission power of all the nodes is identical. We define network's lifetime as the time during which application requirements are satisfied.

In the first scenario, the base station has to receive a packet from every alive node in each round, while, in the second scenario, the base station only needs to receive packets from any subset of nodes covering the whole network in every round. Since we study bounds, we consider an ideal case where nodes sleep during idle time (that is when they do not transmit or are not reached by any packet), and there is no packet collision. In practice, however, nodes consume energy for, for example, retransmitting packets that were collided, and switching to/from sleeping mode. A node u may be reached by a packet which was transmitted for another node v. In this case, we say that u overhears the packet. We separate overhearing form receiving as a node u can consume less energy for overhearing than for receiving using techniques such as early overhearing avoidance [14, 32, 15], and adaptive downclocking [40, 23].

In the first scenario, adding extra nodes may not necessarily increase network's lifetime as extra nodes also generate packets in every round, increasing the load of network. However, when the cost of overhearing (i.e., energy consumption of overhearing) is exponentially smaller than the cost of transmission, in an ideal case, the network's lifetime can almost approach the optimum lifetime. Interestingly, we show that if the cost of overhearing is comparable with that of transmission, we cannot do much better than going with the trivial approach of putting one node in each transmission range distance.

In the second scenario, adding extra nodes to the network can always improve the network's lifetime as extra nodes can act as extra batteries. For example, consider two nodes next to each other. The second node can sleep until the first node runs out of battery, at which point the second node can wake up and take the responsibility of the first one. Clearly, in this scenario, the network's lifetime can arbitrarily increase by adding nodes. Therefore, the main question we study for this scenario is the minimum number of nodes required to achieve a certain lifetime.

We explain the network and energy models used in this chapter in Sections 2.3.1

and 2.3.2, respectively.

2.3.1 Network Model

We assume that nodes use a fixed and identical transmission power, hence their transmission range is fixed and identical. Let R_t and R_s denote the transmission range and sensing range of nodes, respectively. For scenario 1, we assume that $R_s = \Omega(R_t)$, that is R_s is greater than a constant factor of R_t (e.g. $R_s \geq \frac{R_t}{2}$). Therefore, by dividing the network into intervals of size $\tau = \min\{R_t, R_s\} = \theta(R_t)$, and putting at least one node in each interval, we can satisfy both coverage and connectivity requirements. For simplicity, we assume that $\frac{l}{\tau} = m$ is a natural number, where l is the length of the network. We define $I_i = ((m-i)\tau, (m-i+1)\tau]$, $1 \leq i \leq m$, to be the half-open intervals splitting the network. We assume that the base station is located at x-coordinate zero, thus I_1 is the farthest interval to the base station. We divide time into rounds with enough duration such that every packet generated in a round can be delivered to the base station by the end of that round. Finally, we assume that each node generates at most one packet/report at the beginning of each round.

2.3.2 Energy Model

In our problem, since nodes have an identical transmission power, they consume the same amount of energy to transmit packets. We use A, B, and $B' \leq B$ to denote the energy required to transmit, receive and overhear a packet, respectively.

We assume that $B = \mathcal{O}(A)$, that is the energy required for receiving a packet is at most a constant factor of that needed for transmitting it. Let $t = \frac{A}{B'}$. Without loss of generality, we normalize the parameters such that $B' = 1$, $A = t$. We denote the normalized initial energy of nodes by E_{init}, and assume that $E_{init} \gg mt$, where m is the parameter defined in Section 2.3.1.

2.4 Scenario 1: Every Node Generates Reports

In this section, we study network's lifetime for scenarios where, due to the application requirement or simplicity of devices, all nodes must periodically send packets to the base station. This can also be helpful when nodes' measurements are noisy, as extra nodes provides redundancy which can be used to obtain a better estimate of measurements. The analytical results presented in this section are asymptotic. Later, we use simulation to confirm the results, and estimate the constants hidden in the asymptotic analysis.

Since the results of this section are asymptotic, we can ignore the cost of reception because i) the energy required for receiving a packet is at most a constant factor of that needed for transmitting; ii) there are at most one reception per each transmission. In other words, in the asymptotic results, one can ignore the cost of reception, and increase the cost of transmission by a constant factor to compensate for the cost of reception ignored. However, note that we cannot ignore the cost of overhearing as the number of packets a node overhears can be significantly more than the number of packets it receives or transmits.

One trivial approach to the problem is to put exactly one node in the middle of each interval I_i, $1 \leq i \leq m$, and use a simple routing algorithm in which the node in interval I_i, $1 \leq i < m$, forwards its packets to the node in interval I_{i+1}. We refer to this approach as the basic method. In the basic method, at the beginning of each round, each node generates one packet and sends it to the node in the next interval. Therefore, the node in I_i receives $i-1$ packets, and transmits i packets, in every round. In particular, nodes in the intervals close to the base station (i.e., nodes that are at most a constant number of intervals away from the base station) receive and transmit $\theta(m)$ packets in every round. Note that a node may overhear several copies of every packet as it can be reached by nodes that are a few intervals away.

However, in the basic method, the number of copies of each packet overheard is $\mathcal{O}(1)$ because i) due to the length of intervals, a node can only be reached by nodes that are at most a constant number of intervals away; ii) a packet moves to the next interval in each round, thus, it will be out of receiving range of every given node after a constant number of transmissions.

By the definition of τ, and because $R_s = \Omega(R_t)$, we get $\frac{R_t}{\tau} = \mathcal{O}(1)$. Therefore, every node in the interval $\mathcal{I} = \cup_{i=m-\lfloor \frac{R_t}{\tau} \rfloor}^{m} I_i$ has to receive and transmit $\theta(m)$ packets, in every round. Also, every node in $\mathcal{I}$ overhears $\mathcal{O}(1)$ copies of every packet. The network becomes disconnected if all the nodes in $\mathcal{I}$ die, since

$$|\mathcal{I}| = \left(\lfloor \frac{R_t}{\tau} \rfloor + 1 \right) \tau \geq R_t,$$

where $|\mathcal{I}|$ denotes the length of interval $\mathcal{I}$. Therefore, the lifetime of network using the basic method is

$$\mathcal{O}\left(\frac{E_{init}}{mA + mB'} \right) = \mathcal{O}\left(\frac{E_{init}}{mt} \right).$$

2.4.1 Beating the Basic Method

An interesting question is whether we can improve the network's lifetime achieved by the basic method. Following, we propose a method that, in an ideal case, can achieve a factor of $\Omega(\min\{m, \log t\})$ improvement over the basic method when $t > 1$. We further prove that this improvement differs the optimum improvement by $\mathcal{O}(\log t)$ factor. In particular, we prove that the maximum improvement possible over all placements and routing algorithms is $\mathcal{O}\left(\min\{m, \log^2 t\}\right)$. This implies that, non-negligible improvements are theoretically possible only when B' is significantly smaller than A. Techniques like early overhearing avoidance [14, 32, 15], and adaptive downclocking [40, 23] can reduce B' but may have little effect on increasing the lifetime of network unless they can make $\frac{A}{B'}$ exponentially large in terms of the number of intervals. An extreme case is when nodes listen to the channel only when they have to receive a packet. In this case $B' = 0$, and we can achieve a factor of $\theta(m)$ improvement over the lifetime of the basic method, which is asymptotically the best we can do in this model.

2.4.2 Proposed Method

Here, we propose a node placement with a simple routing algorithm, and evaluate the improvement achieved using the proposed method instead of the basic method. Later, we show that this improvement is close to the maximum improvement possible.

Let $\alpha = \min\{2, t^{\frac{1}{m}}\}$. In our method, we put α^{i-1} nodes[1] in interval I_i, $1 \leq i \leq m$, and use a simple routing algorithm in which a node in interval I_i, $1 \leq i < m$, forwards its packet to a node in interval I_{i+1}, selected uniformly and independently at random (nodes in interval I_m send packets directly to the base station).

Theorem 2.1. *The network's lifetime using the proposed method is* $\Omega\left(\frac{E_{init} \cdot \log \alpha}{t+\alpha^m}\right)$, *with high probability.*

Proof. First we establish an upper bound on the expected energy consumption of nodes per round. Then, we show that the actual energy consumption of a node over L rounds is the same order as L times the upper bound on its expected energy consumption per round. Then we find a number L such that over the first L rounds, the maximum energy consumption by any node is not more than E_{init}. We show that $L = \Omega\left(\frac{E_{init} \cdot \log \alpha}{t+\alpha^m}\right)$, which yields the result as the network's lifetime is at least L rounds.

Since our results are asymptotic, and $R_s = \Omega(R_t)$, without loss of generality, we assume that $\tau = R_t$, that is the length of intervals I_i, $1 \leq i \leq m$, is R_t. Let fix an interval I_i and a node u in I_i. The number of packets that enter I_i in any round is

$$\sum_{j=1}^{i-1} \alpha^{j-1} = \frac{\alpha^{i-1} - 1}{(\alpha - 1)},$$

[1]In general, α^{i-1} is not a natural number. We show later how to convert these numbers into natural numbers without affecting the final result.

which is the number of nodes in intervals with index less than i. It is because every packet generated by those nodes should be relayed by a node in I_i. Since nodes are selected uniformly at random to relay packets and there are α^{i-1} nodes in I_i, the expected number of packets that u relays in any round is $\frac{\alpha^{i-1}-1}{(\alpha-1)\alpha^{i-1}}$. In addition to this, u transmits its own packet in every round and overhears $\mathcal{O}(1)$ copies of every packet generated by nodes in intervals j, $1 \leq j \leq i + \mathcal{O}(1)$. Therefore, the expected energy consumption of u in each round is

$$\mathcal{O}\left(\left(\frac{\alpha^{i-1}-1}{(\alpha-1)\alpha^{i-1}}+1\right)\cdot t+\frac{\alpha^{i-1+\mathcal{O}(1)}-1}{(\alpha-1)}\right), \tag{2.1}$$

which is

$$\mathcal{O}(t+\alpha^m) \tag{2.2}$$

if $\alpha = 2$. Otherwise, if $\alpha = t^{\frac{1}{m}}$, (2.1) becomes

$$\begin{aligned}
&\mathcal{O}\left(\frac{\alpha^{i-1}-1}{(\alpha-1)\alpha^{i-1}}\cdot t+\frac{\alpha^{i-1}-1}{(\alpha-1)}\right)\\
&=\mathcal{O}\left(\frac{\alpha^{i-1}}{(\alpha-1)\alpha^{i-1}}\cdot t+\frac{\alpha^{m}}{(\alpha-1)}\right)\\
&=\mathcal{O}\left(\frac{1}{(\alpha-1)}\cdot t+\frac{t}{(\alpha-1)}\right)\\
&=\mathcal{O}\left(\frac{t}{\alpha-1}\right)\\
&=\mathcal{O}\left(\frac{mt}{\log t}\right)
\end{aligned} \tag{2.3}$$

where the last equality is by the fact that $(1+\frac{\log t}{m})^m \leq e^{\frac{\log t}{m}\cdot m} = t$, hence $\alpha = t^{\frac{1}{m}} \geq 1 + \frac{\log t}{m}$. By (2.2) and (2.3), we get that the expected energy consumption of u in every round is

$$\mathcal{O}\left(\frac{t+\alpha^m}{\log \alpha}\right).$$

Therefore, the expected energy consumption of u over L rounds is $\mathcal{O}\left(L\cdot\frac{t+\alpha^m}{\log\alpha}\right)$. Note that nodes are selected uniformly and independently at random. Therefore, when E_{init} is large enough, the actual energy consumption of u over L rounds is still $\mathcal{O}\left(L\cdot\frac{t+\alpha^m}{\log\alpha}\right)$. Formally speaking, if $E_{init} \gg \log(\frac{1}{\epsilon})\cdot L\cdot\frac{t+\alpha^m}{\log\alpha}$, then by a

Chernoff bound we get that the actual energy consumption of u is $\mathcal{O}\left(L \cdot \frac{t+\alpha^m}{\log \alpha}\right)$, with probability at least $1-\epsilon$. Consequently, u will have enough energy for L rounds with high probability if $L = \theta\left(\frac{E_{init} \cdot \log \alpha}{t+\alpha^m}\right)$ and E_{init} is large enough. □

The following corollary is a direct result of Theorem 2.1.

Corollary 2.2. *Suppose $t \leq 2^m$. Then, replacing the basic method with the proposed method improves the lifetime of network by an order of $\log t$, w.h.p.*

Proof. Replacing α with $t^{\frac{1}{m}}$ in Theorem 2.1, we get that the lifetime of network is $\Omega\left(\frac{E_{init} \cdot \log t}{mt}\right)$ w.h.p., an order of $\log t$ better than the network's lifetime using the basic method, which is $\mathcal{O}\left(\frac{E_{init}}{mt}\right)$. □

Corollary 2.3. *The proposed method is asymptotically optimum when $t \geq 2^m$.*

Proof. When $t \geq 2^m$, we get that $\alpha = 2$. Therefore by Theorem 2.1, the network's lifetime will be

$$\Omega\left(\frac{E_{init}}{t+2^m}\right) = \Omega\left(\frac{E_{init}}{t}\right).$$

This is the maximum lifetime of network as lifetime of every node is $\mathcal{O}\left(\frac{E_{init}}{t}\right)$, since, in every round, each node has to transmit at least a packet, which requires t units of energy. □

The proposed method puts α^{i-1} nodes in interval I_i. Since α^{i-1} is not an integer in general, we use the following simple algorithm to round it. The main property of the algorithm stated in Lemma 2.4 is that it does not change i) the order of the number nodes in an interval and ii) the order of the total number of nodes in intervals with index at most i, for every $1 \leq i \leq m$. Therefore, the order of the expected number of transmissions, relaying and overhearing that each node has to do remains the same as before.

Algorithm 1

Input: A (array of real numbers), m (size of array)
Output: B (array of natural numbers)
$B[0] =$ Round $A[0]$
diff $= B[0] - A[0]$
for $i = 1$ to $m - 1$ **do**
  if *diff* > 0 **then**
    $B[i] =$ Round down $A[i]$
  else
    $B[i] =$ Round up $A[i]$
  end if
  diff = *diff* $+(B[i] - A[i])$
end for
Return B

Lemma 2.4. *In Algorithm 1, let A be an array of positive real numbers of length m . Then, for every j, $0 \leq j \leq m-1$, we have*

$$\left| \left(\sum_{i=0}^{j} A[i] \right) - \left(\sum_{i=0}^{j} B[i] \right) \right| \leq \frac{1}{2}$$

Proof. Note that the variable *diff* in Algorithm 1 is equal to $\left(\sum_{i=0}^{j} A[i]\right) - \left(\sum_{i=0}^{j} B[i]\right)$ by the end of jth iteration of the loop. Therefore, to prove the lemma, it suffices to show that

$$|diff| \leq \frac{1}{2} \tag{2.4}$$

We do this by induction on j. Since *diff* is initially set to $Round(A[0]) - A[0]$, Inequality (2.4) holds for $j = 0$. Suppose (2.4) holds for $j = k$. If *diff* > 0 at the beginning of $(k + 1)$th iteration, then $A[k + 1]$ is rounded down to get $B[k + 1]$. Therefore, we have $-\frac{1}{2} \leq B[k+1] - A[k+1] \leq 0$, hence the absolute value of *diff* by the end of $(k + 1)$th iteration will be at most $\frac{1}{2}$. Similarly, we can show that the absolute value of *diff* by the end of $(k + 1)$ iteration will be at most $\frac{1}{2}$ for the case where *diff* ≤ 0 at the beginning of $(k + 1)$th iteration. □

It should be mentioned that the proposed algorithm is not suitable for applications with short rounds. It is because, the proposed algorithm adds extra nodes to the network. When rounds are short, adding extra nodes can increase the network contention at the MAC layer, increasing the number collisions, hence the number of transmissions. The proposed method can be viewed as a theoretical result showing that, in ideal cases, the upper bound proven in the next section can be almost matched.

2.4.3 Upper Bound on Lifetime Improvement

Earlier, we showed that our proposed method is asymptotically optimum when $t \geq 2^m$, i.e., when $\alpha = 2$. Similarly, we can show that the proposed method is asymptotically optimum when $t \geq 2^{\Omega(m)}$. Here, we study the case where $t < 2^{o(m)}$, which is the more practical case. In this case, we show that the maximum improvement possible (with respect to the basic method) over all possible placements and routing algorithms is $\mathcal{O}\left(\log^2 t\right)$. This is an interesting result which roughly states that there is not much improvement possible in this model if the cost of overhearing is not significantly smaller than the cost of transmission.

We say a node is alive in a round if it transmits its own packet in that round. Recall that, in the model used for the first scenario, every node transmits its packet in every round if it has enough energy for that. The network is called alive in a round if there is at least one alive node in every interval, and the base station receives the packets of all the alive nodes in that round. We use L to denote the lifetime of network, that is the number of rounds in which the network is alive. We define the energy of an interval in a round as the sum of the energies of its alive nodes in that round. The following lemma is used in Theorem 2.7 which proves the upper bound on the lifetime of network using any solution, which is determined by the initial node placement and the routing algorithm used.

Lemma 2.5. *Let $l \geq 1$ be an integer, $2 \leq s \leq 2^l$ be a real number, and $r_i \geq 1$,*

$1 \le i \le l$, be an arbitrary sequence of real numbers. Then

$$\exists r_i, 1 \le i \le l \textit{ s.t. } (1+\frac{s}{r_i})\sum_{j=1}^{i} r_j \ge \frac{s}{e}\cdot\lfloor\frac{l}{\log_2 s}\rfloor,$$

where e is the base of natural logarithm.

Proof. Let $g = \lfloor\frac{l}{\log_2 s}\rfloor$. since $2 \le s \le 2^l$, we have $1 \le g \le l$. Lemma's statement holds for $g = l$ (i.e., $s = 2$) because, setting $i = l$, we get

$$(1+\frac{s}{r_l})\sum_{j=1}^{l} r_j \ge \sum_{j=1}^{l} r_j \ge l \ge \frac{2l}{e} = \frac{s}{e}\cdot\lfloor\frac{l}{\log_2 s}\rfloor$$

Therefore, in the following, we assume that $g < l$.

By the lemma's assumption, $r_i \ge 1$, for $1 \le i \le g$. Let k be the smallest integer such that $g < k \le l$, and $r_k < (1+\frac{1}{g})^{k-g-1}$. Later, we cover the case where there is not such an integer k. If there exists such an integer k, we get

$$\begin{aligned}
&(1+\frac{s}{r_k})\sum_{j=1}^{k} r_j \\
&\ge \frac{s}{r_k}\sum_{j=1}^{k} r_j \\
&\ge \frac{s}{r_k}\left(g + r_k + \sum_{j=g+1}^{k-1}(1+\frac{1}{g})^{j-g-1}\right) \\
&= s + \frac{s}{r_k}\left(g + \sum_{j=g+1}^{k-1}(1+\frac{1}{g})^{j-g-1}\right) \\
&\ge s + \frac{s}{(1+\frac{1}{g})^{k-g-1}}\left(g + \sum_{j=g+1}^{k-1}(1+\frac{1}{g})^{j-g-1}\right) \\
&\ge \frac{s}{(1+\frac{1}{g})^{k-g-1}}\left(g + \sum_{j=g+1}^{k}(1+\frac{1}{g})^{j-g-1}\right) \\
&= \frac{s}{(1+\frac{1}{g})^{k-g-1}}\left(g + \frac{(1+\frac{1}{g})^{k-g}-1}{(1+\frac{1}{g})-1}\right) \\
&= \frac{s}{(1+\frac{1}{g})^{k-g-1}}\left(g(1+\frac{1}{g})^{k-g}\right) \\
&= sg(1+\frac{1}{g})
\end{aligned}$$

$$
\begin{aligned}
&= s(g+1) \\
&\geq s\lfloor \frac{l}{\log_2 s} \rfloor \\
&\geq \frac{s}{e} \cdot \lfloor \frac{l}{\log_2 s} \rfloor .
\end{aligned}
$$

This proves the result. Therefore, it remains to cover the case where $r_k \geq (1 + \frac{1}{g})^{k-g-1}$ for every $g < k \leq l$. In this case, we have

$$
\begin{aligned}
(1 + \frac{s}{r_l}) \sum_{j=1}^{l} r_j &\geq \sum_{j=1}^{l} r_j \\
&\geq g + \sum_{j=g+1}^{l} (1 + \frac{1}{g})^{j-g-1} \\
&= g + \frac{(1+\frac{1}{g})^{l-g} - 1}{(1+\frac{1}{g}) - 1} \\
&= g(1 + \frac{1}{g})^{l-g} = g(1 + \frac{1}{g})^{l} / (1 + \frac{1}{g})^{g}
\end{aligned}
\tag{2.5}
$$

Note that

$$
(1 + \frac{1}{x})^x \leq e
$$

for every positive real number x. Also,

$$
(1 + x)^y \geq 2^{xy}
$$

for real numbers $0 \leq x \leq 1$ and $y \geq 0$, because

$$
(1 + x) \geq 2^x
$$

for $0 \leq x \leq 1$. Therefore, by (2.5) and using the fact that $g \geq 1$ (hence $\frac{1}{g} \leq 1$), we get

$$
\begin{aligned}
(1 + \frac{s}{r_l}) \sum_{j=1}^{l} r_j &\geq g(1 + \frac{1}{g})^{l} / (1 + \frac{1}{g})^{g} \\
&\geq g 2^{\frac{l}{g}} / e \\
&\geq \lfloor \frac{l}{\log_2 s} \rfloor 2^{\log_2 s} / e \\
&= \frac{s}{e} \cdot \lfloor \frac{l}{\log_2 s} \rfloor ,
\end{aligned}
$$

which completes the proof.

□

Corollary 2.6. *Let* $l' = \lfloor \frac{l}{2} \rfloor + 1$. *Let* $l \geq 1$ *be an integer,* $2 \leq s \leq 2^{l-l'}$ *be a real number, and* $r_i \geq 1$, $1 \leq i \leq l$, *be an arbitrary sequence of real numbers.*
Then, there exist l' *distinct numbers* $1 \leq i_1, i_2, \ldots, i_{l'} \leq l$ *such that*

$$\forall i \in \{i_1, i_2, \ldots i_{l'}\} \quad (1 + \frac{s}{r_i}) \sum_{j=1}^{i} r_j = \Omega\left(\frac{ls}{\log_2 s}\right).$$

Note that $l' > \frac{l}{2}$.

Proof. By Lemma 2.5, there exists a number r_{i_1} for which we have

$$(1 + \frac{s}{r_{i_1}}) \sum_{j=1}^{i_1} r_j \geq \frac{s}{e} \cdot \lfloor \frac{l}{\log_2 s} \rfloor = \Omega\left(\frac{ls}{\log_2 s}\right).$$

Removing r_{i_1} from the sequence and applying Lemma 2.5 on the remaining sequence of size $l-1$, we get that there exists a number r_{i_2}

$$(1 + \frac{s}{r_{i_2}}) \sum_{j=1\ldots,i_2,j\neq i_1} r_j \geq \frac{s}{e} \cdot \lfloor \frac{l-1}{\log_2 s} \rfloor.$$

Therefore,

$$\begin{aligned}(1 + \frac{s}{r_{i_2}}) \sum_{j=1}^{i_2} r_j &\geq (1 + \frac{s}{r_{i_2}}) \sum_{j=1\ldots,i_2,j\neq i_1} r_j \\ &\geq \frac{s}{e} \cdot \lfloor \frac{l-1}{\log_2 s} \rfloor = \Omega\left(\frac{ls}{\log_2 s}\right)\end{aligned}$$

Again, we can remove r_{i_1}, r_{i_2} and apply Lemma 2.5 on the remaining sequence. Note that for $r_{i_{l'}}$ we get

$$(1 + \frac{s}{r_{i_{l'}}}) \sum_{j=1}^{i_{l'}} r_j \geq \frac{s}{e} \cdot \lfloor \frac{l-l'}{\log_2 s} \rfloor = \Omega\left(\frac{ls}{\log_2 s}\right).$$

□

Theorem 2.7. *Network's lifetime is* $\mathcal{O}\left(\frac{E_{init} \cdot \log^2 t}{mt}\right)$, *irrespective of the number of nodes, the node placement, and the routing algorithm used.*

Proof. The theorem holds when $t = 2^{\Omega(m)}$, because lifetime of every node is $\mathcal{O}\left(\frac{E_{init}}{t}\right)$. Therefore, in the following, we assume that $t = 2^{o(m)}$. We implicitly use this assumption when we apply Corollary 2.6.

Let us divide the network into m' half-open intervals $I'_i = ((m'-i)R_t, (m'-i+1)R_t]$, $1 \le i \le m'$, of size R_t. Since $R_s = \Omega(R_t)$, by definition of m, we get $m' = \theta(m)$, where m is the parameter defined in Section 2.3.1. Without loss of generality, we assume that $m' = 8k$ for some positive integer k. Note that for any $1 \le i \le m'$, irrespective of the routing algorithm used, every packet generated by a node in an interval I'_j, $j < i$, is relayed by at least one node in I'_i, and overheard by all the nodes in I'_i. Let $\bar{n}_i$ denote the average number of alive nodes in the ith interval during the network's lifetime. Suppose $\sum_{i=1}^{8k} \bar{n}_i \ge m't = 8kt$. Consider a node u in interval I'_{8k}, which has been alive during the whole network's lifetime. Node u overhears at least $\sum_{i=1}^{8k} \bar{n}_i - 1$ packets at each round during the network's lifetime as every packet reached by the base station will be reached by u at least once. Therefore u will die in $\mathcal{O}\left(\frac{E_{init}}{m't}\right)$, hence the network's lifetime will be $\mathcal{O}\left(\frac{E_{init}}{m't}\right)$ in this case, which proves the theorem. Therefore, in the remainder of the proof, we assume that

$$\sum_{i=1}^{8k} \bar{n}_i < 8kt. \tag{2.6}$$

Consider the set of $4k$ intervals whose indices are larger than $4k$. This set must have a subset $\mathcal{I}$ of size $2k$ such that the average number of alive nodes during the network's lifetime for any interval in $\mathcal{I}$ is at most $4t$. It is because, by (2.6), the number of intervals in $\mathcal{I}$ with average number of alive nodes higher than $4t$ cannot be more than $2k$.

Consider the second half of the network's lifetime. Note that the number of alive nodes in an interval does not increase over time. Therefore, if the average number of nodes in an interval during the whole network's lifetime is at most $4t$, then the actual number of alive nodes of that interval at the beginning of the second half of network's life time cannot be more than $8t$. We say an interval $I' \in \mathcal{I}$ is "*under pressure*" in a round in the second half of network's lifetime if the average energy consumption of nodes in I' in that round is $\Omega(\frac{kt}{\log t})$.

Let $\mathcal{I} = \{\hat{I}_1, \hat{I}_2, \ldots \hat{I}_{2k}\}$. The average energy consumption of nodes in an interval $\hat{I}_i$, $1 \leq i \leq 2k$, is at least $(1 + \frac{t}{|\hat{I}_i|}) \sum_{j=1}^{i} |\hat{I}_j|$. Therefore, by Corollary 2.6, more than half of the intervals in $\mathcal{I}$ are *under pressure* in every round. Note that the set of nodes in $\mathcal{I}$ that are *under pressure* can change from one round to another. However, since more than half are *under pressure* in every round, there must be an interval $I^* \in \mathcal{I}$ which is *under pressure* in at least half of the rounds during the second half of network's lifetime.

Suppose that the number of alive nodes of an interval I' in a round is a. If I' is *under pressure* in that round, then the fraction of the total energy that I' looses by the end of that round will be $\beta = \frac{\Omega(\frac{kt}{\log t}) \cdot a}{\mathcal{O}(E_{init} \cdot a)} = \Omega(\frac{kt}{E_{init} \log t})$.

The total energy of I^* at the beginning of the second half of network's lifetime is at most $8t \cdot E_{init}$. If I^* is *under pressure* for $\frac{1}{\beta} \cdot \log(8t)$ rounds, then its total energy will reduce to

$$(1 - \beta)^{\frac{1}{\beta} \cdot \log(8t)} (8t \cdot E_{init}) \leq e^{-\log(8t)} (8t \cdot E_{init}) = E_{init}.$$

If the total energy of I^* becomes at most E_{init}, it will be reduce by $\Omega(4k \cdot 2t)$ every round, since all the packets generated by nodes in intervals $I'_1, I'_2, \ldots, I'_{4k}$ will be relayed by at least one node in I^* (recall that a relay requires at least t units of energy). Therefore, all the nodes in I^* will die in at most $\frac{E_{init}}{\Omega(m't)} = \mathcal{O}\left(\frac{E_{init}}{m't}\right)$ rounds after the total energy of I^* becomes at most E_{init}. Since I^* is *under pressure* for at least $\frac{L}{4}$ rounds in the second half of network's lifetime, we get that

$$\begin{aligned}\frac{L}{4} &\leq \frac{1}{\beta} \cdot \log(8t) + \mathcal{O}\left(\frac{E_{init}}{mt}\right) \\ &= \frac{1}{\Omega(\frac{kt}{E_{init}\log t})} \cdot \log(8t) + \mathcal{O}\left(\frac{E_{init}}{mt}\right) \\ &= \mathcal{O}\left(\frac{E_{init}\log^2 t}{kt}\right) + \mathcal{O}\left(\frac{E_{init}}{mt}\right) \\ &= \mathcal{O}\left(\frac{E_{init} \cdot \log^2 t}{mt}\right),\end{aligned}$$

which completes the proof. □

2.5 Scenario 2: A Subset of Nodes Generates Reports

In this section, we study the case where only a subset of nodes covering the network (rather than all the nodes) are required to send report packets to the base station in every round. Note that the subset of nodes sending reports can change from one round to another.

In this model, adding extra nodes can increase the network's lifetime as extra nodes can act as extra batteries. For example, if we put two nodes next to each other, the second node can sleep until the first node dies, upon which it will wake up and take the responsibility of the first node. Therefore, we can arbitrarily increase the network's lifetime by adding extra nodes to the network. The question that we investigate here is the minimum number of nodes required to achieve a certain lifetime.

Definition 1 (L-operational network)**.** *We call a network L-operational if there are L rounds such that in every round the base station receives reports from a subset of nodes covering the whole network.*

As before, we denote the energy required for transmission, reception and overhearing by A, B, and B'. Without loss of generality, we assume that every node is located on the x-axis from coordinate 0 (where the base station is located) to coordinate l, where l is the size of network. For simplicity, we assume that $\frac{l}{2R_s} = m$ is a natural number. Following, we first establish a lower bound on the number of nodes required for the network to be L-operational. We then, propose a setting to create an L-operational network with at most a constant factor of the minimum number of nodes in the optimum solution. Note that, we do not make any assumption on the relationship between R_s and R_t.

To analyze the lower bound, we construct a 2-layer network. Nodes in the first layer are only responsible for generating and transmitting reports (not relaying other nodes' reports), and those in the second layer are only responsible for relaying reports of the nodes in the first layer. Therefore, nodes in the second layer do not generate reports. In the first layer, we put minimum number of nodes to cover the whole network for L rounds, and we use minimum number of nodes to satisfy connectivity for L rounds, in the second layer.

In the first layer, the number of nodes sending reports in each round is at least m, as this is the minimum number of nodes that can cover the network. Therefore, the total number of transmissions in L rounds in the first layer is at least mL. Since each node has enough energy for up to $T = \lfloor \frac{E_{init}}{A} \rfloor$ transmissions, we get that the number of nodes in the first layer is at least $\lceil \frac{mL}{T} \rceil$. Next Lemma, improves this lower bound to $m\lceil \frac{L}{T} \rceil$. It also provides a lower bound for the number of nodes required in the second layer, irrespective of the number of nodes used in the first layer.

Definition 2 ($Rnd(r)$, where r is a real number)**.**

$$Rnd(r) = \begin{cases} \lfloor r \rfloor & \textit{if } r - \lfloor r \rfloor \leq 0.5 \\ \lceil r \rceil & \textit{otherwise} \end{cases}$$

Let $\mathcal{L}_1(m, L, T) = m \cdot \lceil \frac{L}{T} \rceil$, $\mathcal{L}_2(l, L, T', R_s, R_t) = \sum_{i=1}^{\lfloor \frac{l}{R_t} \rfloor} \left\lceil Rnd\left(\frac{l - iR_t}{2R_s}\right) \cdot \frac{L}{T'} \right\rceil$, where $T' = \lfloor \frac{E_{init}}{A+B} \rfloor$ is the maximum number of packets a node can relay.

Lemma 2.8. *The minimum number of nodes required in layer 1 and layer 2 are* $\mathcal{L}_1(m, L, T)$*, and* $\mathcal{L}_2(l, L, T', R_s, R_t)$*, respectively.*

Proof. Let $0 < \epsilon < \frac{R_s}{m-1}$ be a real number. Consider the set of points p_i, $1 \le i \le m$, at locations $(2i-1)R_s + (i-1)\epsilon$. Note that for every $1 \le i \le m$, $0 < x_{p_i} < l$, that is every point p_i is in the network. Let $\mathcal{C}(p_i)$ be the set of nodes covering p_i. Since each node has enough power for only T transmissions, we must have $|\mathcal{C}(p_i)| \times T \ge L$, or equivalently

$$|\mathcal{C}(p_i)| \ge \lceil \frac{L}{T} \rceil. \tag{2.7}$$

It is because, otherwise, there will be a round in which no alive node covers p_i. The distance between two distinct points p_i and p_j is more than $2R_s$. Therefore, different points p_i and p_j are covered by different nodes as each node can cover an interval of size at most $2R_s$. Hence,

$$i \neq j \Longrightarrow \mathcal{C}(p_i) \cap \mathcal{C}(p_j) = \emptyset$$

Consequently, the number of nodes in the first layer is at least

$$\begin{aligned} |\cup_{i=1}^{m} \mathcal{C}(p_i)| &= \sum_{i=1}^{m} |\mathcal{C}(p_i)| \\ &\overset{\text{by (2.7)}}{\ge} m\lceil \frac{L}{T} \rceil \end{aligned}$$

Next, we investigate the minimum number of nodes required to relay all the reports transmitted by the nodes in the first layer. Consider the half open intervals $I_i = ((i-1)R_t, iR_t]$, $1 \le i \le \lfloor \frac{l}{R_t} \rfloor$. We refer to these intervals as *relaying intervals*. For any relaying interval I_i, $1 \le i \le \lfloor \frac{l}{R_t} \rfloor$, and any given round t, let $R(I_i, t)$ denote the set of nodes in the first layer that i) send reports at round t, and ii) are located at x-coordinates larger than iR_t. All the reports sent by the first layer nodes in $R(I_i, t)$ have to be relayed by at least one node in the relaying interval I_i, since the transmission range is R_t. Therefore, the number of nodes required in a relaying interval I is at least $\lceil\frac{\sum_{t=1}^{L}|R(I,t)|}{T'}\rceil$. Note that the interval $(iR_t+R_s,l]$, can only be covered by the nodes in first layer that are in the interval $(iR_t,l]$. Therefore,

$$
\begin{aligned}
|R\left(I_i,t\right)| &\geq \lceil\frac{l-iR_t-R_s}{2R_s}\rceil \\
&= Rnd\left(\frac{l-iR_t}{2R_s}\right).
\end{aligned}
$$

Consequently, in the relaying interval I_i, we need at least

$$\left\lceil\frac{1}{T'}\cdot\sum_{j=1}^{L}|R\left(I_i,t\right)|\right\rceil \geq \left\lceil\frac{L}{T'}\cdot Rnd\left(\frac{l-iR_t}{2R_s}\right)\right\rceil$$

Thus, the minimum number of relaying nodes required is at least
$\sum_{i=1}^{\lfloor\frac{l}{R_t}\rfloor}\left\lceil\frac{L}{T'}\cdot Rnd\left(\frac{l-iR_t}{2R_s}\right)\right\rceil$. □

Lemma 2.9. *The minimum number of nodes required to have an L-operation linear network of size l is:*

$$\max\left\{\mathcal{L}_1(m,L,T),\mathcal{L}_2(l,L,T,R_s,R_t)\right\}$$

Proof. This is by Lemma 2.8 and the fact that the number of nodes in each layer is a lower bound for the network to be L-operational. □

In this model, one may ignore the cost of overhearing, because i) the cost of overhearing does not increase as extra nodes are added, since extra nodes act as extra batteries, ii) if nodes are loosely synchronized and each round is large enough, one can use a simple time division multiple access scenario to avoid overhearing at nodes in the second layer. Note that nodes in the first layer do not overhear as they are not required to receive any packet. If we ignore the cost of overhearing, the following theorem shows that an L-operation network with at most twice the number of nodes required in the optimum solution can be theoretically achieved.

Theorem 2.10. *There is an L-operational line network with at most $\mathcal{L}_1(m,L,T)+\mathcal{L}_2(l,L,T',R_s,R_t)$ nodes.*

Proof. We do this by putting $\lceil \frac{L}{T} \rceil$ reporting nodes at each x-coordinate $(2i-1)R_s$, $1 \leq i \leq m$, and $\left\lceil \frac{L}{T'} \cdot Rnd\left(\frac{l-iR_t}{2R_s}\right)\right\rceil$, at the highest coordinate of each relaying interval I_i, $1 \leq i \leq \lfloor \frac{l}{R_t} \rfloor$.

It is somewhat straightforward to verify that, in the first layer, there will be at least one alive node in all L rounds, so the network satisfies the coverage condition. Also, the number of reports sent by reporting nodes at one side (the side where the base station is not located) of a relaying intervals I divided by T' is not more than the number of relaying nodes placed in I, thus we will have at least one alive in each relaying interval for at least L rounds. □

When the cost of overhearing cannot be ignored, more nodes are required to be added to the network to compensate the energy consumption due to overhearing. Note that the nodes distribution in layer 1 can remain the same as before since nodes in the first layer do not require to receive/overhear other nodes' packets. However, the nodes in layer 2 placed in equal R_t distances can consume energy for overhearing. In particular, nodes placed in x-coordinate iR_t are required to receive and relay all packets from nodes located at x-coordinates larger than iR_t, and also can overhear the packets sent by nodes located at x-coordinates from iR_t to $(i-1)R_t$. Thus, the number of packets a node in the second layer can overhear is $\mathcal{O}\left(\frac{R_t}{R_s}\right)$ which is in the order of the number of packets that these nodes have to relay. Therefore, assuming that the cost of overhearing is not more than the cost of receiving, we only need to increase the number of nodes in layer 2 by at most a constant factor. Therefore, the proposed method is asymptotically optimum even if we take the cost of overhearing into account.

Special Case: $R_t = 2R_s$

In this case, we can divide the network into intervals of size $2R_s$, and put one node in the centre of each interval. This basic method provides a $\lfloor \frac{T'}{m} \rfloor$-operational network. If we have $2m$ nodes, i.e., twice as many nodes used in the basic method, we can put two nodes in each interval constructing two networks (each with m nodes). This way, the lifetime increases by a factor of two since the second network can replace the first one when it becomes disconnected. However, we can increase the lifetime by a factor of two with less than $2m$ nodes. Next theorem is about the minimum number of nodes needed to set the lifetime of network to L.

Lemma 2.11. *Suppose $R_t = 2R_s$. To make a line network L-operational, it is necessary and sufficient to have $\sum_{i=1}^{m} \left\lceil \frac{i \times L}{T'} \right\rceil$ nodes.*

Proof. To prove the theorem, let us divide the the network into intervals of size R_t. Let m be the number of intervals and assume that the first interval is the farthest one from the base station (and the mth interval is the closest one to the base station). Since the nodes in the ith interval, $1 \leq i \leq m$, have to receive $i-1$ and transmit i packets in each round, and each node has enough energy for relaying at most T' packets, the ith interval should contain at least $\left\lceil \frac{i \times L}{T'} \right\rceil$ nodes; otherwise all the nodes in the ith interval will die in less than L rounds making the network disconnected. For the sufficient condition, note that if, for every $1 \leq i \leq m$, we put $\left\lceil \frac{i \times L}{T'} \right\rceil$ nodes in the centre of ith interval, there will be at least one alive node in each interval for L rounds as the nodes in each interval have enough energy to transmit/relay all the messages during the L rounds. □

Corollary 2.12. *To make a line network T'-operational, we need exactly $\frac{m(m+1)}{2}$ nodes.*

Proof. By Lemma 2.11, we have

$$\begin{aligned}\sum_{i=1}^{m} \left\lceil \frac{i \times L}{T'} \right\rceil &= \sum_{i=1}^{m} \left\lceil \frac{i \times T'}{T'} \right\rceil \\ &= \sum_{i=1}^{m} i \\ &= \frac{m(m+1)}{2}.\end{aligned}$$

□

Corollary 2.13. *The lifetime of the basic method can be increased by a factor of F, $F = a \times m$ where $a \in N$, if we have $\frac{am(m+1)}{2}$ nodes.*

Proof. By Lemma 2.11, we get

$$\begin{aligned}\sum_{i=1}^{m}\left\lceil\frac{i \times L}{T'}\right\rceil &= \sum_{i=1}^{m}\left\lceil\frac{i \times am\lfloor\frac{T'}{m}\rfloor}{T'}\right\rceil \\ &\leq \sum_{i=1}^{m}\left\lceil\frac{am \times i}{m}\right\rceil \\ &= \sum_{i=1}^{m}\lceil a \times i\rceil \\ &= a\sum_{i=1}^{m} i \\ &= \frac{am(m+1)}{2}.\end{aligned}$$

□

Theorem 2.14. *The lifetime of the basic method can be increased by a factor of F, $F = a \times m + b$ where $a \in N$ and $0 < b < m$, if we have less than $\frac{m(F+a+1)}{2} + b - 1 - \lfloor\frac{-m}{b}\rfloor(\lceil b\rceil - b)$ nodes.*

Proof. By Lemma 2.11, the number of nodes needed to have $L = F\lfloor\frac{T'}{m}\rfloor$ (i.e., F times the lifetime of the basic method) is at most

$$\begin{aligned}\sum_{i=1}^{m}\left\lceil\frac{i \times L}{T'}\right\rceil &= \sum_{i=1}^{m}\left\lceil\frac{i \times F\lfloor\frac{T'}{m}\rfloor}{T'}\right\rceil \\ &\leq \sum_{i=1}^{m}\left\lceil\frac{F \times i}{m}\right\rceil \\ &= \sum_{i=1}^{m}\left\lceil\frac{(a \times m + b) \times i}{m}\right\rceil \\ &= a\sum_{i=1}^{m} i + \sum_{i=1}^{m}\left\lceil\frac{b \times i}{m}\right\rceil \\ &= \frac{am(m+1)}{2} + \sum_{i=1}^{m}\left\lceil\frac{b \times i}{m}\right\rceil.\end{aligned}$$

Now, we calculate the second part, we have

$$\sum_{i=1}^{m} \left\lceil \frac{b \times i}{m} \right\rceil = \sum_{i=1}^{\lfloor \frac{m}{b} \rfloor} 1 + \sum_{i=\lfloor \frac{m}{b} \rfloor + 1}^{\lfloor \frac{2m}{b} \rfloor} 2 + ... + \sum_{i=\lfloor \frac{(b-1)m}{b} \rfloor + 1}^{m} \lceil b \rceil$$

$$= \sum_{i=1}^{\lfloor \frac{m}{b} \rfloor} 1 + \sum_{i=\lfloor \frac{m}{b} \rfloor + 1}^{\lfloor \frac{2m}{b} \rfloor} 2 + ... + \sum_{i=\lfloor \frac{(b-1)m}{b} \rfloor + 1}^{m} b$$

$$+ \sum_{i=\lfloor \frac{(b-1)m}{b} \rfloor + 1}^{m} (\lceil b \rceil - b)$$

$$= \lfloor \frac{m}{b} \rfloor + 2 \left(\lfloor \frac{2m}{b} \rfloor - \lfloor \frac{m}{b} \rfloor \right) + ...$$

$$+ b \left(m - \lfloor \frac{(b-1)m}{b} \rfloor \right)$$

$$+ \sum_{i=\lfloor \frac{(b-1)m}{b} \rfloor + 1}^{m} (\lceil b \rceil - b)$$

$$= \sum_{i=1}^{b} i \left(\lfloor \frac{im}{b} \rfloor - \lfloor \frac{(i-1)m}{b} \rfloor \right)$$

$$+ \sum_{i=\lfloor \frac{(b-1)m}{b} \rfloor + 1}^{m} (\lceil b \rceil - b)$$

$$= \sum_{i=1}^{b} i \lfloor \frac{im}{b} \rfloor - \sum_{i=1}^{b-1} (i+1) \lfloor \frac{im}{b} \rfloor$$

$$+ \sum_{i=\lfloor \frac{(b-1)m}{b} \rfloor + 1}^{m} (\lceil b \rceil - b)$$

$$= \sum_{i=1}^{b} i \lfloor \frac{im}{b} \rfloor - \sum_{i=1}^{b-1} i \lfloor \frac{im}{b} \rfloor$$

$$- \sum_{i=1}^{b-1} \lfloor \frac{im}{b} \rfloor + \sum_{i=\lfloor \frac{(b-1)m}{b} \rfloor + 1}^{m} (\lceil b \rceil - b)$$

$$= b \times m - \sum_{i=1}^{b-1} \lfloor \frac{im}{b} \rfloor + \sum_{i=\lfloor \frac{(b-1)m}{b} \rfloor + 1}^{m} (\lceil b \rceil - b)$$

$$< b \times m - \sum_{i=1}^{b-1} \left(\frac{im}{b} - 1 \right)$$

$$+ \left(m - \lfloor \frac{(b-1)m}{b} \rfloor \right) (\lceil b \rceil - b)$$

$$
\begin{aligned}
&= b \times m + b - 1 - \sum_{i=1}^{b-1} \frac{im}{b} - \lfloor \frac{-m}{b} \rfloor (\lceil b \rceil - b) \\
&= b \times m + b - 1 - \frac{m}{b} \sum_{i=1}^{b-1} i - \lfloor \frac{-m}{b} \rfloor (\lceil b \rceil - b) \\
&= b \times m + b - 1 - \frac{m}{b} \times \frac{b(b-1)}{2} \\
&- \lfloor \frac{-m}{b} \rfloor (\lceil b \rceil - b) \\
&= b \times m + b - 1 - m \times \frac{(b-1)}{2} \\
&- \lfloor \frac{-m}{b} \rfloor (\lceil b \rceil - b) \\
&= \frac{m(b+1)}{2} + b - 1 - \lfloor \frac{-m}{b} \rfloor (\lceil b \rceil - b) .
\end{aligned} \tag{2.8}
$$

As a result, the total number of nodes is

$$
\begin{aligned}
\sum_{i=1}^{m} \left\lceil \frac{i \times L}{T'} \right\rceil &\leq \frac{am(m+1)}{2} + \sum_{i=1}^{m} \left\lceil \frac{b \times i}{m} \right\rceil \\
&< \frac{am(m+1)}{2} + \frac{m(b+1)}{2} + b \\
&- 1 - \lfloor \frac{-m}{b} \rfloor (\lceil b \rceil - b) \\
&= \frac{m(F+a+1)}{2} + b - 1 - \lfloor \frac{-m}{b} \rfloor (\lceil b \rceil - b) .
\end{aligned}
$$

□

Corollary 2.15. *The lifetime of the basic method can be increased by a factor of F, $F = a \times m + b$ where $a, b \in N$ and $0 < b < m$, if we have less than $\frac{m(F+a+1)}{2} + b - 1$ nodes.*

Corollary 2.16. *The lifetime of the basic method can be increased by a factor of two if we increase the number of nodes by about 50%.*

Proof. By Theorem 2.14, we have

$$
\begin{aligned}
\sum_{i=1}^{m} \left\lceil \frac{i \times L}{T'} \right\rceil &= \sum_{i=1}^{m} \left\lceil \frac{i \times 2 \lfloor \frac{T'}{m} \rfloor}{T'} \right\rceil \\
&< 1.5m + 1.
\end{aligned}
$$

□

2.6 Simulation Results

The analysis of our proposed method for the first scenario is asymptotic. Specifically, we showed that our proposed method can gain $\Omega(\log t)$ improvement over the basic method, where $t = \frac{A}{B'}$. To understand the range of constants missed in the asymptotic analysis we performed a simulation using a code written in JAVA. We divided network into intervals of size equal to the transmission range. For the basic method we put one node in each interval (the end with higher coordinate). The number of nodes in each interval for the proposed method was determined by Algorithm 1. The routing algorithm for both methods selects a node from the next interval uniformly at random to relay a packet.

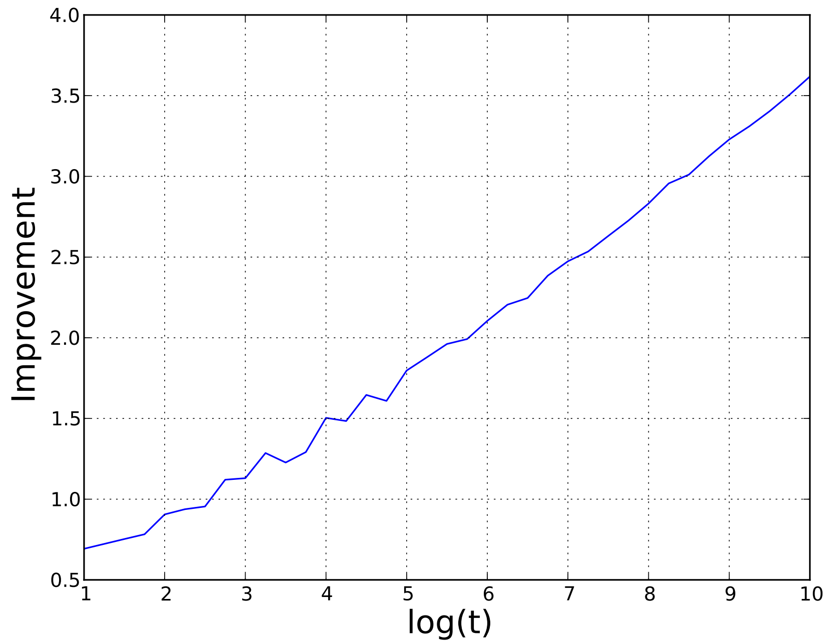
In both cases, we assume that network's lifetime ends as soon as a node dies. Note that, although this holds for the basic method, in our method, the network can still satisfy the coverage and connectivity requirements after a few nodes die. However, when a node dies, we require some coordinations between nodes to inform others to avoid further transmissions to that node. Also, the packets in the buffer of the node that runs out of battery have to be discovered and relayed by other nodes, which is challenging in a distributed setting. We used two different values for m, namely $m = 10$ and $m = 15$. We set $E_{init} = 1J$, $R_t = d = 70m$, and $n = 2$. Thus, $A = 99nJ$, $B = 50nJ$, and $B' = \frac{A}{t}$ as defined in Section 2.1.1.

In Figure 2.1 part (a), the logarithmic diagram of lifetime improvement of our proposed method to the basic method is shown for $m = 10$. Since the improvement of our solution is optimum for $t > 2^m$ according to analytical results, Figure 2.1 part (a) only shows the improvement for $t < 2^{10}$ or $\log(t) < 10$. The slope of the curve indicates about $\frac{1}{3}\log(t)$ improvement, which is in line with the analytical results. To obtain Figure 2.1 part (b), we set $m = 15$. Similar to Figure 2.1 part (a) we observe about $\frac{1}{3}\log(t)$ improvement.

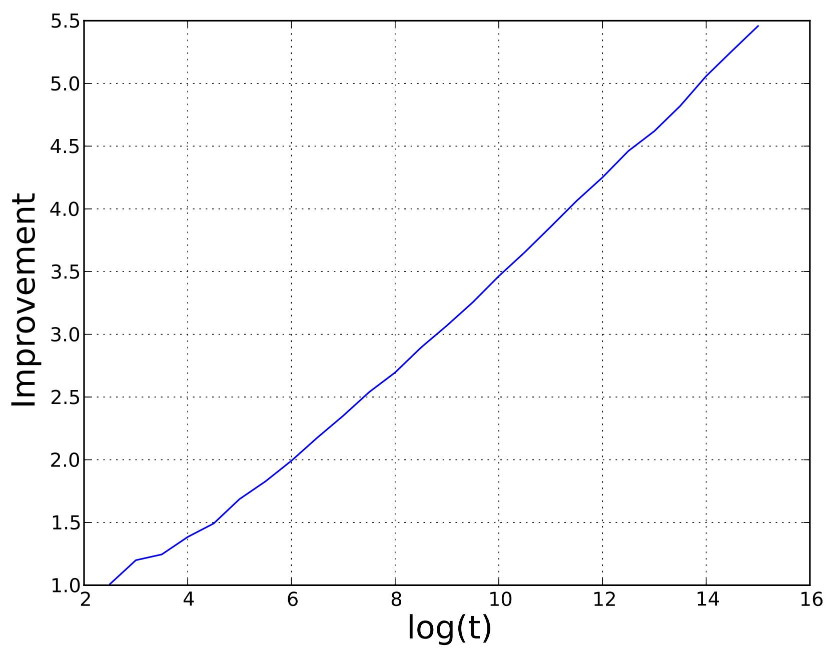
2.7 Conclusion

In this chapter, for the first scenario, we showed that, for large networks, there is no practical solution to the energy hole problem through adding extra nodes, and careful node placement and routing. On one hand, we proved that to increase network's lifetime, the cost of overhearing must be reduced significantly. On the other hand, we showed that, even when there is no overhearing, to achieve optimum lifetime, the number of nodes required (hence the delay) grows exponentially. However, we proposed a simple node placement and routing algorithm that can almost optimize network's lifetime in an ideal situation. Our main objective in introducing the proposed method is to show that it is theoretically possible to almost match the upper bound proven. Nevertheless, the proposed method may be used in small networks, or within clusters in a large network, particularly when network is delay tolerant.

For the second scenario, we derived a lower bound on the minimum number of nodes needed to achieve a certain lifetime, as in this case, network's lifetime can unboundedly grow with the number of nodes in the network.



(a)



(b)

Figure 2.1: Logarithmic diagram of lifetime improvement for (a) $m = 10$ and (b) $m = 15$

Chapter 3

Two-dimensional Wireless Sensor Networks

In this chapter, we study the lifetime of two-dimensional networks in convergecast applications, where every node sends packets to a base station, periodically. The main focus of this chapter is to study the effect of multi-hop routing algorithms on network's lifetime, and to propose new ones. We also consider packet delay as our secondary objective when designing routing algorithms.

We explain the related background in Section 3.1. Then, we review the existing literature related to our problem in Section 3.2. In Section 3.3, we specify our assumptions and define the problem studied in this chapter. Next, we propose our routing algorithm in Section 3.4. We conclude the chapter by presenting our simulation results in Section 3.5.

3.1 Background

3.1.1 Perfect Aggregation Convergecast

In aggregation convergecast applications, each node in the network aggregates all the packets it has received and its own packet into one packet, and sends it to its next hop according to the routing algorithm. The goal of the routing is to deliver every packet to the *base station*. The case where the aggregated packets have the same size as each received packet is called *perfect aggregation* in the literature [30]. Examples of perfect aggregation include applications in which the sum, minimum, maximum, average, or count needs to be calculated. Data aggregation prevents transmitting redundant data that impose heavier traffic on the final nodes that are close to the base station.

For a wireless sensor network, we define a communication graph $G(V, E)$ where V represents the set of nodes, and for every pair of nodes $u, v \in V$, there is an edge in E if and only if u and v are both in the communication range of each other. A spanning tree of G is simply a tree $T(V, E')$ with $E' \subseteq E$. In aggregation convergecast applications, a routing algorithm can be represented by a spanning tree of the communication graph, rooted at the base station. Given the spanning tree, each node can send its packet to its parent, who in turn aggregates all the received packets from its children into one and sends it to its own parent.

Another component important in convergecast applications is *scheduling*, which determines the sets of nodes that can transmit simultaneously. Collision free scheduling is a spacial kind of scheduling in which the set of nodes transmitting simultaneously do not cause any collision at any of the transmitters' parents. Next, we explain collision free scheduling in more details, and briefly describe some of the existing collision free scheduling methods that we will use in this work.

3.1.2 Centralized Collision Free Scheduling

In wireless sensor networks, when a node transmits a packet, the packet reaches all its neighbours, i.e. nodes that are within the sender's transmission range. If a node is simultaneously reached by more than one packet, a collision occurs. We assume that collided packets are not received.

In a centralized collision free scheduling, time is divided into slots. In each slot, a subset of nodes are scheduled to transmit. Every node scheduled to transmit in any given slot has the following properties: 1) it is a leaf node in the routing tree, or it has already received the packets from all its children; 2) its transmission does not cause collision at parent of any other node scheduled in that slot.

A scheduling round is defined as a sequence of consecutive time slots in which every node is scheduled exactly in one slot. If a scheduling has both of the aforementioned properties, the base station will receive all the packets in one scheduling round. Note that in a centralized scheduling, every node only needs to wake up in the slots when it is supposed to receive or transmit a packet. Therefore, the number of slots in which a node is required to be awake is equal to the number of its children plus one (the extra slot is for the node to transmit its own packet).

In this work, we use two existing collision free scheduling algorithms, First-fit [28] and WIRES [25], to evaluate the delay of our proposed routing algorithms against some of the existing routing algorithms.

First-fit Scheduling

First-fit scheduling maintains an eligible set, E, that contains all the nodes that are ready to be scheduled, i.e., that have not been scheduled yet but all their children have been scheduled before. It iteratively assigns a collision free group of nodes from E to each time slot, and updates E for the next iteration. Initially, the eligible set contains only the leaf nodes. In each iteration, the algorithm builds a set X of the nodes to be scheduled in the next time slot. Initially X is empty. In each step, the algorithm selects a node from E that does not cause collision at parents of the nodes already in X, and moves to X. This process is repeated until all the nodes that remain in E cause collision with the parent of at least one node in X. Then, the algorithm assigns all the nodes in X to the next available time slot, updates E by adding any new nodes that become ready to be scheduled, and proceeds to the next iteration.

WIRES

Another collision free scheduling method that we use is Weighted Incremental Ranking for convergEcast with aggregation Scheduling (WIRES). WIRES works similar to the First-fit algorithm. However, it tries to reduce the delay by selecting the nodes from the eligible set in an order that helps reducing the number of slots in a scheduling round. To achieve this, before starting each iteration, the nodes in E are sorted in decreasing order by their number of unscheduled neighbours that are not in E. Then nodes form E are considered to be moved to X starting from the first node of E (i.e., the node with the maximum number of unscheduled neighbours that are not in E). The authors showed that WIRES outperforms several existing scheduling methods through extensive simulations.

3.2 Related Work

Both lifetime improvement and delay reduction problems have been studied extensively for two-dimensional WSNs. However, for aggregation convergecast, only a small subset of the existing works try to address both problems simultaneously. In this section, we review the literature on lifetime improvement, and then look at some works that try to address both lifetime improvement and delay reduction.

The authors of [30] studied lifetime improvement problem for the perfect aggregation convergecast problem and proposed a distributed and dynamic routing approach, which they call localized power-efficient data aggregation protocol (L-PEDAP). They assume that nodes can adjust their transmission range to save energy. Each edge (u,v) in the communication graph is assigned a cost that is the total energy consumed in u (sender) and v (receiver), so each node can build a localized sparse topology based on these costs and its local information. Then these localized sparse topologies are used to build a routing tree in a distributed manner. The authors investigated three different criteria by which a node selects its parent: 1) the first neighbouring node; 2) the nearest neighbour with minimum hop path to the base station; 3) the neighbour that minimized the total energy consumption through the path to the base station. Through simulations, the authors study the performance of using different combinations of localized topologies and parent selection criteria.

In [35], the lifetime maximization problem was studied for the setting in which all nodes have a fixed and identical transmission power, but possibly different initial energy. The authors proposed an approximation algorithm to find a near-optimal spanning tree, but did not consider delay in their solution.

The algorithm proposed in [9] considers both throughput and delay by simultaneously optimizing the maximum degree and the radius of the routing tree. Their (α, β) approximation algorithm builds a Bounded-Degree Minimum-Radius spanning tree in which for a given degree bound Δ^*, the radius of the resulting tree is at most β times the minimum possible radius, and the maximum degree of nodes is at most $\Delta^* + \alpha$, where α and β are positive constants. To further reduce the delay, they also used multiple frequency channels.

Balanced Shortest Path Tree (BSPT), proposed in [25] is another algorithm to reduce delay and improve lifetime. The setting studied in [25] is the closest to the setting considered in this chapter. The algorithm starts with a shortest path tree (SPT) of the network's communication graph. It then iteratively balances the load on the nodes at each level of the tree by redistributing their children among themselves so that the maximum degree of the nodes is minimized. Note that this operation does not change the shortest path property of the tree.

3.3 Problem Statement

In this chapter, similar to scenario 1 in Chapter 2, we assume that all nodes are required to send report packets periodically to the base station with the help of a multihop routing. Also, we do not use any power control mechanism, and we assume that all nodes have an identical and fixed transmission power. However, unlike scenario 1 in Chapter 2, we assume that the number of nodes and their positions are fixed. Therefore, our solution here is based on routing rather than node placement.

Furthermore, we assume that nodes use perfect aggregation and send only one packet per convergecast round, which is the time needed for receiving all packets by the base station. However, the number of packets that different nodes receive can be different. The routing is represented as a spanning tree of the communication graph of the network. We assume that the communication graph is connected at the beginning, i.e., each node has a path to the base station. Lifetime is defined as the number of convergecast rounds during which the packets of all nodes are successfully received by the base station. To improve lifetime, we try to reduce the maximum degree of our spanning tree. This is because the node with maximum degree receives the most number of packets per convergecast round and its energy is depleted sooner than others, ending the lifetime of the network.

Improving lifetime by decreasing the degree of nodes can produce long paths in the network, potentially increasing the overall delay defined as the number of slots used in the scheduling round. Our proposed algorithm aims to improve lifetime while accounting for the delay. To evaluate the delay of our generated routing tree, we use two scheduling methods, namely WIRES and First-fit.

3.4 Proposed Centralized Routing Algorithm

Our routing algorithm is centralized. Note that for WSNs, distributed algorithms are more practical. We construct our routing tree in two phases. In the first phase, given any spanning tree of the communication graph, we try to reduce its maximum degree. In the second phase, we aim at reducing the overall delay while preserving the lifetime. Since these phases work independently, each of them can be replaced by any other algorithm sharing the same objective. The next two sections explain these two phases.

3.4.1 Phase 1: Lifetime Improvement

Since the network's lifetime ends by the first node's death, the maximum degree of nodes in the routing tree determines the whole lifetime. In general, finding a Minimum Degree Spanning Tree (MDST) of a graph, G, in which the maximum degree of nodes is minimum among all spanning trees of G, is shown to be NP-hard [7]. Therefore, to reduce the maximum degree we use an approximation algorithm proposed in [7], which constructs a tree with maximum degree of at most $\mathcal{O}\left(\Delta^* + \log n\right)$, where Δ^* is the maximum degree of the optimal solution, and n is the number of nodes.

The algorithm proposed in [7] starts from any spanning tree T of a graph G, and iteratively replaces some edges of T with some edges of G. Let $d(k)$ denote the degree of vertex k in T, and C denote the unique cycle generated by adding an edge (u, v) to T. Suppose there is a vertex z in C that $d(z) \geq \max\{d(u), d(v)\} + 2$. The operation of adding (u, v) to T and removing one of the edges in C incident to z is called an "improvement". By this improvement, $d(z)$ decreases by one and neither $d(u)$ nor $d(v)$ will be equal to or greater than $d(z)$. The algorithm continues these improvements until there is no edge in G that causes an improvement in T.

In the next section, we propose an algorithm that iteratively refines the resulting tree of this phase to decrease its delay.

3.4.2 Phase 2: Delay Reduction

The basic idea of our delay reduction algorithm is to reduce the height of the leaf nodes while preserving the maximum degree. While we use single channel scheduling in this chapter, having a tree with smaller height is potentially useful for decreasing delay in the settings that deploy multichannel scheduling (e.g., the setting used in [9]).

Given a spanning tree T of the communication graph, our algorithm iteratively tries to reduce the height of a leaf node l in the tree by finding a new parent with smaller height, either for l or one of its ancestors in T. We call this process a *height reduction*. A height reduction operation is called *eligible* if it does not increase the maximum degree of T. By only applying eligible height reductions, our algorithm guarantees that we do not sacrifice the lifetime to reduce the delay. Note that in general, the degree of the base station does not affect the lifetime, since it typically has access to an unlimited source of energy. However, the degree of the base station can become a bottle-neck for the delay, since its children must be scheduled in different time slots to avoid collisions. Hence, our algorithm iteratively checks the degree of the base station against the height of the tree (which is a lower bound on the number of required time slots). In particular, in each round of our algorithm, we apply eligible height reduction operations that increase the degree of the base station at most by one. Then, at the end of the round, we check the degree of the base station, and the algorithm continues only if this degree is less than the height of the tree. The psudo code of this algorithm is presented in Algorithm 2.

Algorithm 2 Delay Reduction

```
Input: G (communication graph), T (any spanning tree of G).
repeat
    Let d denote the current degree of the base station.
    while there exists an eligible height reduction operation in T such that after
    applying it, the degree of base station is at most d + 1 do
        Update T by applying that eligible height reduction.
    end while
until no eligible height reduction operations remain, or the degree of the base
station gets larger than the height of T.
```

Time Complexity of the Delay Reduction Algorithm

In our algorithm, the total number of height reduction operations is $\mathcal{O}(n^2)$, since i) the number of leaf nodes is O(n), ii) the height of every leaf is O(n), and iii) the high of a leaf is decremented in each operation. Now, we calculate the time complexity of each height reduction operation. For each heigh reduction operation, the algorithm iterates through all the leaves to find a leaf node whose heigh can be decreased. To know that the height of a leaf node can be decreased, the algorithm does $\mathcal{O}(n^2)$ iterations, since, in the worst case, for each leaf, the algorithm checks the possibility of height reduction by visiting all the neighbours of every ancestor of the leaf node. As a result, the time complexity of our algorithm is $\mathcal{O}(n^2) \times \mathcal{O}(n^3)$ or $\mathcal{O}(n^5)$.

3.5 Simulation Results

In this section, we present simulation results to demonstrate the effectiveness of our method. We call the routing tree generated by our method Height-reduced approximated minimum Degree Spanning Tree (HDST), and compare its lifetime (i.e., the maximum degree of all nodes except the base station), and its delay to four different routing trees: the Breath First Search (BFS) tree, the BSPT proposed in [25], a refined version of BSPT (referred to as $\mathtt{BSPT}_{\mathtt{ref}}$) which has a smaller delay, and the tree generated by a modified version of the algorithm of Section 3.4, which attempts to increase the lifetime in the first phase more aggressively ($\mathtt{HDST}_{\mathtt{ref}}$).

Recall that in BSPT, explained in Section 3.2, nodes in each level of the tree are spreaded among the nodes in the proceeding level, i.e., the level closer to the root. In the $\mathtt{BSPT}_{\mathtt{ref}}$ algorithm, we restrict BSPT to spread only to the non-leaf nodes. This way, the leaf nodes are not assigned any child, and can therefore be scheduled earlier, which results in a better delay compared to BSPT.

To construct the $\mathtt{HDST}_{\mathtt{ref}}$, we follow the algorithm of Section 3.4. However, in the first phase, we ignore improvement operations that reduce the degree of the base station, giving other nodes the chance to have smaller degrees. This way, it is possible to achieve a better lifetime than the initial algorithm. Note that the base station has access to an unlimited source of energy so the number of its children does not affect the lifetime of the network.

We generated connected communication graphs with 400, 500, ..., and 1000 nodes. We placed nodes in a 1×1 square, uniformly and independently at random. Finally, we simulated three different transmission ranges, i.e., 0.125, 0.15, and 0.175. The base station is placed at (0.5,0.5). We report the average lifetime and delay of the algorithms over a set of 50 randomly generated graphs.

Figure 3.1 shows the trees generated by the aforementioned routing algorithms when the number of nodes is 150, and the transmission range is 0.175. `BFS` tree has the largest maximum degree (11), while in `BSPT`, where the load in each level is balanced, the maximum degree is 6. $\texttt{BSPT}_{\texttt{ref}}$ algorithm does not assign a child to leaf nodes (in an attempt to reduce delay), and achieves the maximum degree of 9. $\texttt{HDST}_{\texttt{ref}}$ algorithm does not try to reduce the degree of root (the base station), so the degree of root in this case goes up to 15. However, the maximum degree of other nodes reduces to 2. `HDST` algorithm reduces the tree height at the expense of increasing the degree of root, but only as long as the degree of root does not become a bottle-neck in the delay itself. Hence, although `HDST` has a maximum degree of 3, its root degree is 6, smaller than its height (7). Also, it is worth mentioning that the optimal height is 6, attained by the shortest path tree algorithms (`BFS`, `BSPT`, and $\texttt{BSPT}_{\texttt{ref}}$).

Figure 3.2, 3.3, and 3.4 show the maximum degree of the five routing trees as a function of the number of nodes, for three different transmission ranges. In these three figures, `HDST` and $\texttt{HDST}_{\texttt{ref}}$ have smaller maximum degrees compared to others. As expected, $\texttt{HDST}_{\texttt{ref}}$ achieves a smaller maximum degree than `HDST`. As the transmission range increases, this difference becomes larger, since the increased number of neighbours of the base station in the communication graph gives $\texttt{HDST}_{\texttt{ref}}$ algorithm more room to decrease the degree of other nodes. The refinement in

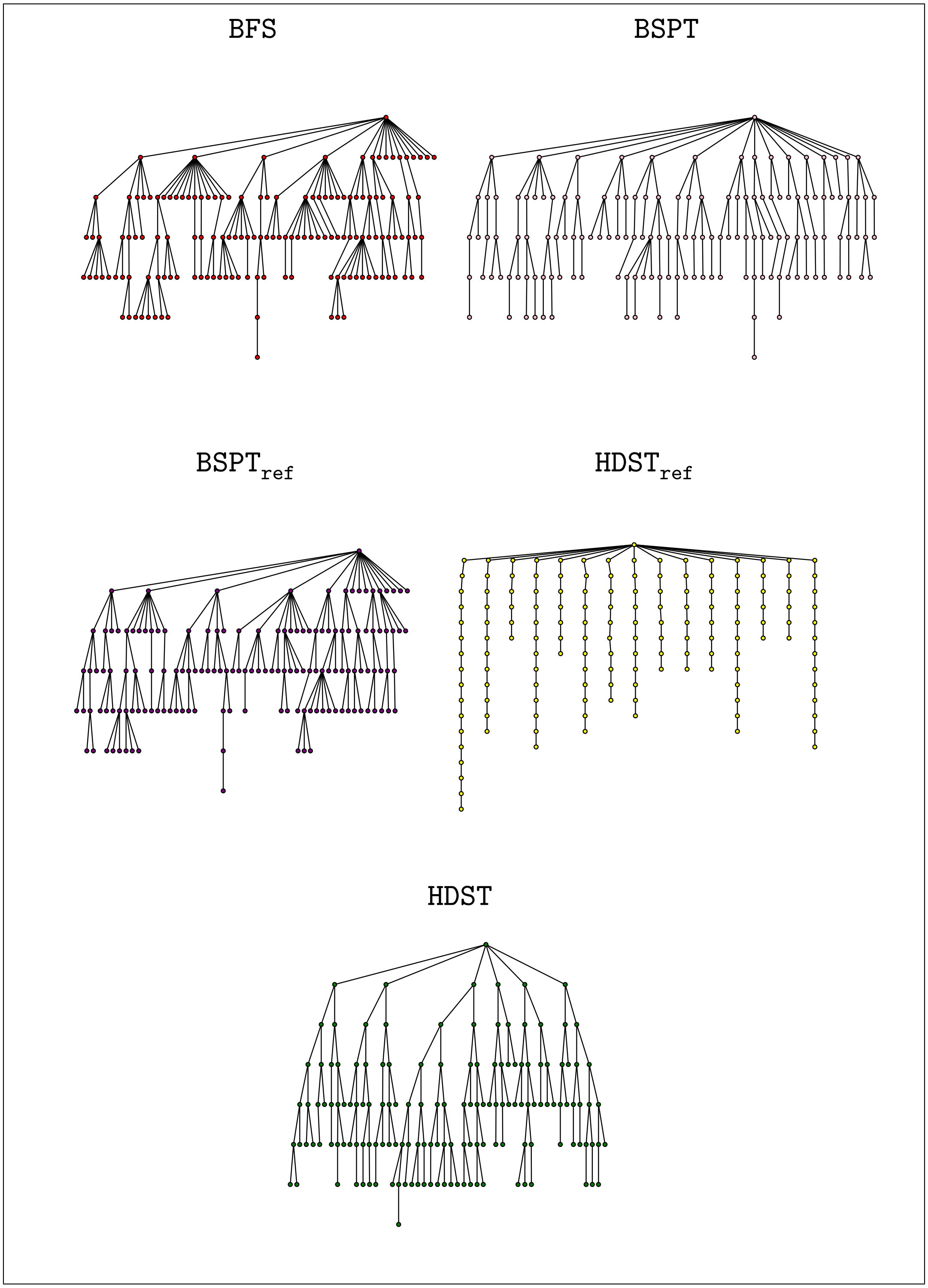


Figure 3.1: The trees generated by the five routing algorithms when the number of nodes and the transmission range are 150 and 0.175, respectively.

$BSPT_{ref}$ algorithm aimed at decreasing the delay increases the maximum degree. That is because $BSPT_{ref}$ algorithm cannot use the leaf nodes to decrease the max

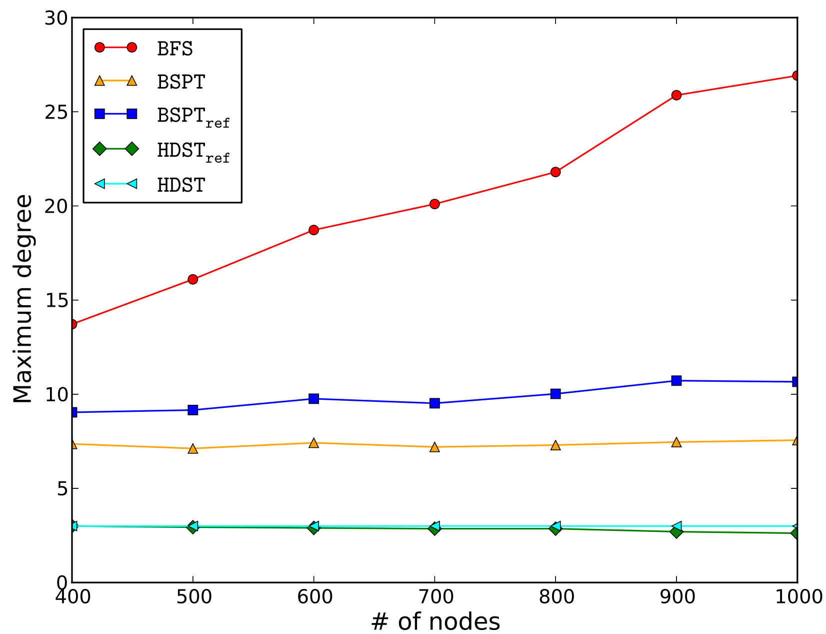


Figure 3.2: Maximum degree of the five routing trees as a function of the number of nodes. The transmission range is fixed at 0.125.

degree.

Figure 3.5, 3.6, and 3.7 show the number of slots of one scheduling round for the five routing trees, using two scheduling methods (WIRES and First-fit), for three different transmission ranges. $\mathtt{BSPT}_{\mathtt{ref}}$ has smaller delay than `BSPT`. Although `HDST` is not a shortest path tree, its delay is close or better than the other three SPTs (`BFS`, `BSPT`, and $\mathtt{BSPT}_{\mathtt{ref}}$) especially when using First-fit, which does not try to reduce delays itself. Also, as the transmission range or the number of nodes increases, this improvement becomes larger, since the number of neighbours of each node increases, giving `HDST` algorithm more chance to decrease the tree height in its second phase of construction. Besides lifetime, reducing the degree of nodes in the first phase of `HDST` algorithm helps in reducing the delay as well. This is because a node with fewer children has to wait a smaller number of slots before it can be scheduled. Results of $\mathtt{HDST}_{\mathtt{ref}}$ show that keeping the degree of the base station

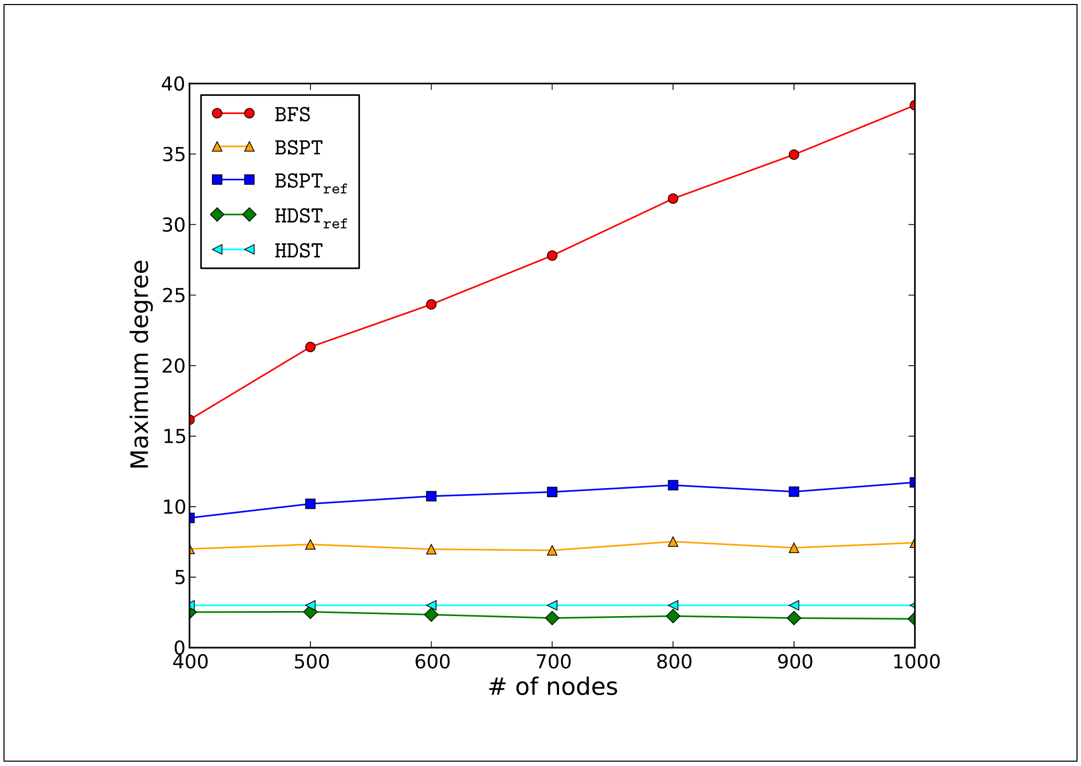


Figure 3.3: Maximum degree of the five routing trees as a function of the number of nodes. The transmission range is fixed at 0.15.

small is also important in reducing the delay.

3.5.1 Conclusion

In this chapter, we studied aggregation convergecast applications and proposed a routing algorithm to improve lifetime of the network while also taking the delay into account. The algorithm consists of two independent phases. In the first phase, to increase lifetime, we try to reduce the maximum degree of the routing tree of the network. To that end, we find an approximate minimum degree spanning tree of the communication graph of the network using an existing algorithm from the graph theory literature. Then, in the second phase, we refine the tree generated in the first phase to reduce its delay. This is achieved by iteratively decreasing the hight of leaf nodes in the tree. We call the tree generated by this algorithm `HDST`. While `HDST` takes the degree of the base station into account when building the routing tree, this

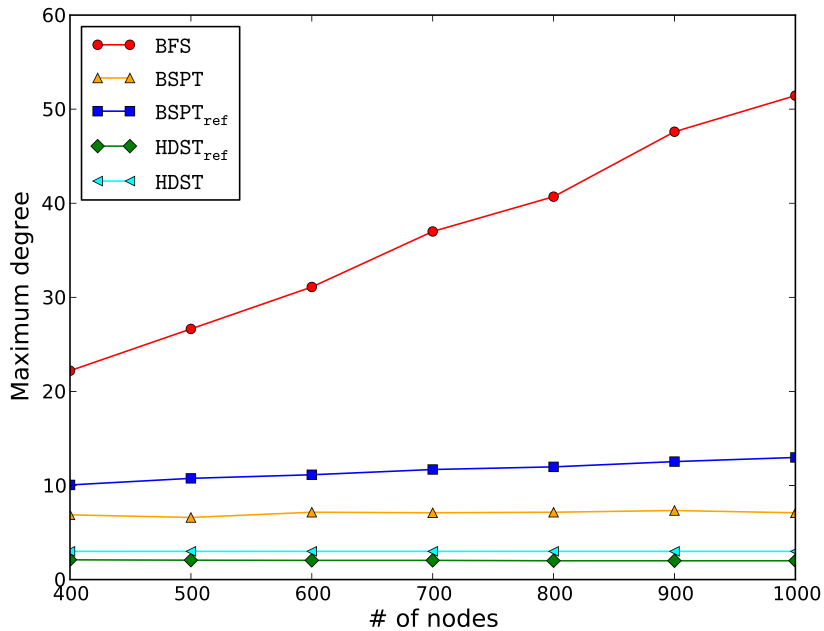


Figure 3.4: Maximum degree of the five routing trees as a function of the number of nodes. The transmission range is fixed at 0.175.

degree is usually not important from a lifetime perspective, since the base station usually has access to unlimited power. Therefore, we also created a refined version of HDST, called $\mathtt{HDST_{ref}}$, which tries to further improve the lifetime by ignoring the degree of the base station. Our simulations show that although $\mathtt{HDST_{ref}}$ has a better lifetime compared to HDST, it suffers from a larger delay. Thus, $\mathtt{HDST_{ref}}$ can be useful in applications where the lifetime is the most important constraint, while HDST is a better choice when shorter delays are required.

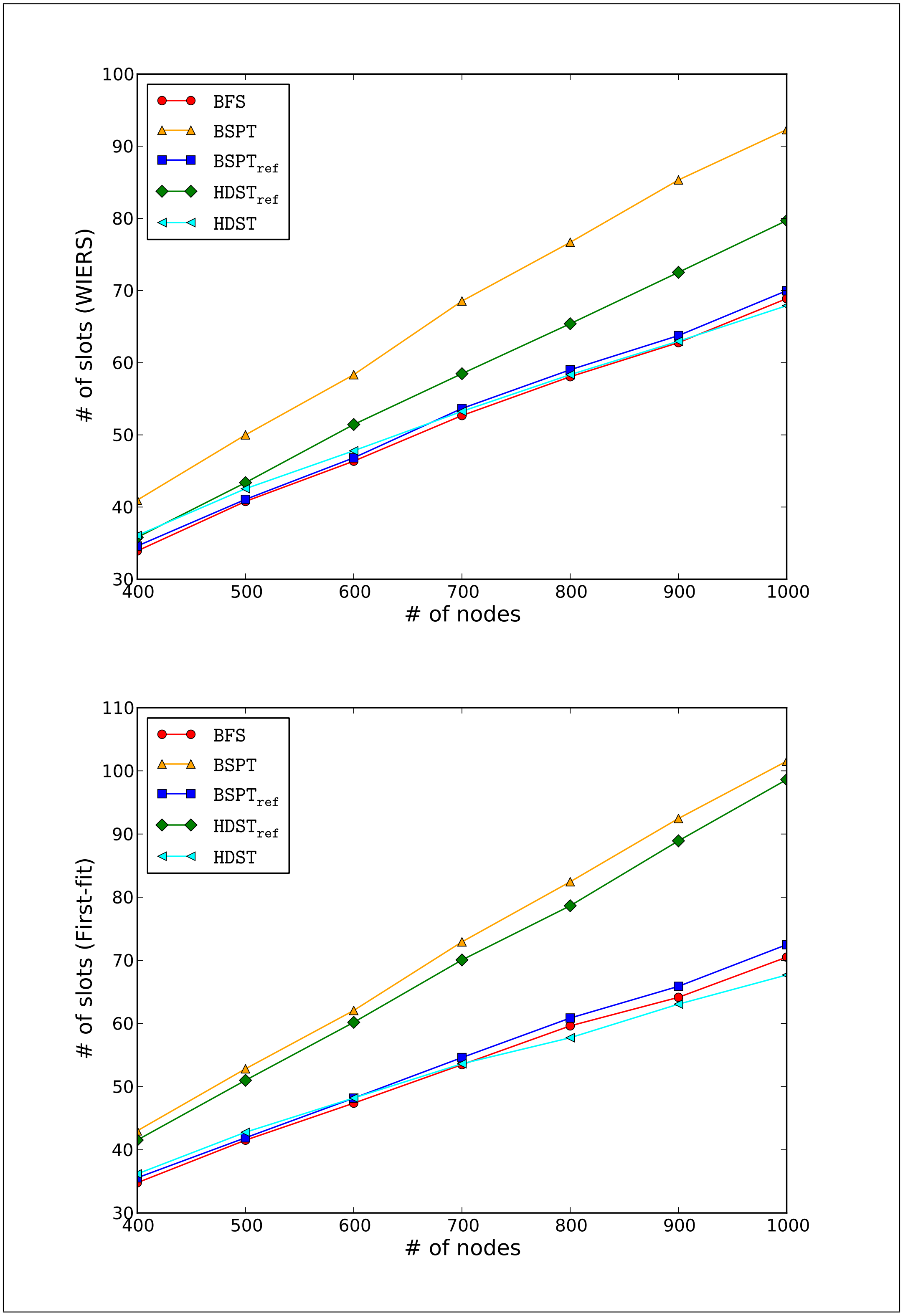


Figure 3.5: The number of slots in one scheduling round for the five routing trees, using two different scheduling algorithms. The transmission range is fixed at 0.125.

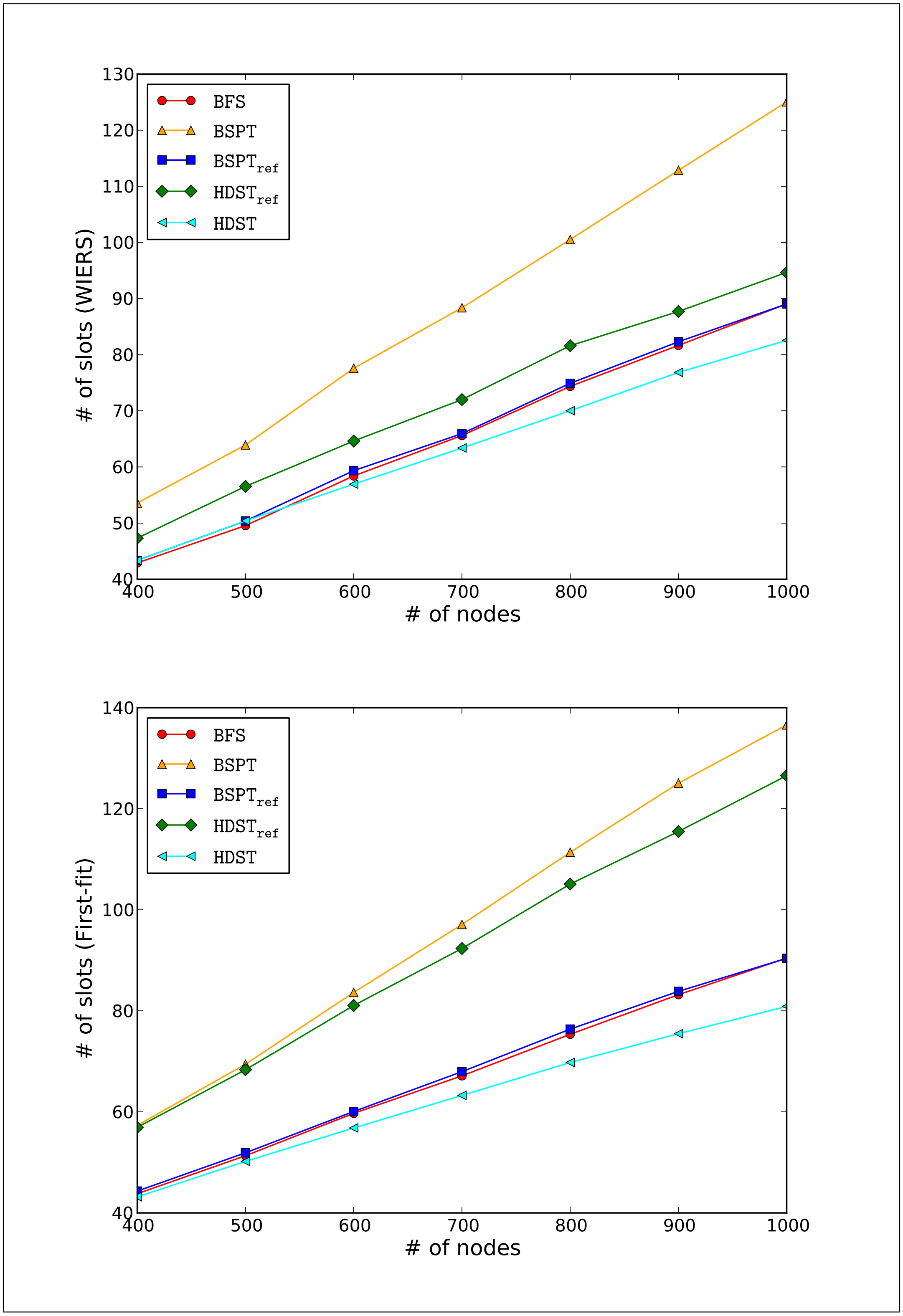


Figure 3.6: The number of slots in one scheduling round for the five routing trees, using two different scheduling algorithms. The transmission range is fixed at 0.15.

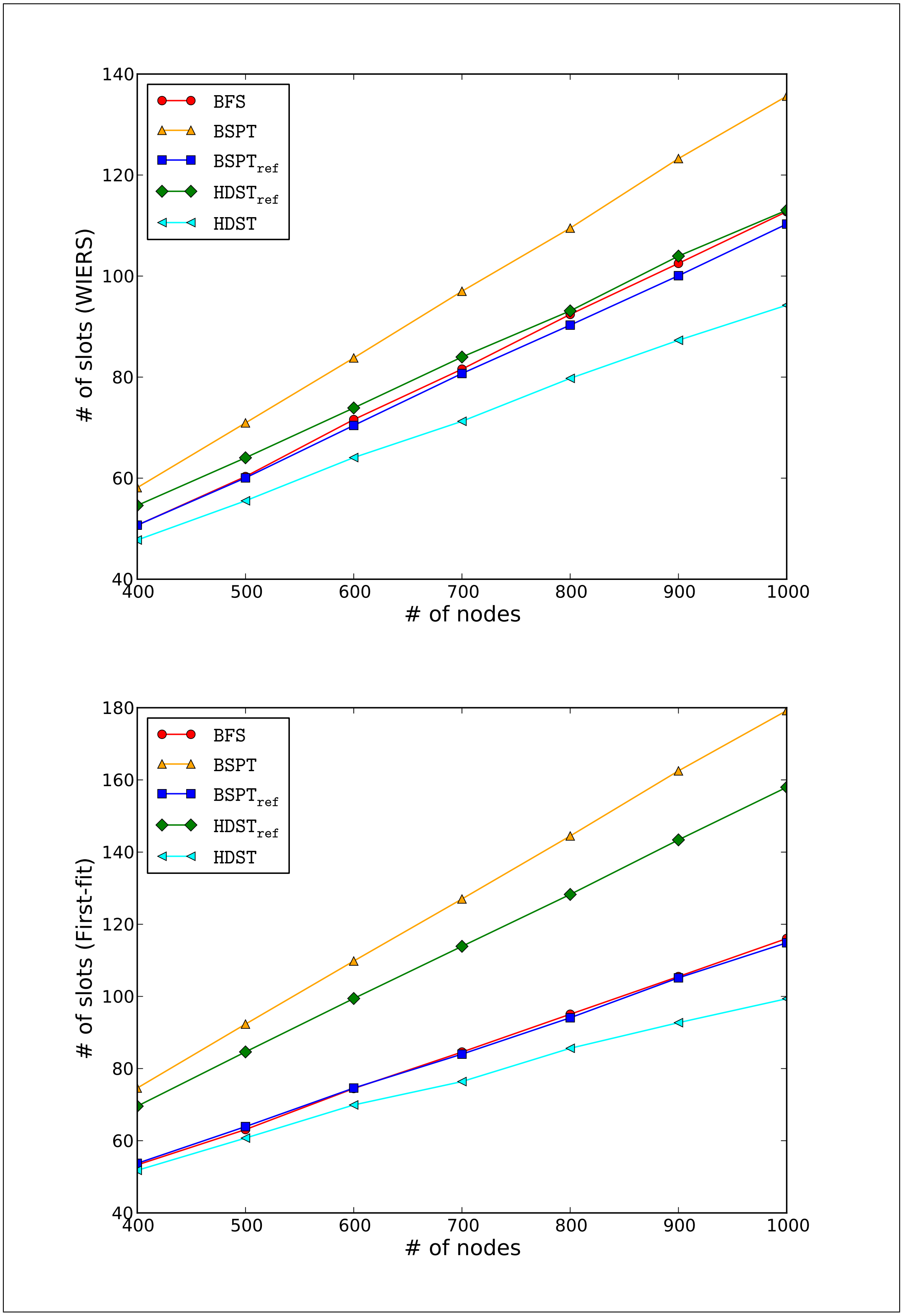


Figure 3.7: The number of slots in one scheduling round for the five routing trees, using two different scheduling algorithms. The transmission range is fixed at 0.175.

Chapter 4

Conclusion

In this thesis, we studied the lifetime of linear and two-dimensional wireless networks. For linear networks, we studied two different scenarios using methods based on node placement. In the first scenario, where every node has to periodically send reports to the base station, we showed that the network's lifetime reduces almost linearly with the size of network unless the cost of overhearing approaches zero. We also showed that, even when we can ignore the cost of overhearing, to get close the maximum lifetime possible, we have to use exponentially (in terms of the length of network) many nodes in the network. This is clearly impractical when the network is long (for example, to cover a long pipeline). However, the idea of using exponentially many nodes may be useful over small segments of network, e.g., for delivering packets generated in a segment to a local relay in a hierarchical solution. In the second scenario, we considered the case where only a subset of nodes covering the network are required to send packets to the base station. We derived a lower bound on the number of nodes required to achieve a given lifetime, and proposed a general method which is asymptotically optimal.

For two-dimensional networks with fixed node placement, we proposed a routing algorithm that improves lifetime, while accounts for network's delay. Our proposed routing algorithm works in two phases: in phase 1, we used an existing minimum degree spanning tree approximation algorithm to improve lifetime; in phase

2, we reduced the tree height to get smaller delay, while preserving the lifetime of the tree.

4.1 Future work

For linear wireless sensor networks, another interesting question is how different initial energies for the nodes influence the network's lifetime. In particular, we might assign larger initial energies to the nodes close to the base station instead of adding extra nodes to solve the energy hole problem. This method has the additional advantage of avoiding the traffic generated by the extra nodes, which is itself an energy drain. Another mechanism that might be useful for solving the energy hole problem is energy harvesting, by which nodes gain power from natural sources such as solar, wind, or thermal power.

For two-dimensional wireless sensor networks, our proposed routing algorithm works in a centralized manner. An interesting question is designing a distributed version of our routing algorithm with desirable lifetime and delay. Another direction for improving our algorithm is to use an adaptive routing scheme, which considers the residual energy of nodes when choosing the next hop at each round.

Bibliography

[1] A. Alfieri, A. Bianco, P. Brandimarte, and C. F. Chiasserini. Maximizing system lifetime in wireless sensor networks. *European Journal of Operational Research*, 181(1):390402, 2007.

[2] M. Bhardwaj, T. Garnett, and A. P. Chandrakasan. Upper bounds on the lifetime of sensor networks. In *Communications, 2001. ICC 2001. IEEE International Conference on*, volume 3, page 785790, 2001.

[3] M. Cao, L. T. Yang, X. Chen, and N. Xiong. Node placement of linear wireless multimedia sensor networks for maximum network lifetime. In *Proceedings of the 3rd international conference on Advances in grid and pervasive computing*, page 373383, 2008.

[4] C. W. Chen and Y. Wang. Chain-type wireless sensor network for monitoring long range infrastructures: Architecture and protocols. *International Journal of Distributed Sensor Networks*, 4(4):287314, 2008.

[5] P. Cheng, C. N. Chuah, and X. Liu. Energy-aware node placement in wireless sensor networks. In *Global Telecommunications Conference, 2004. GLOBECOM'04. IEEE*, volume 5, page 32103214, 2004.

[6] Q. Dong. Maximizing system lifetime in wireless sensor networks. In *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, page 1319, 2005.

[7] Martin Fürer and Balaji Raghavachari. Approximating the minimum degree spanning tree to within one from the optimal degree. In *Proceedings of the third annual ACM-SIAM symposium on Discrete algorithms*, pages 317–324. Society for Industrial and Applied Mathematics, 1992.

[8] Deepak Ganesan, Răzvan Cristescu, and Baltasar Beferull-Lozano. Power-efficient sensor placement and transmission structure for data gathering under distortion constraints. *ACM Transactions on Sensor Networks (TOSN)*, 2(2):155–181, 2006.

[9] Amitabha Ghosh, Özlem Durmaz Incel, VS Kumar, and Bhaskar Krishnamachari. Multichannel scheduling and spanning trees: throughput-delay tradeoff for fast data collection in sensor networks. *IEEE/ACM Transactions on Networking (TON)*, 19(6):1731–1744, 2011.

[10] Y. Guo, F. Kong, D. Zhu, A. Tosun, and Q. Deng. Sensor placement for lifetime maximization in monitoring oil pipelines. In *Proceedings of the 1st ACM/IEEE International Conference on Cyber-Physical Systems*, page 6168, 2010.

[11] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on*, pages 10–pp. IEEE, 2000.

[12] A. Hossain, T. Radhika, S. Chakrabarti, and P. K. Biswas. An approach to increase the lifetime of a linear array of wireless sensor nodes. *International Journal of Wireless Information Networks*, 15(2):7281, 2008.

[13] I. Howitt and J. Wang. Energy balanced chain in distributed sensor networks. In *Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE*, volume 3, page 17211726, 2004.

[14] S. Hu and M. Motani. Early overhearing avoidance in wireless sensor networks. In *Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet*, pages 26–35. 2008.

[15] B. Jang, J. B. Lim, and M. L. Sichitiu. An asynchronous scheduled mac protocol for wireless sensor networks. *Computer Networks*, 2012.

[16] I. Jawhar, N. Mohamed, and D. P. Agrawal. Linear wireless sensor networks: Classification and applications. *Journal of Network and Computer Applications*, 34(5):16711682, 2011.

[17] I. Jawhar, N. Mohamed, K. Shuaib, and N. Kesserwan. An efficient framework and networking protocol for linear wireless sensor networks. *Ad Hoc and Sensor Wireless Networks Journal, Old City Publishing, London, UK*, 2008.

[18] Sukun Kim, Shamim Pakzad, David Culler, James Demmel, Gregory Fenves, Steven Glaser, and Martin Turon. Health monitoring of civil infrastructures using wireless sensor networks. In *Information Processing in Sensor Networks, 2007. IPSN 2007. 6th International Symposium on*, pages 254–263. IEEE, 2007.

[19] H. Li and X. Shunjie. Energy-efficient node placement in linear wireless sensor networks. In *Measuring Technology and Mechatronics Automation (ICMTMA), 2010 International Conference on*, volume 2, page 104107, 2010.

[20] Jian Li and Prasant Mohapatra. An analytical model for the energy hole problem in many-to-one sensor networks. In *IEEE Vehicular Technology Conference*, volume 62, page 2721, 2005.

[21] Weifa Liang and Yuzhen Liu. Online data gathering for maximizing network lifetime in sensor networks. *Mobile Computing, IEEE Transactions on*, 6(1):2–11, 2007.

[22] Stephanie Lindsey and Cauligi S Raghavendra. Pegasis: Power-efficient gathering in sensor information systems. In *Aerospace conference proceedings, 2002. IEEE*, volume 3, pages 3–1125. IEEE, 2002.

[23] F. Lu, G. Voelker, and A. C. Snoeren. Slomo: Downclocking wifi communication. In *Proc. of USENIX conference on Networked Systems Design and Implementation*, pages 255–268, 2013.

[24] Alan Mainwaring, David Culler, Joseph Polastre, Robert Szewczyk, and John Anderson. Wireless sensor networks for habitat monitoring. In *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 88–97. ACM, 2002.

[25] Baljeet Malhotra, Ioanis Nikolaidis, and Mario A Nascimento. Aggregation convergecast scheduling in wireless sensor networks. *Wireless Networks*, 17(2):319–335, 2011.

[26] Fernando Martincic and Loren Schwiebert. Introduction to wireless sensor networking. *Handbook of Sensor Networks: Algorithms and Architectures, I. Stojmenovic, Ed. John Wiley & Sons*, pages 1–41, 2005.

[27] M. Noori and M. Ardakani. Characterizing the traffic distribution in linear wireless sensor networks. *IEEE Communications Letters*, 12(8):554 –556, August 2008.

[28] C-H Huang Scott, Wan Peng-Jun, TV Chinh, et al. Nearly constant approximation for data aggregation scheduling in wireless sensor networks. In *Proceedings of INFOCOM*, pages 366–372, 2007.

[29] Stanislava Soro and Wendi B Heinzelman. Prolonging the lifetime of wireless sensor networks via unequal clustering. In *Parallel and Distributed Processing Symposium, 2005. Proceedings. 19th IEEE International*, pages 8–pp. IEEE, 2005.

[30] Huseyin Ozgur Tan, Ibrahim Korpeoglu, and Ivan Stojmenovic. Computing localized power-efficient data aggregation trees for sensor networks. *Parallel and Distributed Systems, IEEE Transactions on*, 22(3):489–500, 2011.

[31] Chris Townsend and Steven Arms. Wireless sensor networks. *Principles and Applications, Sensor Technology Handbook Newness Publication*, pages 575–589, 2005.

[32] H. W. Tseng, S. H. Yang, P. Y. Chuang, H. K Wu, and G. H. Chen. An energy consumption analytic model for a wireless sensor mac protocol. In *Proc. of IEEE Conference on Vehicular Technology*, volume 6, pages 4533–4537, 2004.

[33] J. Wan, Y. Yu, Y. Wu, R. Feng, and N. Yu. Hierarchical leak detection and localization method in natural gas pipeline monitoring sensor networks. *Sensors*, 12(1):189214, 2011.

[34] Zhengjie Wang, Xiaoguang Zhao, and Xu Qian. The application and issuse of linear wireless sensor networks. In *System Science, Engineering Design and Manufacturing Informatization (ICSEM), 2011 International Conference on*, volume 2, pages 9–12. IEEE, 2011.

[35] Yan Wu, Zhoujia Mao, Sonia Fahmy, and Ness B Shroff. Constructing maximum-lifetime data gathering forests in sensor networks. *IEEE/ACM Transactions on Networking (TON)*, 18(5):1571–1584, 2010.

[36] G. S. Yang and M. B. Xiao. Node placement strategies to prolong lifetime in one-dimensional linear wireless sensor network. 55:17051710, 2011.

[37] Fan Ye, Gary Zhong, Jesse Cheng, Songwu Lu, and Lixia Zhang. Peas: A robust energy conserving protocol for long-lived sensor networks. In *Distributed Computing Systems, 2003. Proceedings. 23rd International Conference on*, pages 28–37. IEEE, 2003.

[38] Haibo Zhang and Hong Shen. Balancing energy consumption to maximize network lifetime in data-gathering sensor networks. *Parallel and Distributed Systems, IEEE Transactions on*, 20(10):1526–1539, 2009.

[39] Haibo Zhang, Hong Shen, and Yasuo Tan. Optimal energy balanced data gathering in wireless sensor networks. In *Parallel and Distributed Processing Symposium, 2007. IPDPS 2007. IEEE International*, pages 1–10. IEEE, 2007.

[40] X. Zhang and K. Shin. E-mili: energy-minimizing idle listening in wireless networks. *IEEE Transactions on Mobile Computing*, 11(9):1441–1454, 2012.

[41] Marco Zimmerling, Waltenegus Dargie, and Johnathan M Reason. Localized power-aware routing in linear wireless sensor networks. In *Proceedings of the 2nd ACM international conference on Context-awareness for self-managing systems*, pages 24–33. ACM, 2008.