DUTY CYCLE OPTIMIZATION IN ENERGY HARVESTING SENSOR NETWORKS WITH APPLICATION TO BLUETOOTH LOW ENERGY

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

BERK AKGÜN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

JUNE 2014

Approval of the thesis:

DUTY CYCLE OPTIMIZATION IN ENERGY HARVESTING SENSOR NETWORKS WITH APPLICATION TO BLUETOOTH LOW ENERGY

submitted by **BERK AKGÜN** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen
Dean, Graduate School of **Natural and Applied Sciences** ______________

Prof. Dr. Gönül Turhan Sayan
Head of Department, **Electrical and Electronics Engineering** ______________

Prof. Dr. Elif Uysal Bıyıkoğlu
Supervisor, **Electrical and Electronics Engineering Dept., METU** ______________

Examining Committee Members:

Prof. Dr. Semih Bilgen
Electrical and Electronics Engineering Dept., METU ______________

Prof. Dr. Elif Uysal Bıyıkoğlu
Electrical and Electronics Engineering Dept., METU ______________

Prof. Dr. Gözde Akar
Electrical and Electronics Engineering Dept., METU ______________

Assoc. Prof. Dr. Ece Güran Schmidt
Electrical and Electronics Engineering Dept., METU ______________

Dr. Ahmet Utku Yargıçoğlu
HBT TD TYM, ASELSAN ______________

Date: **11.6.2014**

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: BERK AKGÜN

Signature :

ABSTRACT

DUTY CYCLE OPTIMIZATION IN ENERGY HARVESTING SENSOR NETWORKS WITH APPLICATION TO BLUETOOTH LOW ENERGY

Akgün, Berk
M.S., Department of Electrical and Electronics Engineering
Supervisor : Prof. Dr. Elif Uysal Bıyıkoğlu

June 2014, 72 pages

The intracluster communication protocols of wireless sensor networks (WSNs) in the literature are designed according to time division multiple access (TDMA) with random slot allocations. This thesis proposes a novel scheduling algorithm for WSNs, in which cluster members (CMs) request time slots according to their energy predictions, and cluster head (CH) assigns these slots to members. The aim of this work is to increase the network lifetime by arranging the duty cycling of wireless sensor nodes. The simulations show that the proposed algorithm outperforms the intracluster communication protocol designed in LEACH. In addition, this thesis includes the implementation of a new communication technology, Bluetooth Low Energy (BLE), in a way to ensure energy neutral operation. BLE113 Development Kit transmits the advertisement packets, whose advertising intervals are arranged according to the energy harvesting pattern, to the BLE112 USB Dongle by Bluegiga. These advertisement packets also allow us to observe the conditions of the energy sources that power up BLE113 Development Kit. The tests conducted using this application in indoor and outdoor environments show that the throughput is maximized.

Keywords: Wireless sensor networks, clustering, TDMA, duty cycle optimization, Bluetooth Low Energy, energy harvesting, energy efficient scheduling, throughput maximization

ÖZ

ENERJİ HASATI YAPAN SENSÖR AĞLARINDA GÖREV DÖNGÜSÜ ENİYİLEMESİ VE BLUETOOTH LOW ENERGY AĞLARINA UYGULANMASI

Akgün, Berk
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi : Prof. Dr. Elif Uysal Bıyıkoğlu

Haziran 2014 , 72 sayfa

Literatürde kablosuz sensör ağları için varolan küme içi haberleşme protokolleri zaman bölmeli çoklu erişime (TDMA) göre, gelişigüzel zaman aralıkları atayarak yapılmıştır. Bu tez çalışması, küme elemanlarının enerji tahminlerine göre bir zaman aralığı isteğinde bulunduğu ve küme liderinin bu zaman aralıklarını küme elemanlarına atadığı yeni bir zaman çizelgelemesi methodu önerir. Bu yöntemle kablosuz sensörlerin görev döngüsü ayarlanarak yaşam sürelerinin uzatılması hedeflenir. Yapılan simülasyonlar neticesinde algoritmamızın LEACH'de kullanılan küme içi haberleşme protokolüne kıyasla daha iyi sonuçlar verdiği gösterilmiştir. Ayrıca, bu tez, yeni bir haberleşme teknolojisi olan Bluetooth Low Energy (BLE) protokolü için geliştirilen sadece hasat edilen enerjinin kullanıldığı yeni bir duyuru çizelgelemesi tekniğinin uygulamasını yapar. BLE113 Development Kit, duyuru aralıkları enerji hasatına göre uyarlanmış duyuru paketlerini Bluegiga firmasının bir başka ürünü olan BLE112 USB Dongle cihazına gönderir. Aynı zamanda, bu duyuru paketleri, BLE113 Development Kit ürününü besleyen enerji kaynaklarının durumunu gözlememizi sağlar. Bu uygulama kullanılarak iç ve dış mekanlarda alınan ölçümlerde, duyuru yayılmasının enyükseltildiği görülmüştür.

Anahtar Kelimeler: Kablosuz sensör ağları, kümelenme, TDMA, görev döngüsü en iyileme, Bluetooth Low Energy, enerji harmanlama, enerji-verimli çizelgeleme, çıktı enyükseltme

To my family

ACKNOWLEDGMENTS

I would like to express my special thanks to my supervisor Prof. Elif Uysal Bıyıkoğlu for her valuable guidance and sharing her knowledge through every step of this thesis.

I also thank to Prof. Semih Bilgen, Prof. Gözde Akar, Assoc. Prof. Ece Güran Schmidt and Dr. Ahmet Utku Yargıçoğlu for their valuable contributions to my thesis defense committee.

In addition, I thank to Baran Tan Bacınoğlu and Fatih Özçelik for their precious help to my thesis research.

I want to thank to TUBITAK as well, for supporting me financially through my MS degree program.

I want to thank to ASELSAN Inc. and my employers for letting me involve in this academic study.

I want to thank to my love Irmak Aykın for all her support. Without her, this thesis could not be completed.

I also thank to Erman Zaim, Ulaş Turan and Hakan Akış for their motivational support through all my stressful moments. In addition, I thank to my cat, Böcek, for keeping me awake at nights and pushing me to study, especially in spring time☺ I also thank to Fenerbahçe FC, for finishing the Turkish Football League as the leader and not adding any extra stress to my already stressful term.

Finally, for their valuable support in every step of my life, I am grateful to my mother, my father, my dear brother Ediz Akgün and the rest of my family. Without them, I would not be who I am today.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

BLE	Bluetooth Low Energy
CH	Cluster Head
CM	Cluster Member
CSMA	Carrier Sense Multiple Access
EHPBS	Energy Harvesting Prediction Based Scheduling
HEED	Hybrid Energy Efficient Distributed Clustering
ICT	Information and Communication Technologies
LEACH	Low Energy Adaptive Clustering Hierarchy
MAC	Multiple Access Control
SIG	Special Interest Group
TDMA	Time Division Multiple Access
WSN	Wireless Sensor Network

CHAPTER 1

INTRODUCTION

In order to compensate for the increasing demand of today's ICT infrastructure on energy, renewable energy sources have become a focal point of interest. Renewable energy technologies serve a large variety of needs. Particularly, in the communication area, as devices become smaller with advanced wireless technologies, they need to adapt themselves to renewable energy technologies more than ever. Wireless sensor networks which have attracted great interest for the last two decades, are a good example to this. It is often not practically possible to collect and replace the sensor nodes' batteries when they are depleted, as the nodes are usually dispersed to a large area. Due to this, many energy-efficient operation mechanisms have been developed for wireless sensor networks. Up to recent years, most of these algorithms focused on minimizing battery usage and maximizing the lifetime of nodes. However, recently, the focus of the work has changed, due to the emerging of the *energy harvesting* capability of sensor nodes.

Energy harvesting is the process of deriving energy for low-power electronics from external sources such as solar, wind and kinetic power. Recently, one of the research areas on wireless sensor networks has become obtaining maximum task performance using the harvested energy. To achieve this, one of the most interesting problems is duty cycling. Duty cycle can be defined as the ratio of the active time of an electronic device to the sleep time. Different duty cycling arrangements have been made, considering the energy inside the batteries and energy harvesting capabilities of the devices. In addition, various energy prediction algorithms have been designed, in order to calculate the energy that can be harvested from the environment. Especially in solar energy applications, the predictions that are based on historical data have made a significant contribution to the development of online algorithms.

Previous work in these areas can be briefly summarized as follows: [28] and [31] predict future energy profiles and arrange duty cycles of wireless sensor nodes in a way to ensure energy neutral operation. [29] and [32] are clustering algorithms that aim to maximize energy efficiency is WSNs, namely, LEACH and HEED. Moreover, [33] and [30] includes the effect of energy harvesting opportunities to the algorithms in LEACH and HEED. However, they do not consider the effect of energy harvesting opportunities while specifying the intra-cluster communication protocol. In addition, [34], [35] and [36] are TDMA based MAC protocols; whereas, [37], [38] and [39] are CSMA based MAC protocols, where all of them aim to improve energy efficiency for WSNs by arranging duty cycling of wireless sensor nodes. However, none of these papers consider the effect of energy harvesting capabilities of devices.

Our contribution is to design an energy prediction based scheduling algorithm for wireless sensor networks that use clustering. In this way, we aim to increase network lifetime and maximize task performance by deciding which sensors to communicate in which time slot. However, since we were not able to implement our algorithm, we moved on with a similar problem in Bluetooth Low Energy.

BLE is a short range communication technology that provides low cost and low power communication and aims to improve the functionality and area of application of classic Bluetooth. It is an ideal choice for mobile and sensor applications that use point-to-point communication. Advertising intervals of the BLE modules that are capable of harvesting energy should be chosen properly, in order to consume the harvested energy efficiently. That way, an efficient scheduling algorithm can be designed for many mobile applications that use advertisement packets.

In the literature, BLE devices are inspected in terms of energy efficiency. BLE and its alternative ZigBee/802.15.4 are compared in [40],in terms of energy consumptions. BLE is proved to be more efficient than ZigBee/802.15.4 in terms of bits/J and it is emphasized that it can become even more efficient if some constraints in the protocol are eliminated. In [41], the energy analysis of BLE devices is made. In the paper, the energy consumed by an advertising device and corresponding delays of the packets captured by the scanning device are defined and investigated by experiments. The scheduling algorithm designed in [24] arranges advertising intervals according to energy harvest rate, and in return, maximizes transmitted advertisement packets and minimizes the probability of wasting packets by not capturing them in scan mode.

Our contribution is to inspect the offline scheduling algorithm designed in [24] and [25], and test its efficiency in a real test environment with BLE113 Bluetooth Smart Modules. With the data that we acquired using a real energy harvesting system, we have made valuable contributions to the literature. In addition, by using this simple hardware and the data acquired, many energy efficient mobile applications can be developed.

The rest of this thesis is organized as follows. In Chapter 2, the related background information is provided. Wireless sensor networks and clustering algorithms, with their benefits, are described briefly. The emerging of energy harvesting networks,and their design issues are explained. In addition, their energy harvesting architectures are described. Then, Bluetooth and BLE technologies and their evolutions are described. Moreover, their protocol structures are briefly outlined.

In Chapter 3, an energy harvesting prediction based scheduling algorithm that exploits battery round-trip efficiency is introduced. Battery types and energy prediction models are explained in detail and the proposed algorithm is given. Then, the simulation environment is laid out and simulation results are provided with detailed explanations. After that, implementation problems regarding this algorithm are explained and some information on how to overcome these problems with BLE technology is given.

Then, in Chapter 4, a scheduling problem with the goal of maximizing the advertisement packet throughput is introduced. The theoretical definition of the problem is given and optimum solution to this problem is described, similar to [24] and [26].

In Chapter 5, the hardware and software equipments that are used in simulations are introduced and their related properties are explained. Also, optimum solution to data rate maximization problem is verified using simulations, and the comparisons between random and optimum scheduling are made.

Finally, in Chapter 6, the problems and proposed solutions are summarized and the deductions of the results are made. Then, the thesis is concluded.

CHAPTER 2

BACKGROUND

2.1 Wireless Sensor Networks (WSNs)

WSNs are dense wireless networks of small, low-cost sensor nodes, which collect and distribute environmental information [1]. These cheap and smart sensors that provide novel opportunities for documenting and controlling homes, cities, and the environment, are connected through wireless links and deployed in large numbers [2]. They are commonly used in both military and civilian applications such as weather and environmental monitoring, target detection, and natural disaster prevention.

WSN nodes are typically small and their physical qualifications are limited. These units typically consist of 3 components, namely, processing unit, sensing unit and transmission unit. Compared to others, transmission unit has the major impact on battery lifetime [3]. Each sensor node is capable of only a limited amount of processing. However, in coordination with the data from a large number of other nodes, they are able to measure a given physical environment in great detail. Thus, a sensor network can be described as a collection of sensor nodes which work together to perform some specific action. Unlike traditional networks, sensor networks depend on dense deployment and coordination to perform their tasks [1].

Previously, sensor networks consisted of small number of sensor nodes that communicated with a central processing station. However, nowadays, distributed sensing is more common. This is necessary when communication is the major consumer of energy. A centralized system means that some of the sensors may need to communicate over long distances that lead to more energy consumption. Therefore, a distributed system architecture which can process as much data as possible locally so as to minimize the total number of bits transmitted is desirable [1].

2.1.1 Clustering

Sensor nodes are often grouped into clusters in order to achieve the network scalability objective. Every cluster has a leader, which is called the cluster-head (CH). The aim is to generate stable clusters in environments with mobile nodes; even though, various clustering algorithms with different objectives have been proposed in the literature. Many of those algorithms consider node reachability and route stability most, without much concern about critical design goals such as network longevity and coverage. [43] implemented the energy efficient and competitive clustering algorithm ,HEED, using MICAz motes. With the testbed they set, the authors worked on power aware routing, topology control, time synchronization, sleep scheduling and transmission power adaptation problems.

Figures 2.1 and 2.2 illustrate an application where sensors periodically transmit data to a remote base station. The figures show that clustering can reduce the communication overhead for both single-hop and multihop networks [5].

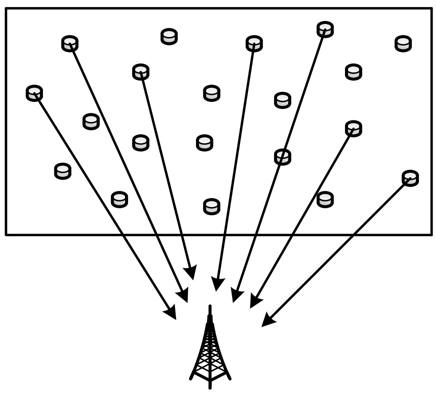

Figure 2.1: Sensor information forwarding without clustering and aggregation.

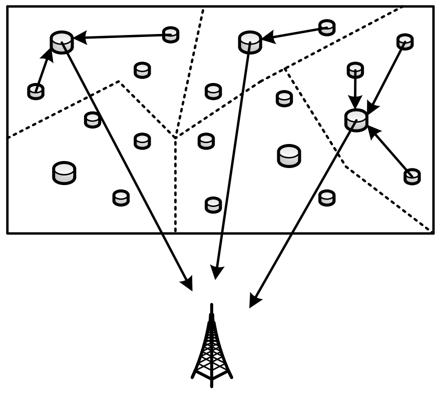

Figure 2.2: Sensor information forwarding with clustering and aggregation.

In addition to supporting network scalability, clustering has many other advantages. It can reduce the size of the routing table stored at the individual node by localizing the route set up within the cluster [6]. Clustering can also save bandwidth since it limits the inter-cluster interactions to CHs and avoids unnecessary exchange of messages among sensor nodes [7]. Moreover, clustering can stabilize the network topology at the level of sensors and thus they are freed from topology maintenance overhead. Sensors would only be concerned about connecting with their CHs and would not be affected by changes at the level of inter-CH tier [8]. A CH can schedule the activities of the cluster so that nodes can switch to the low-power sleep mode most of the time and reduce the total energy consumption. Sensors can be engaged in a round-robin order and the time for their transmission and reception can be determined so that the sensors reties are avoided, redundancy in coverage can be limited and medium access collision is prevented [9], [10], [11] and [12]. Furthermore, a CH can aggregate the information gathered together by the sensors in its cluster and decrease the number of relayed packets [13].

2.2 Energy Harvesting Networks

Wireless and embedded systems are commonly powered using batteries. For applications where the system is expected to function for long durations, energy usage becomes a severe constraint and researchers spend much effort on the efficient use of battery power. Lately, another alternative has begun to be used to supplement or even replace batteries: harvesting energy from the environment. For example, [44] is a recent work which demonstrates the energy-neutral operation of a wireless sensor network of MicaZ motes through electromagnetic vibration energy harvesting. Here, the authors appropriately adjust the duty cycles of the nodes employing the standard ZigBee protocol; thus, the energy harvesting fully compensates for the energy used for the operation of the nodes. On the other hand, [46] investigates the resource allocation problem of the dynamic access points in energy harvesting networks and offers online heuristic solutions. Furthermore, [48] seeks an offline solution for scheduling in energy harvesting systems with fading channels, and do the implementation of three scheduling algorithms for energy efficiency on Universal Software Radio Peripheral (USRP). Also, [49] suggests a low-complexity scheduling policy for solving a restless multi-armed bandit problem occurring in a single hop wireless network. In addition, [47] works on the optimization of feedback in MISO downlink multi-user system with energy harvesting users. This is important for WSNs with users that harvest energy at a low rate, such as users that employ indoor solar cells. In most energy harvesting studies, it is assumed that the transmitter has full knowledge of channel conditions and do the power allocation according to it. However, in practical, this information is provided to the transmitter by feedback channels.

As seen here, some considerations of an energy harvesting sources are fundamentally different from that in using a battery, since, it has a limit on the maximum rate of energy usage, rather than a limit on the maximum energy. In energy harvesting networks, energy that can be harvested from the medium is usually more critical than the energy level of the battery. Moreover, the available harvested energy typically varies with time in a nondeterministic manner. For instance, in solar energy applications, direct sunlight can be blocked by a cloud or the node may be located in a shadow. In that case, according to the needs of our task, we may decide to increase the activity level of our system when we can benefit the sunlight and switch the system to sleep mode when the nodes cannot harvest much energy. That's why, while a deterministic metric, such as residual battery, is enough to characterize the energy availability in the case of batteries, a more complicated characterization may be required for a harvesting source [15].

An effective approach is to adapt duty cycling to energy availability in the environment. There are models that enable energy harvesting nodes to predict future energy opportunities based on historical data. [28] focuses on periodic energy sources such as solar energy. Since the energy source is periodic, the authors can make energy predictions according to Exponentially Weighted Moving-Average (EWMA) model, without having any information on future energy profiles. With this model, the energy predictions can adapt seasonal changes in addition to daily variations. Under the light of these predictions, [28] tries to find the optimal duty cycle that allows energy neutral operation. In short, energy profiles can be predicted and used for better performances of the network.

According to [4] energy harvesting systems employ two different architectures to use the harvested energy. The first one is harvest-use model, that exploits using the harvested energy instantly; whereas, in the second one, harvest-store-use model, energy is harvested whenever possible and stored for future use. Figure 2.3 shows harvest-use and harvest-store-use architectures.

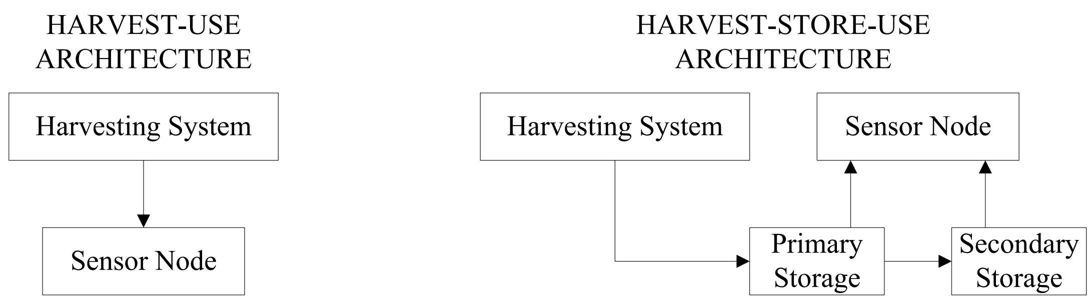

Figure 2.3: Harvest-Use and Harvest-Store-Use Architectures [4]

On the other hand, [45] introduces another architecture for the system with a non-ideal buffer to use the harvested energy immediately, instead of storing in the buffer. Then, the remaining energy after the transmission is stored into the buffer. Thus, the wasted power due to storage inefficiencies can be decreased. This is called harvest-use-store architecture.

2.3 Bluetooth

Bluetooth is a wireless technology standard for short range communications (using short-wavelength UHF radio waves in the ISM band from 2.4 to 2.485 GHz [17]) between fixed and mobile devices. Developed by telecom vendor Ericsson in 1994,[18] it was originally designed as a wireless alternative to RS-232 data cables. It can connect several devices, and overcome synchronization problems.

Bluetooth is a technology that enables exchanging data over short distances. Its robustness, low power, and low cost make Bluetooth ideal for a wide range of devices. The standards of Bluetooth technology is defined and maintained by the Bluetooth Special Interest Group (SIG) in the "Core Specification" [19].

2.3.1 Bluetooth Smart

The Bluetooth SIG completed the Bluetooth Core Specification version 4.0 (called Bluetooth Smart) and it has been adopted as of 30 June 2010. Bluetooth Smart includes Classic Bluetooth, Bluetooth High Speed and Bluetooth Low Energy protocols. Bluetooth High Speed is based on Wi-Fi, and Classic Bluetooth consists of conventional Bluetooth protocols. The changes necessary to facilitate BLE modes, in addition to Generic Attribute Profile (GATT) and Security Manager (SM) services constitute general improvements of version 4.0.

2.4 Bluetooth Low Energy

Bluetooth Low Energy (BLE) is an emerging wireless technology developed by the Bluetooth SIG for short-range communication. Different from previous Bluetooth technologies, BLE has been designed as a low-power solution for control and monitoring applications. BLE is the unique feature of the Bluetooth 4.0 specification [21].

BLE defines two roles for devices at the Link Layer for an established connection: the master and the slave. These devices act as initiator and advertiser during a connection, respectively. A master can establish simultaneous connections with several slaves, whereas each slave can only belong to one master. Thus, the network consists of a master and its slaves, which is called a piconet, and it creates a star topology, as shown in Figure 2.4. Currently, a BLE device can belong to only one piconet [22].

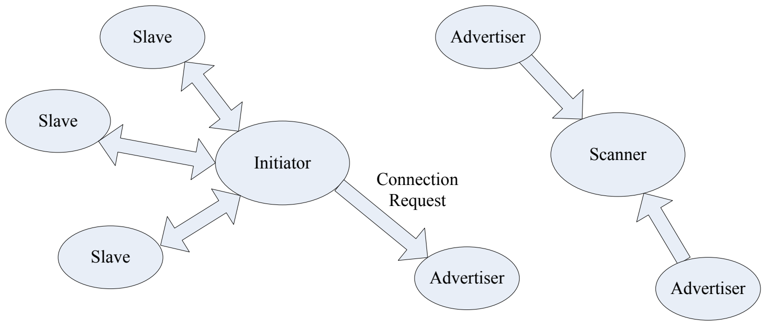

Figure 2.4: Bluetooth Low Energy Topology [42]

One major difference between BLE and Classic Bluetooth is that BLE uses only 40 channels, 2 MHz wide, while Classic Bluetooth uses 79 channels, 1 MHz wide. Three channels (advertising channels) are used for discovering devices and establishing connections. These channels are used to search for other devices or promote its own presence to devices that might be looking to make a connection. On the other hand, Classic Bluetooth technology uses 32 channels for the same task [19].

This drastic reduction is another trick that BLE uses to minimize time on air, so as to reduce power consumption. BLE has to switch to "ON" mode for just 0.6 to 1.2 ms to scan for other devices using its three advertising channels. Classic Bluetooth, instead, requires 22.5 ms to scan its 32 channels. As a result, power savings are significant: BLE consumes 10 to 20 times less power than Classic Bluetooth for device discovery [19].

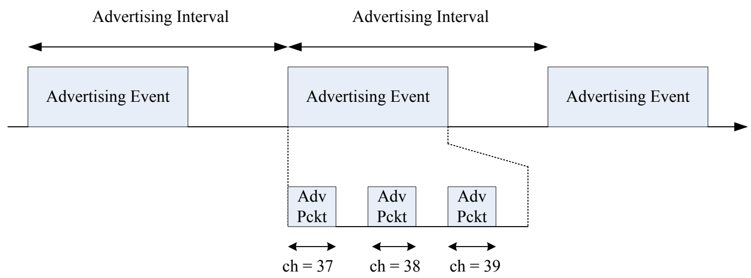

Figure 2.5: Bluetooth Low Energy advertisement mode [23]

As shown in Figure 2.5, device in advertising mode is called advertiser and it periodically sends advertisement packets in channels 37, 38 and 39. Then, it listens the channels for responses from other devices. In that respect, the operation of sending advertisement packets over each of the advertising channels is defined as *advertising event*. Also, time spent between two successive advertisement packets in the same channel is called *advertising interval* [23].

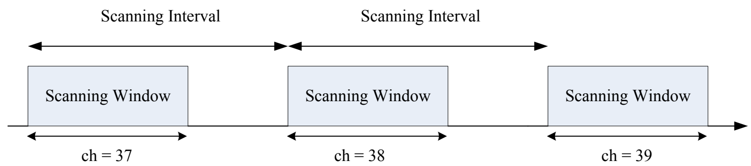

Figure 2.6: Bluetooth Low Energy scan mode [23]

On the other hand, device in scanning is called scanner and it periodically scans the advertising channels, as shown in Figure 2.6. The duration of listening to a channel is called *scanning window*; whereas, time spent between the events of listening to different advertising channels is called *scanning interval* [23].

CHAPTER 3

EHPBS: ENERGY HARVESTING PREDICTION BASED SCHEDULING IN WIRELESS SENSOR NETWORKS

3.1 Problem Definition

Clustering algorithms are used for load balancing and scalability in distributed networks. [29] developed LEACH operation for WSNs, in which intra-cluster and inter-cluster communication schemes are defined, together with the cluster formation algorithm.

LEACH operation starts with a set-up phase, where cluster heads (CHs) are selected and members are assigned to clusters. Within that phase, CH learns about its member nodes and sets up a TDMA schedule among them. Then, it broadcasts this schedule to all its members. This way, collusions within a cluster are prevented and energy efficiency is provided by allowing non-transmitting nodes to go sleep mode.

In the next step, steady-state phase, the operation consists of frames, where one transmission slot per frame is allocated to member nodes. The time line of LEACH operation is shown in the Figure 3.1.

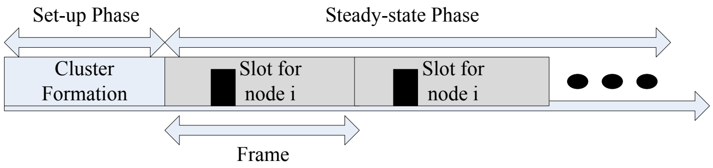

Figure 3.1: Time line showing LEACH operation and transmission slots of a random node i

The intra-cluster communication protocol of LEACH will be called the regular scheme hereafter. In this system, CH assigns slots to members randomly, in a TDMA based manner. That is, CH doesn't consider any efficiency parameter when allocating the slots. However, these time slots should be assigned according to a pattern that considers available energy.

If our system has the ability to harvest energy, we can write the following approximate equation using the rule of energy conservation [28]

$$B_0 + \eta \int_0^T [P_S(t) - P_C(t)]^+ \, dt \; - \int_0^T [P_C(t) - P_S(t)]^+ \, dt \; - \int_0^T P_{leak} \, dt \; \geq 0 \quad (3.1)$$

Here, $P_S(t)$ is defined as the energy harvested from the medium at time t. $P_C(t)$ is, on the other hand, the energy being consumed by the load at that time. Round-trip efficiency, the ratio of the energy stored in the battery to the energy brought to the battery, is denoted as η and constant leakage power is denoted as P_{leak}. B_0 is the current power level of the battery.

What we understand from here is that while charging the battery, we waste some harvested energy, as η is strictly less than 1. In short, it means that our buffer is inefficient, and if we can spend the harvested energy without storing it into the battery, we can decrease the wasted power considerably.

Using this information, we are trying to set up a TDMA schedule for clusters according to the energy predictions. In other words, CHs will assign time slots according to the requests from cluster members, and the requests will be made according to the energy predictions. This process will take place in the setup phase and it will not add any overhead to the system, since in LEACH, CMs already send a join-request message to the CH in the setup phase. Thus, when a cluster member wakes up and starts harvesting energy, it will also be able to transmit, since its time slot will be assigned accordingly. This way, nodes will not waste their energy trying to recharge their battery. Instead, they will use the harvested energy instantly, as much as possible. Consuming the harvested energy instantly is beneficial, since some of the harvested energy is lost due to storage inefficiencies. Thus, not only longer lifetime duration of wireless sensor networks but also maximum task performance will be achieved.

In the next section, we will inspect the battery types, develop an energy prediction model and propose a novel TDMA scheduling algorithm for intra-cluster communication of WSNs.

3.2 Proposed Scheme

3.2.1 Battery Selection

Since the parameter that we deal with most in our algorithm is η, we should investigate it in detail for several different types of batteries. Li-ion and Li-polymer batteries have many advantages such as their high output voltages, energy densities and round trip efficiencies. On the other hand, NiMH, NiCd and SLA batteries have lower values of these parameters, compared to Li-based batteries. However, they can be trickle charged and thus, can be directly recharged by a power source, where Li-based batteries require pulse charging with an extra battery or a charging circuit for that purpose. Since wireless sensor nodes are very small, this pulse charging circuit would add an overhead to the system. Therefore, Li-based batteries are not very suitable to be used in wireless sensor nodes. Super capacitors are also a good alternative for their high η values, but they have high leakage. For that reason, they are not used alone but they can be combined with other batteries in double-storage energy harvesting systems [4]. As a result, since NiMH batteries are used most widely in WSNs, we took 0.66 for η value, using Table 3.1.

Table3.1: Battery Comparison [4]

Battery Type	Nominal Voltage (V)	Weight Energy Density (Wh/kg)	Power Density (W/kg)	Efficiency	Self Discharge (%/months)	Charging Method
SLA	6	26	180	0.7 - 0.92	20	Trickle
NiCd	1.2	42	150	0.7 - 0.9	10	Trickle
NiMH	1.2	100	250 - 1000	0.66	20	Trickle
Li-ion	3.7	185	1800	0.99	< 10	Pulse
Li-polymer	3.7	156	3000	0.99	< 10	Pulse

3.2.2 Energy Harvesting Architecture

As expressed before, the main point of our proposed algorithm is to consume the harvested energy instantly, without charging the battery. That way, we aim to prevent wasted power resulting from storage inefficiencies. However, in order to design a realistic system, we should first investigate the energy harvesting architectures.

[14] develops a double-storage energy harvesting system that works with solar energy. Using this system, we can design an energy harvesting architecture that meets the needs of our algorithm. That way, it would be possible to implement the algorithm proposed in this thesis.

Figure 3.2 shows the energy harvesting architecture that is designed to meet the needs of our system. Here, the energy harvested by the solar panel is used to charge the super capacitor. Since the storage efficiencies of super capacitors are quite large (97 - 98 %), only a small amount of the energy is wasted during this process [4]. However, due to their high self-discharge values, super capacitors need a secondary storage. As the secondary storage, we decided to use NiMH batteries, for the reasons explained before.

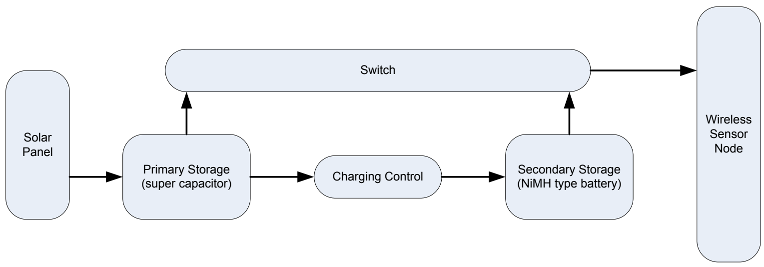

Figure 3.2: Energy harvesting architecture with two storage units

The architecture designed in [14] can choose the power source for the node by using a software driver. Similarly, the switch in Figure 3.2 decides whether the primary storage or the secondary storage will be used to power up the system. If the sensor node is active, switch chooses primary storage which prevents wasted power. On the other hand, if the sensor node is in sleep mode, the harvested energy passes through the primary storage and is used to recharge the NiMH battery. That way, the energy within super capacitor can be transferred to the secondary storage with minimum leakage. When the sensor node becomes active again but cannot harvest energy, the switch powers it up using NiMH battery. In short, this architecture meets all needs of our algorithm.

3.2.3 Energy Prediction

In our energy prediction model, we divide each day to slots and at the end of each slot, we try to calculate the energy that can be harvested in the next slot. We use Exponentially Weighted Moving-Average (EWMA) [28]. In this method, historical data is stored and used for future energy predictions, since the energy generation of a given slot is expected to be similar to the energy generations at the same slots on the previous days. Due to exponential weights, ss the observed day gets older, its effect on the energy prediction diminishes.

As a result ,when estimating the energy in day n slot i, we use the following equation

$$E_{est1}(n,i) = \alpha E_{gen}(n-1),i) + (1-\alpha)E_{est1}(n-1,i) \tag{3.2}$$

Here, E_{est1} is the energy forecasted in a succeeding slot; E_{gen} is the value of the generated energy in that interval and α is the filter coefficient.

Then, when estimating the energy in day n, we include the generated energy value of the previous day with weight α and the term $E_{est1}(n-1,i)$ involves the energy generations of older days. If we replace the term $E_{est1}(n-1,i)$ with its corresponding function as in Equation 3.2 iteratively, until the beginning of historical data, we reach

$$E_{est1}(n,i) = \sum_{k=0}^{n-1} \alpha(1-\alpha)^k E_{gen}(n-(k+1),i) \tag{3.3}$$

Here, we have assumed that consecutive days have similar weather conditions. However, this may not be a true assumption and a day can have different weather. To find the error percentage in estimated energies in that day up to slot i, we can use the generated energy values. Then, using the error percentage we calculated, we can improve our future estimations. Then, we achieve the next equation.

$$E_{est}(n,i) = E_{est1}(n,i)\left(\frac{\sum_{k=0}^{i-1} E_{gen}(n,k)}{\sum_{k=0}^{i-1} E_{est1}(n,k)}\right) \tag{3.4}$$

Here, since both of the terms in the denominator and numerator include α, they cancel out. However, the effect of α remains, since $(1-\alpha)$ remains in the equations.

Using $E_{est}(n,i)$ and E_{res}, residual energy, values we achieve the parameter $E(n,i)$ as

$$E(n,i) = w_1(E_{est}(n,i)) + w_2 E_{res} \tag{3.5}$$

where $w_1 = w_2 = 1$. That way, $E_{est}(n,i)$ and $E_{res}(n,i)$ will have equal weight on $E(n,i)$ calculation. However, if one parameter becomes more important, we can adjust the coefficients accordingly.

3.2.4 EHPBS

To further improve the existing intra-cluster communication protocol of LEACH, we developed Energy Harvesting Prediction Based Scheduling (EHPBS). The slot allocation algorithm is explained in Algorithm 1.

In EHPBS, all the members compute the $E_{est}(n,i)$ value for each slot and send this and E_{res} values to the CH during the set-up phase. CH collects this information from all its members and runs the algorithm. Among the nodes whose $E(n,i)$ values exceeds the battery capacity, the algorithm selects the node with the largest exceeding energy. If not, harvested energy can neither be used nor be stored; it would be wasted. If there aren't any such nodes, the algorithm will try to find the nodes such that their $E(n,i)$ values are above the power to be consumed per slot. Because if it cannot find such nodes, intracluster communication would stop.

If it can find such nodes that are capable of harvesting energy, CH assigns the slot to the node with largest $E_{est}(n,i)$. This way, we are planning to assign the channel

Algorithm 1 Assign a Node to the Next Slot

for *all members of the cluster* **do**
  if *Node.E(n,i) > BatteryCapacity* **then**
    selectedNode ← the Node having the highest $E(n,i)$ up to now
  end if
end for
return *selectedNode*
for *all members of the cluster* **do**
  if *Node.E(n,i) > RequiredEnergyPerSlot* **then**
    if *ThereIsAnyEnergyHarvestingOpportunity* **then**
      selectedNode ← the Node having the highest $E_{est}(n,i)$ up to now
    else
      selectedNode ← the Node having the highest E_{res} up to now
    end if
  end if
end for
return *selectedNode*

to the node that is harvesting most. Thus, the wasted power will be decreased, since harvested energy will be used to transmit without charging the battery. Because, as $E_{est}(n, i)$ increases, the wasted power due to round trip inefficiency also increases. If there are no nodes that are capable of harvesting energy, the slot is given to the node with largest E_{res}. This way, we try to avoid the batteries with low energy levels from being depleted. The algorithm is run for all the slots assigned for that cluster time.

Then, CH distributes the resulting TDMA schedule to its members and steady-state phase begins. The difference between EHPBS and the regular scheme in this phase is that EHPBS does not consist of frames. Figure 3.3 shows one round of the cluster.

Since no slot is skipped without transmission, the regular scheme and EHPBS will have the same throughput. In addition, similar to the regular scheme, non-transmitting nodes in EHPBS goes into sleep mode.

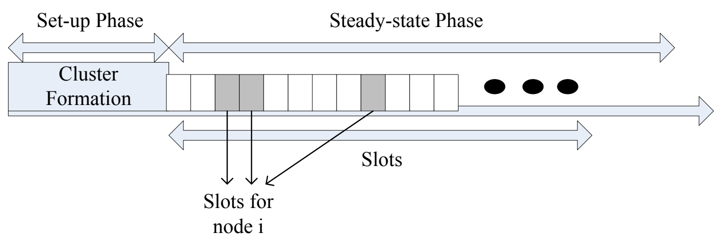

Figure 3.3: Time line showing EHPBS operation and transmission slots of a random node i

3.3 Simulation Results

3.3.1 Simulation Environment

We assume that the CH is selected according to LEACH. The performances of the regular algorithm and our scheme have been compared in terms of network lifetime. As the regular algorithm, we take the intra-cluster communication algorithm of LEACH since many other clustering algorithms also use that method [29]. The wasted power caused by the round-trip efficiency is expected to be decreased and thus the network lifetime will be increased with our proposed scheme. Synchronization problem has not been considered since it is outside the scope of this thesis. Nodes are assumed to have packets to send all the time and die when their unrechargeable battery becomes empty.

As the simulation parameters, we have decided to use the parameters in [30], since it also simulates clustering algorithms for energy harvesting WSNs. For that purpose, using these parameters would make our simulations more consistent with the ones in the literature. Then, we have set the parameters as follows; the time between two consecutive cluster setup phases=60 minutes, slot duration=30 seconds, initial energy of the rechargeable battery=50 J, initial energy of the unrechargeable battery=100 J, power consumption in active mode=38 mW and power consumption in sleep mode=30 μW. The effect of leakage is neglected since it is the same in both schemes.

When calculating the generating power of a solar cell, parameters that define weather conditions, location of the sensor node and the characteristics of solar cell should be known. Thus, to calculate the generating power, we use the following equation that involves all these parameters [30]

$$P_{charge}(t) = R_{weather} \times R_{position} \times f(t) \tag{3.6}$$

where $f(t)$ is the maximum power generation ability of the solar cell as a function of time. If $R_{weather}$, determined according to Table 3.2 and stays the same throughout the day, and $R_{position}$, depending on the position of the node, namely sun and shade, get their maximum values, the highest power generation is achieved. According to these values, a node can generate at most 30mW, as seen from Figure 3.4 [30].

From Equation 3.6, we can see that the parameters $R_{weather}$, $R_{position}$ and $f(t)$ determine the generating power of a solar cell. Since same type of solar cell is used in all nodes, $f(t)$ value is the same for all. In addition, all the nodes are subjected to same weather conditions. Thus, their $R_{weather}$ parameters are also same. As a result, the only parameter that varies between nodes is $R_{position}$, which takes the value 0.7 - 1.0 when the nodes are exposed to direct sunlight and 0.2 - 0.4 for the nodes in shade. However, since the nodes that form a cluster are close to each other, they are either exposed to sun or remain in shade all together.

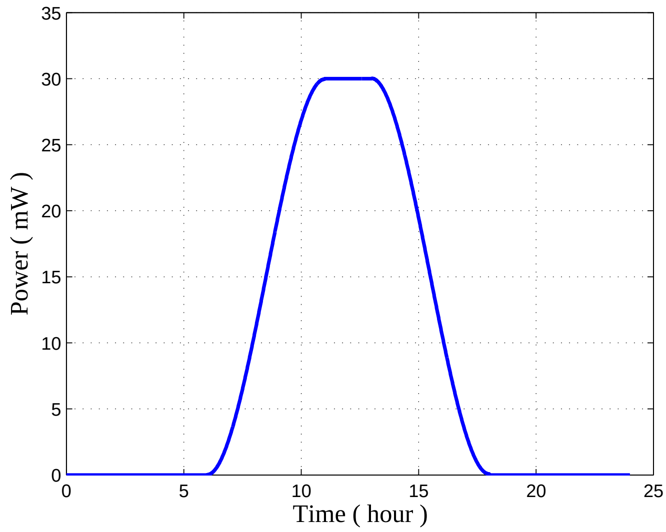

Figure 3.4: Energy Harvesting Capability vs. Time [30]

Table3.2: Weather Parameters [30]

type	prob	$R_{weather}$
fine	0.2	0.8 - 1.0
clear	0.3	0.6 - 0.8
cloudy	0.3	0.15 - 0.5
rainy	0.2	0.05 - 0.2

3.3.2 Simulation Results

Network lifetime is defined as the time spent until the first node dies. After the simulations we did to find an appropriate filter coefficient value α to be used in Equation 3.3, we achieved Figure 3.5. Then, α is taken as 0.2, which is found to be closest to the best value.

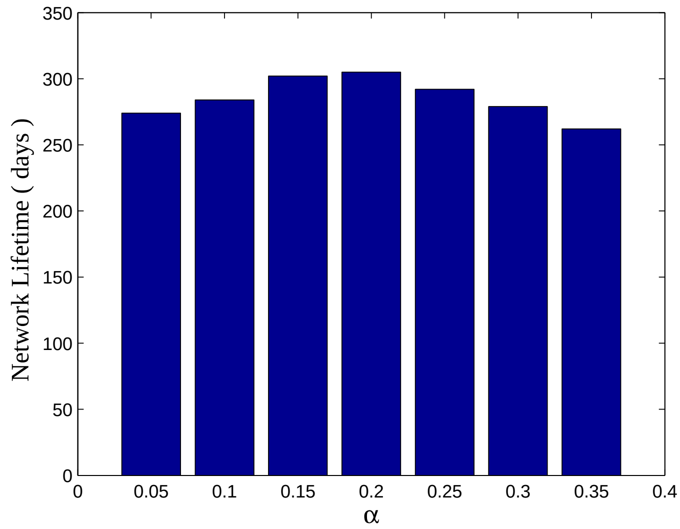

Figure 3.5: Network Lifetime versus Filter Coefficient α for randomly distributed 100 nodes in 200×200 m area by averaging 100 different realizations

As expressed before, the intracluster communication protocol of LEACH where the TDMA slots are assigned to CMs randomly by the CH, is called the regular scheme [29]. The regular scheme will be used in the following simulations to compare with EHPBS.

In Figure 3.6, we have considered wireless sensor nodes with different types of batteries. We have compared EHPBS and regular scheme in terms of network lifetime for various round trip efficiency values. Using the results acquired from the simulations, we have determined that the average increase in network lifetime while using EHPBS is around 3.76%.

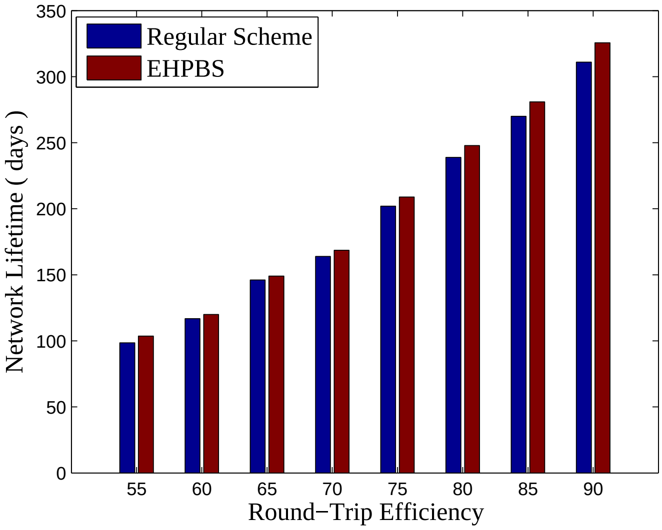

Figure 3.6: Network Lifetime versus Round-Trip Efficiency for randomly distributed 100 nodes in 200×200 m area by averaging 100 different realizations

In Figure 3.7, we have assumed that wireless sensor nodes use various types of solar panels. According to this figure, if we use solar panels with better harvesting capabilities, we can increase the network lifetime significantly. From Figure 3.7, we can see that the difference between EHPBS and the regular scheme increases with increasing energy harvest rate. Here, the average gain in network lifetime while using EHPBS is around 2.38%.

Finally, in Figure 3.8, we have inspected the effect of transmitter properties on network lifetime. Here, we see that when transmitters with higher dissipated powers are used, batteries are depleted faster. In this simulation, EHPBS provided 2.25% longer network lifetime.

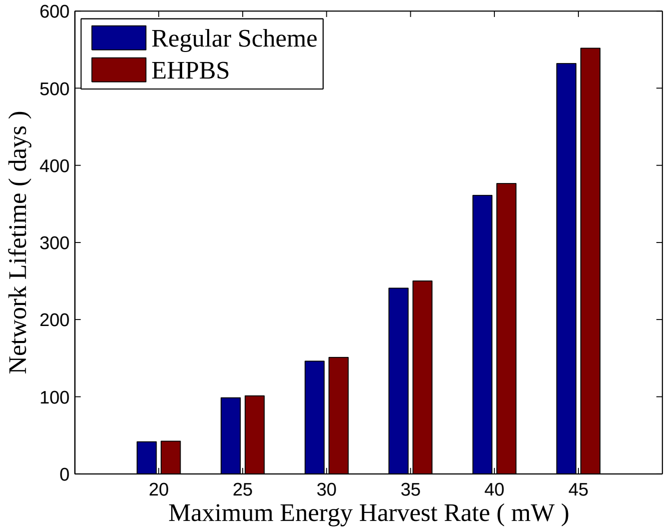

Figure 3.7: Network Lifetime versus Maximum Energy Harvest Rate for randomly distributed 100 nodes in 200 × 200 m area by averaging 100 different realizations

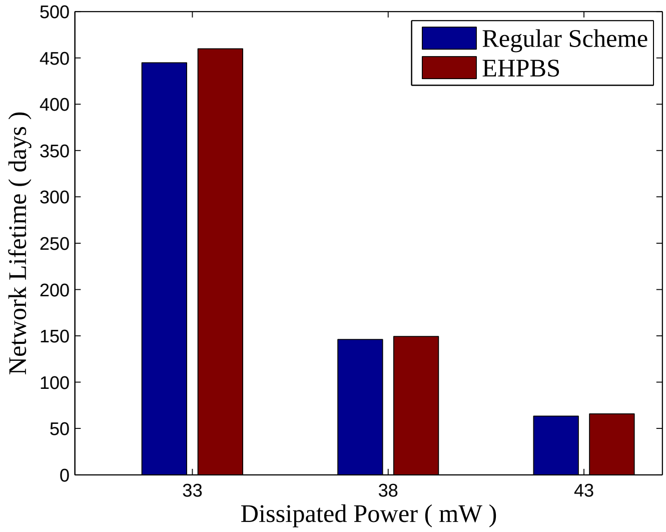

Figure 3.8: Network Lifetime versus Dissipated Power for randomly distributed 100 nodes in 200 × 200 m area by averaging 100 different realizations

In the simulations, the difference between regular scheme and EHPBS in terms of network lifetime is not very large. This is because, the only power generation parameter that varies between nodes is $R_{position}$, as expressed in Equation 3.6. Also, the nodes that form a cluster are close to each other, which causes their $R_{position}$ parameters to be close. Thus, the difference between the node that harvests most energy and the node that harvests least energy is quite small, within a cluster. For that reason, using that model, we were able to observe a small difference between the algorithms.

3.4 Conclusions and Further Issues

In this section, we developed an algorithm to increase the network lifetime of energy harvesting WSNs. We allowed nodes to transmit their data immediately after harvesting energy in order to decrease the wasted power due to storage inefficiencies. We used NiMH batteries and its round trip efficiency values in our calculations, since it is the most widely used battery in solar energy applications. In LEACH, CH does not consider any energy efficiency parameters when assigning a slot to a node and gives the channel to a node randomly; whereas, in EHPBS, the slot is given to the node that harvests most. That way, since the wasted power is decreased, network lifetime can be increased using EHPBS.

The next step of this work is to try the algorithm on the real test environment with actual nodes, in order to prove the benefits and results of EHPBS. However, due to several reasons, we cannot conduct the tests.

The first reason for that is, we cannot do the time synchronization of the micaZ motes. That is a critical step, since TDMA is used in our algorithm, instead of CSMA. micaZ motes do not allow us to construct time slots and make a coordination among the cluster. We could have tested a configuration consisting of one CH and one member node, but that would not make a realistic environment.

Second, in order to prove our results, our energy harvesting structure should have supported both harvest-use, harvest-store-use and harvest-use-store architectures. Because, the nodes with harvested energy less than consumed energy will use all the harvested energy instantly; whereas, nodes with harvested energy larger than consumed energy will use some of the harvested energy and store the rest inside their batteries. On the other hand, the nodes that are in sleep mode and do not consume any energy will store the all harvested energy instantly, inside their batteries. Since we do not have an energy harvesting structure that support both harvest-use, harvest-store-use and harvest-use-store architectures, we were not able to test our algorithm.

After that point, we decided to work on another duty cycling problem, using the energy harvesting opportunities in hand. We chose a recent and improvable area, Bluetooth Low Energy, and decided to use the information we gathered on this area.

As mentioned, BLE is used on applications where energy sources are limited. We specifically focus on mobile applications as they have unlimited potential. Indoor tracking is an example to mobile applications of our interest. It is possible to keep track of customers inside shopping malls, in terms visited stores, time spent in stores, searched products etc., using the Bluetooth modules. However, since these modules are tiny, their batteries should be used efficiently. Empty batteries can be located and replaced, but as the number of the modules increase, this would become harder. It would also cause the services to be disrupted for some time. Thus, the best service can be provided by effective algorithms designed for those modules that are capable of harvesting energy.

Similar to the problem in this chapter, we also deal with the scheduling of energy harvesting devices in BLE problem, using the available energy harvesting opportunities. In this chapter, our network consisted of a CH and its members; whereas, in BLE, the network consists of a server and a client. Thus, our previous problem was about selecting the most suitable CM and arranging the communication between CH and that CM, in order to maximize network lifetime. On the other hand, in BLE, we try to maximize the packet transmission of a point-to-point communication between server and client nodes, ensuring energy neutral operation. Also, in this chapter, when allocating a slot to a CM, the energy harvesting opportunities of all members are considered; whereas, in BLE, only the harvested energy of server is of concern. In short, two problems have some differences but they both try to optimize the communication between energy harvesting devices by arranging duty cycling.

With this motivation and broader development capabilities that Bluetooth modules ensured, we focused our research on BLE. Thanks to Bluetooth modules, we can easily establish a test setup using one master and one slave node and get rid of complex and out of scope time synchronization problem, as it is already provided by Bluetooth infrastructure. For all those reasons, we changed the direction of our research to BLE, using the information we gathered on EHWSNs.

CHAPTER 4

OPTIMIZATION OF ADVERTISING INTERVALS FOR ENERGY HARVESTING BLUETOOTH LOW ENERGY NETWORKS

4.1 Background

BLE is an important technology for energy harvesting networks that run applications which require low data rates. BLE's optimization is a concern of this work. In [24], the authors model the scheduling of advertisement packets of the BLE module that works on advertisement mode and harvests energy. The BLE module that works on advertisement mode is called node S. As expressed before, BLE modules use 1 Mbps data rate and advertise on channels 37,38 and 39 with constant average power. If we assume that channel conditions remain the same throughout transmission, we can also assume that the data throughput of S within a given time is directly proportional to the number of transmitted advertisement packets.

Another reason for this assumption is the recent information from the mobile device manufacturers which suggests that microlocation infrastructure works on advertisement packets. Here, even though S transmits separate packets through channels 37, 38 and 39, since these packets are consecutive, they are modelled as a single packet. The time spent between two successive packets in the same channel will be called advertising interval.

4.2 Problem Definition

The objective of the problem is to maximize an average data rate of S over a time frame of duration T, ensuring energy neutral operation. In order to do that, we should transmit as many advertisement packets as possible, with the given rate of harvested energy, within that interval T. However, advertising interval selection is also an important parameter to increase the packets' probability of being captured by the devices in scan mode. As we increase the frequency of our transmissions, we increase the packets' probability of being captured but we also decrease the remaining harvested energy and we may not be able to transmit at the remaining time. On the other hand, if we transmit less frequently, we may end up having extra energy by the time T. Then, we would have used the harvested energy inefficiently and we would also have decreased the packets' probability of being captured by the devices in scan mode.

The total number of transmitted advertisement packets up to time T is defined as $N_A(T)$ and the time spent for the transmission of an advertisement packet is defined as w. When not transmitting, the module is assumed to be in the sleep mode. Let us call the summation of energies spent in advertising and sleep modes up to time T as the consumed energy curve, $E_C(T)$. Then, if we write the consumed energy curve $E_C(T)$ up to that time T, in terms of $N_A(T)$, the equation becomes

$$E_C(T) = P_{adv} w N_A(T) + P_{sleep}(T - wN_A(T)) \tag{4.1}$$

We can rewrite this equation as

$$E_C(T) = (P_{adv} - P_{sleep}) w N_A(T) + P_{sleep} T \tag{4.2}$$

Here, P_{adv} and P_{sleep} are dissipated powers in advertising and sleep modes, respectively.

Since we only plan to consume the harvested energy, up to time T, the relation between $E_C(T)$ and $H(T)$, harvested energy curve, is as

$$E_C(T) \leq H(T) \tag{4.3}$$

Then, using Equation (4.2) and (4.3), we achieve

$$N_A(T) \leq \frac{H(T) - P_{sleep} T}{(P_{adv} - P_{sleep}) w} \tag{4.4}$$

Let us call the maximum number of packets that can be transmitted up to time T as $N_{max}(T)$. According to Equation (4.4), we can reach $N_{max}(T)$ when we consume all the harvested energy up to time T.

Since $N_A(T)$ will always be an integer, the upper limit of Equation (4.4) can be redefined using integer valued $N_{max}(T)$ as

$$N_{max}(T) = \left\lfloor \frac{H(T) - P_{sleep}T}{(P_{adv} - P_{sleep})w} \right\rfloor \tag{4.5}$$

Then, we find the solution of the first part of our problem, maximum number of packets that can be transmitted. Now, we should consider scheduling. As expressed before, in order to maximize the throughput, the device in scan mode should capture maximum number of packets and for that purpose, advertising intervals should be chosen properly.

Let us consider the devices in scan mode near S. We can say that these devices do active scanning by turning on and off their receivers at random times. Let us say that these scanning intervals are distributed evenly. Let these scanning intervals be defined for the time interval (u, v) and their maximum length be W_{max}. Figure 4.1 shows two consecutive scanning intervals starting at u_1 and u_2, and three possible end times of the second scanning interval, namely v_{21}, v_{22} and v_{23}. As seen from the figure, probability of v_2 being in the interval $(u_2, u_2 + W_{max})$ is 1 and v_2 occurring at any instant within that interval is equiprobable.

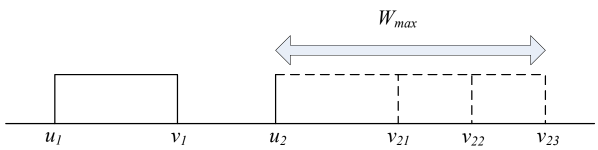

Figure 4.1: Time line of two different scanning intervals. The second scanning interval starting at u_2 can finish at any time within W_{max}.

Let us call the event of $(k+1)^{th}$ packet not being captured in the i^{th} scan interval as A and try to find its probability $P(A)$. Figures 4.2 and 4.3 illustrate this for a scan interval with a defined onset. As seen from Figure 4.2, if $t_{(k+1)} - u_i > W_{max}$; then, $P(A) = 1$, where $t_{(k+1)}$ is the transmission time of $(k+1)^{th}$ packet. Because, it is certain that the

i^{th} scan interval will end before the transmission of $(k+1)^{\text{th}}$ packet. On the other hand, as seen from Figure 4.3, if $t_{(k+1)} - u_i \leq W_{max}$ then $P(A) = (t_{(k+1)} - u_i)/W_{max}$, since the probability of v_i being in the interval $(u_i, t_{(k+1)})$ is equal to $(t_{(k+1)} - u_i)/W_{max}$. Then, we can write

$$P(A) = \min(1, \frac{t_{(k+1)} - u_i}{W_{max}}) \tag{4.6}$$

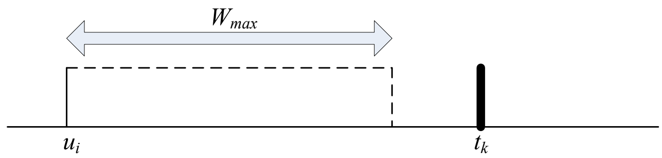

Figure 4.2: The k^{th} packet with transmission time t_k and a scanning interval with a defined onset, u_i, scenario with no capturing probability.

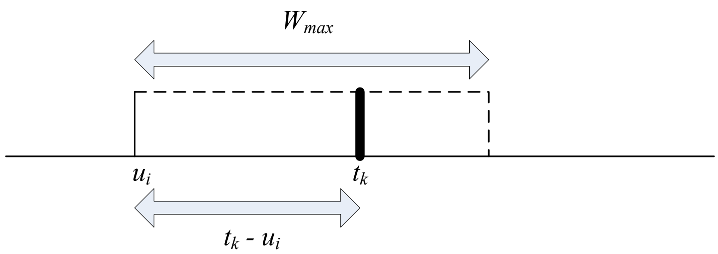

Figure 4.3: The k^{th} packet with transmission time t_k and a scanning interval with a defined onset, u_i, scenario with capturing probability.

Now, let us say the onset of the i^{th} scan interval can be at any point between the transmission times of k^{th} and $(k + 1)^{\text{th}}$ packets, i.e., $u_i \in (t_k, t_{k+1})$. Then, we can write

$$P(A) = \frac{1}{t_{k+1} - t_k} \int_{t_k}^{t_{k+1}} \min(1, \frac{t_{k+1} - u_i}{W_{max}})\, du_i \tag{4.7}$$

In this step, let the time line of transmitted packets until a given time T be as in Figure 4.4. Let the onset of the i^{th} scan interval u_i occur equiprobably in the interval $(0, T)$. Then, using the Equation 4.7, we can write

$$P(A) = \frac{1}{T} \sum_{k=1}^{N_A(T)-1} \int_{t_k}^{t_{k+1}} \min(1, \frac{t_{k+1} - u_i}{W_{max}})\, du_i \tag{4.8}$$

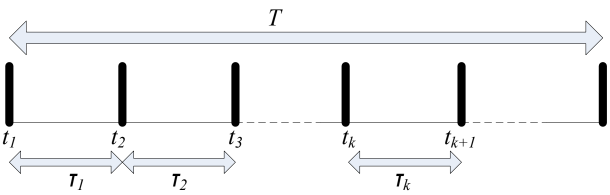

Figure 4.4: Illustration of a scheduling example of the transmitted packets up to time T. The advertising interval after k^{th} packet is denoted as τ_k.

Let the k^{th} advertising interval be $\tau_k = t_{k+1} - t_k$, as in Figure 4.4. In our model, we assume that no τ_k is larger than W_{max}. Then, the condition in Figure 4.2 do not occur and the Equation 4.8 can be rewritten as

$$P(A) = \frac{1}{T} \sum_{k=1}^{N_A(T)-1} \int_{t_k}^{t_{k+1}} \frac{t_{k+1} - u_i}{W_{max}} \, du_i \tag{4.9}$$

and if we solve the integral, we achieve

$$P(A) = \frac{1}{T} \sum_{k=1}^{N_A(T)-1} \frac{\tau_k^2}{2W_{max}} \tag{4.10}$$

In the scheduling problem, we want $P(A)$ to be minimum, since we want to capture as many packets as possible. For that purpose, we are looking for τ_ks that minimize $P(A)$, given that $N_A(T)$ is equal to $N_{max}(T)$. Then, the problem can be defined as

$$\arg\min_{\tau_k} \sum_{k=1}^{N_{max}(T)-1} \tau_k^2 \tag{4.11}$$

4.3 Optimum Solution

The value above to be minimized can be defined as [24]

$$Q_{N_{max}(T)} = \sum_{k=1}^{N_{max}(t)-1} \tau_k^2 \tag{4.12}$$

Lemma 4.3.1 *The values of τ_k should be equal in order to minimize $Q_{N_{max}(T)}$.*

Proof. Let us consider the scheduling of n^{th} packet. Let its transmission time be $t_n \in (t_{n-1}, t_{n+1})$, as illustrated in Figure 4.5. Now, let us change its transmission time from t_n to t'_n, trying to optimize the packet scheduling. Then, the change in $Q_{N_{max}(T)}$ would be

$$\Delta Q_{N_{max}(T)} = (t'_n - t_{n-1})^2 + (t_{n+1} - t'_n)^2 - (t_n - t_{n-1})^2 - (t_{n+1} - t_n)^2 \tag{4.13}$$

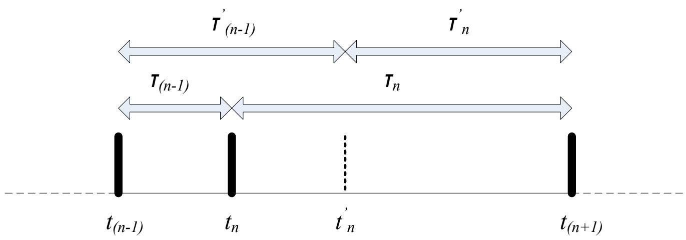

Figure 4.5: The aim is to reschedule n^{th} advertisement packet optimally and minimize $Q_{N_{max}(T)}$.

If the new transmission time of the n^{th} packet, t'_n, is the optimum, then $\Delta Q_{N_{max}(T)}$ should be as negative as possible. Due to convexity, to find the value that minimizes $\Delta Q_{N_{max}(T)}$, we should take the derivative of the right hand side of Equation 4.13 with respect to t'_n, and set it to zero. Then, we find the optimum t'_n as

$$t'_n = \frac{t_{n+1} + t_{n-1}}{2} \tag{4.14}$$

which is satisfied by $\tau_{n-1} = \tau_n$. If we continue this process by rescheduling all packets, it yields to all equal τ_ks. Thus, in order to schedule all packets optimally, the values of τ_k should be equal. ■

Since the problem is to maximize the data rate of S, our first step is to maximize $N_A(T)$. From that, we get $N_{max}(T)$. The next step is to capture as many packets as possible. For that purpose, we should schedule advertising intervals optimally, which corresponds to minimizing $Q_{N_{max}(T)}$. As expressed in Lemma 4.3.1, the solution to $\arg\min_{\tau_k} Q_{N_{max}(T)}$ is evenly distributed τ_ks. If we define $E(t)$ as the transmitted energy curve, that corresponds to a line that connects start and end points of energy harvesting curve. The reason for that is, in order to reach $N_{max}(T)$, the module should consume all the harvested energy. In addition, $E(t)$ is a straight line due to the fact that when τ_ks are equal, the module consumes the energy at a constant rate, which corresponds to a constant slope. On the other hand, as τ_ks decrease with increasing ks, since the module will be on advertising mode for longer durations, the slope of $E(t)$ increases and likewise, as τ_ks increase with increasing ks, the slope of $E(t)$ decreases.

However, equal τ_ks may not always be possible, as for convex energy harvesting curves; we may exceed the curve when we draw the line. That is not possible since we cannot exceed harvested energy. In that case, even though we cannot distribute τ_ks evenly, we can distribute them as close as possible. For that purpose, we can use "Stretched String Method", to find optimum τ_ks [24].

The Stretched String method can be visualised as this: If we tie one end of a string to the origin and connect it to the point $(T, H(T))$ tightly while constraining it to lie under $H(t)$, this string gives us the optimal transmitted energy curve, $E_{opt}(T)$ [26].

An algebraic expression for the derivative of $E_{opt}(t)$, where $t \in (0, T)$ can be the following:

$$\frac{dE_{opt}(t)}{dt} = \min_{\tau \in (t,T)} \frac{H(\tau) - E_{opt}(t)}{\tau - t} \tag{4.15}$$

Some examples of harvested energy curves and corresponding $E_{opt}(t)$s are given in Figures 4.6, 4.7, 4.8 and 4.9. An important point here is that $H(t)$s are continuous nondecreasing functions.

In Figure 4.6, $H(t)$ is a linear function. If we use the Stretched String Method, when we bring an imaginary string from below the harvested energy curve, the string follows the line. Then, we can choose all τ_ks equally, without exceeding the harvested energy curve and resulting consumed energy curve $E_{opt}(t)$ would become a line that overlaps with $H(t)$.

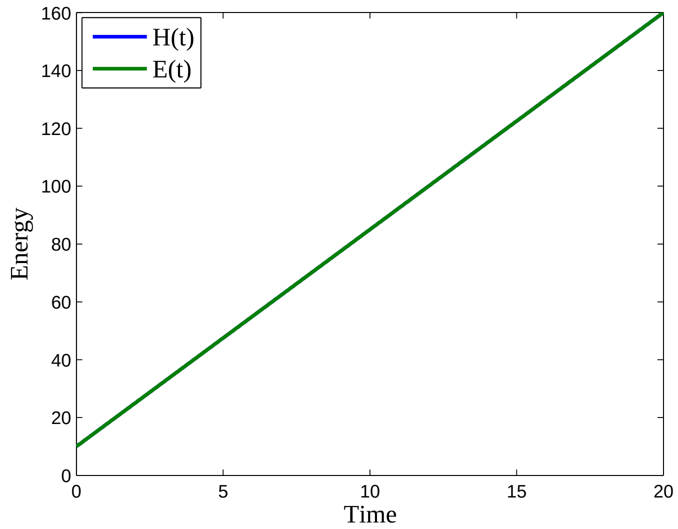

Figure 4.6: Consumed Energy Curve for Linear Harvested Energy Curve

However, for convex energy harvesting functions, we cannot make all τ_ks equal, without exceeding the function. An example to this is shown in Figure 4.7.Here, if we connect the starting and end points of $H(t)$, the resulting line would be above $H(t)$. However, since we cannot consume more than the harvested energy at any time, that consumed energy curve is not possible. If we use the Stretched String Method, when we bring an imaginary string from below the harvested energy curve, the string follows the curve at all points. Thus, once again, harvested and consumed energy curves overlap. Since $H(t)$ is convex, as its slope increases with increasing t, we can say that τ_ks decrease with increasing k.

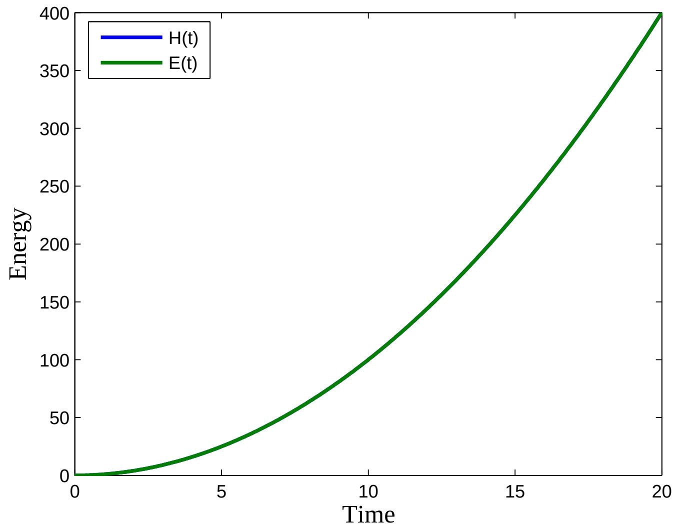

Figure 4.7: Consumed Energy Curve for Convex Harvested Energy Curve

An example to a concave harvested energy curve is as Figure 4.8. Here, when we connect the starting and end points of $H(t)$, the resulting line do not cross $H(t)$ at any point. Thus, if we use the Stretched String Method, when we bring an imaginary string from below the harvested energy curve, the string draws a line. Then, we infer that for that kind of harvested energy curve, the module should consume energy at a constant rate.

Certainly, there are also some functions that consist of the combination of both convex and concave curves. An example to that kind of harvested energy curve is as Figure 4.9. Here, if we connect the starting and end points of $H(t)$, the resulting line exceeds $H(t)$ at the beginning, but then it lies below $H(t)$. Thus, if we use the Stretched String Method, when we bring an imaginary string from below the harvested energy curve, the string follows the curve up to around time $t = 9$, and then draws a straight tangent line that ends at the end point of $H(t)$. Therefore, in that case, we can say that the packets are sent with an increasing rate at first and after some point, the rate remains constant.

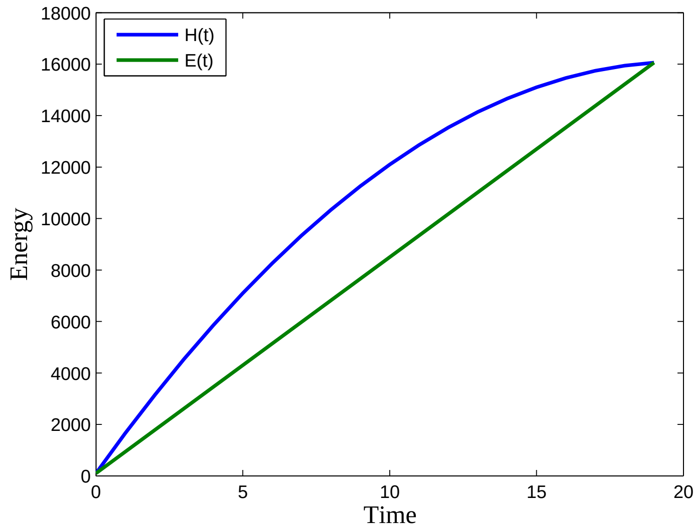

Figure 4.8: Consumed Energy Curve for Concave Harvested Energy Curve

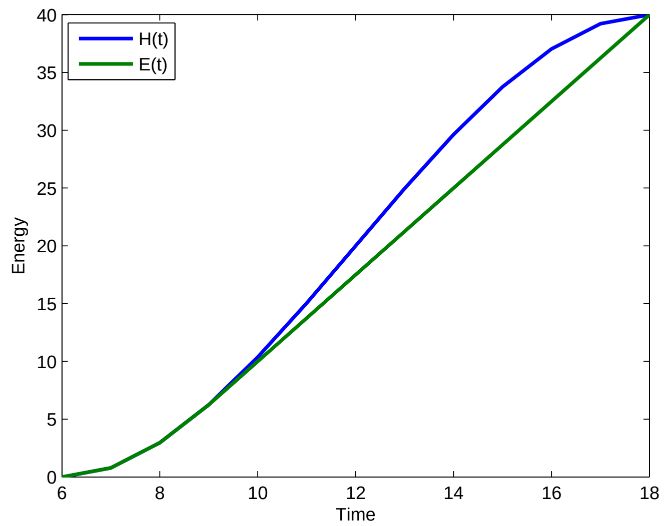

Figure 4.9: Consumed Energy Curve for Hybrid Harvested Energy Curve

CHAPTER 5

IMPLEMENTATION RESULTS

5.1 Application Development Environment

5.1.1 BLE113 Bluetooth Smart Module

BLE113 is a Bluetooth Smart module that is designed for small and low power sensors and accessories. It combines all necessary features for a Bluetooth Smart application which are Bluetooth radio, software stack and GATT based profiles. As the Bluetooth radio, AT3216 Series Multilayer Chip Antenna, which is a small, low-profile and light-weight monolithic SMD and have wide bandwidth, is used. BLE113 Bluetooth Smart module can also run user applications, which means no external microcontroller is needed in constrained devices. The module has flexible hardware interfaces that enables connecting to different peripherals and sensors. It can be powered up directly from a standard 3V coin cell battery or pair of AAA batteries and in lowest power sleep mode it consumes only 500nA. In addition, according to datasheet, it consumes 18.2 mA when the transmit power is 0 dBm and in receive mode, it consumes 14.3 mA. BLE113 Development Kit is primarily designed for engineers developing Bluetooth 4.0 systems utilizing BLE113 modules which can be seen from the upper right corner of the Figure 5.1.

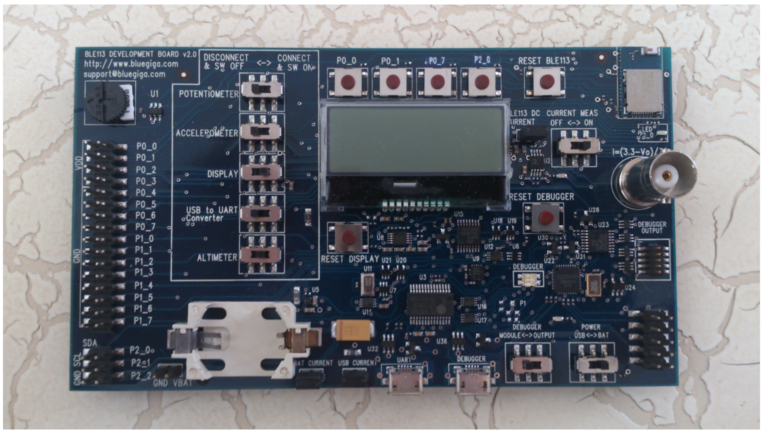

Figure 5.1: BLE113 Development Kit

Bluetooth low energy applications generally have a server and a client, where the server is the device that provides the information like the sensor devices such as thermometers or heart rate sensors, and client is the device that collects the information from one or more sensors and typically either displays it to the user or passes it forward. Client devices usually do not implement any service, and just collect the data provided by the server devices. They are typically devices like mobile phones, tablets and PCs. Figure 5.2 shows the relationship of these two roles.

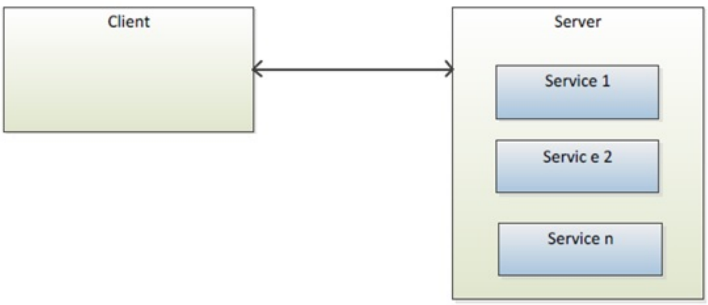

Figure 5.2: Bluetooth low energy device roles

BLE113 Development Kit includes both server and client modes. It implements all Bluetooth Smart functionalities such as GAP, L2CAP, ATT, GATT , Security manager: bonding, encryption and Bluetooth Smart profiles. From Figure 5.3, we can see the layers of Bluetooth Smart software architecture. With the help of Generic Access Profile(GAP) layer, we can manage discoverability and connetability modes and open connections and adjust advertising interval values according to our needs.

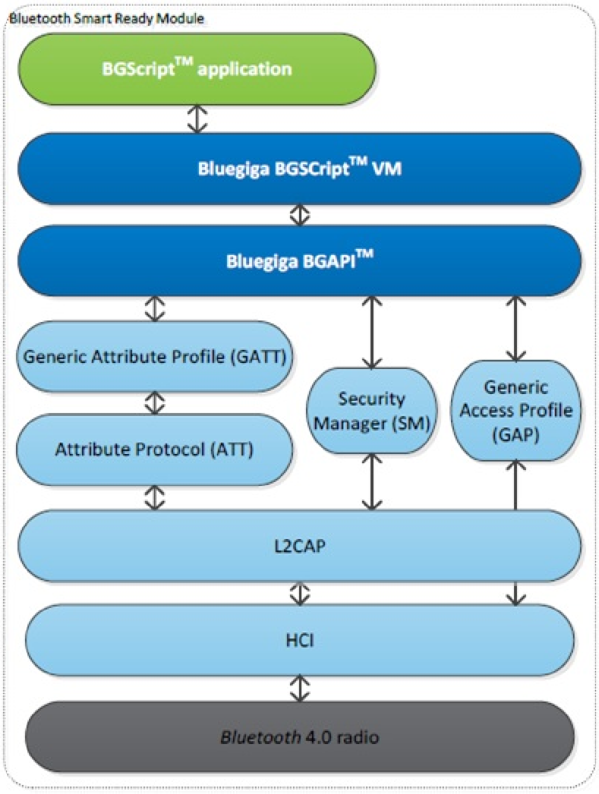

Figure 5.3: Layers of Bluetooth Smart Software Architecture

BLE113 Development Kit supports standalone applications as well. Our application is executed on the on-board 8051 processor. The application we use during the tests is monitoring the battery while constantly broadcasting advertisement packets with advertising intervals changing according to our algorithm. In order to calculate the battery level, we read the voltage values supplied to the module by using the A/D converter that is directly connected to the BLE113 via the IO interfaces and send it to another Bluetooth module. For that reason, BLE113 Development Kit is run on server mode.

Other software tools we use when developing our application are BGScriptTM, a simple scripting language for writing applications, and Blutoooth Smart Profile ToolkitTM which is an XML based development tool for Bluetooth Smart profiles.

In order to collect the data sent by BLE113 Development Kit, we use BLED112 USB dongle in client mode. In addition, in order to inspect the collected data in the computer, we used BLEGUI. BLEGUI can be used to control BLED112 over USB and it is a terminal-like program that sends the commands to the Bluetooth Smart module and displays the received responses on the screen as logs. With the help of these logs, we can parse the required test values such as the number of captured advertisement packets, capture times and battery levels. We can see BLED112 in Figure 5.4.

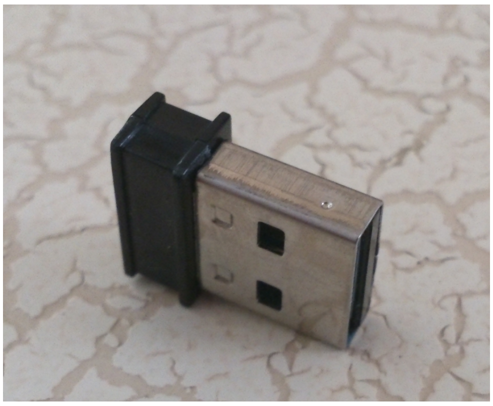

Figure 5.4: BLED112 USB Dongle

5.1.2 Battery

During the tests, we use two AA NiMH rechargeable batteries and we monitor the voltage values that they supply to the module. We can see these batteries' visuals in Figure 5.5 and graphs that show the typical discharge profile in Figure 5.6.

Figure 5.5: AA NiMH Rechargeable Batteries

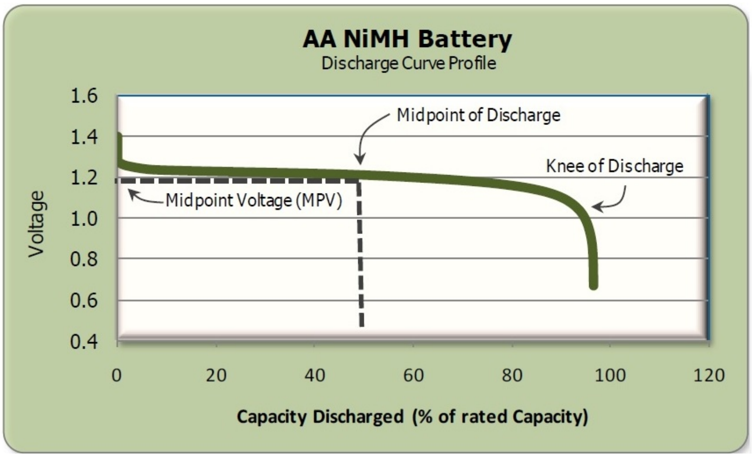

Figure 5.6: Typical Discharge Profile [20]

5.1.3 Solar Panel

The solar panels that we use on our tests can harvest at most 4.5 V and it can be seen in Figure 5.7.

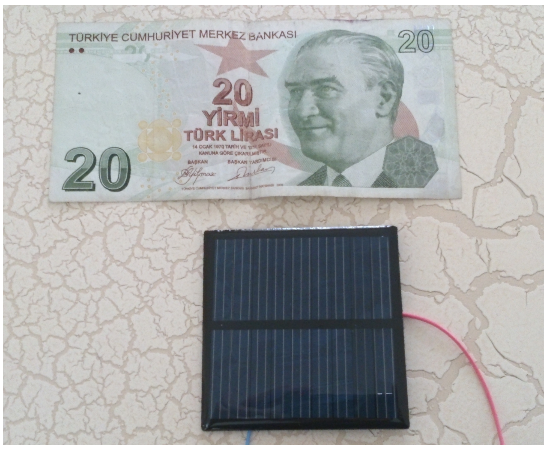

Figure 5.7: Solar Panel

The circuit that we use to integrate the solar panels and batteries to the system can be seen from Figure 5.8. The reason why we connect the panels in parallel when we use more than one is because we need more supply current, rather than more voltage.

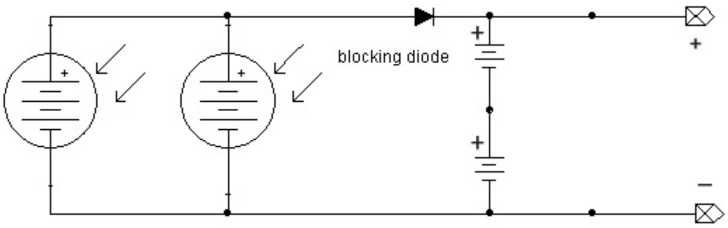

Figure 5.8: Integration Circuit of the Batteries and Solar Panels

5.1.4 Test Environment

The overall test environment that consists of rechargeable batteries, solar panels, BLE113 Development Kit, BLED112 and a computer to run BLEGUI and process the collected data, can be seen from Figure 5.9.

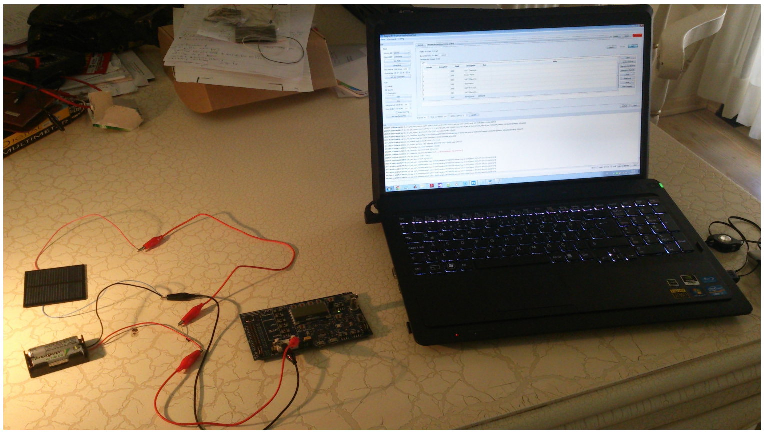

Figure 5.9: Test Environment

5.2 Implementation Results

First, we have powered up the module only by using batteries, without solar panels. We have observed how the system works without the integrated solar panels, by changing the advertising intervals between transmitted packets. During the tests, we have used all 3 advertisement channels. The average inter-arrival times between the packets captured by BLED112 are shown in Figure 5.10, for different advertising intervals. Capturing times change linearly for smaller advertising interval values; whereas, they form a convex-like shape as the advertising intervals become longer. While collecting these results, we have also observed the energy levels of the batteries. It is critical for our algorithm to know the consumed battery power, for various advertising intervals. For that purpose, we have investigated the average voltage drop per minute of the battery for different advertising intervals, as seen from Figure 5.11.

In this section, we aim to consume the harvested energy most efficiently by adjusting the advertising interval values. From the figure, we see that, as the advertising intervals become longer, consumed energy decreases exponentially. Thus, advertisement packet scheduling has a significant effect on battery lifetime. This is an important result, that increases the significance of this work, since the theoretical model developed in the previous chapter assumes that the advertising interval values have the major effect on energy consumption. The convexity in Figure 5.11 verifies this assumption. In addition, as seen from Figure 5.10, as the advertising interval values increase, the number of captured packets decrease linearly; whereas, the consumed energy decreases exponentially with a large exponent. This can be interpreted as a small change in advertising interval value results in a considerable effect on system performance. As a result, we can say that Figure 5.11 is a crucial point of this study.

In Figures 5.10 and 5.11, since we do not work with solar panels, the results are not location-based. Thus, the tests conducted using BLE113 Development Kit will always give similar results. However, the average voltage drop per minute in Figure 5.11 depends on battery type. The capacities of the batteries used in our tests are 2000 mAh. If batteries with smaller capacities were used in our tests, Figure 5.11 would be more convex.

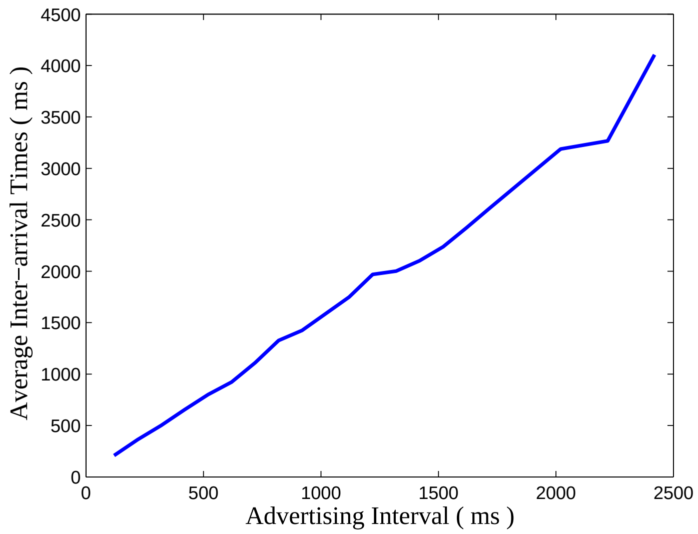

Figure 5.10: Average inter-arrival times between advertising packets versus advertising intervals, where BLE113 uses all three advertisement channels.

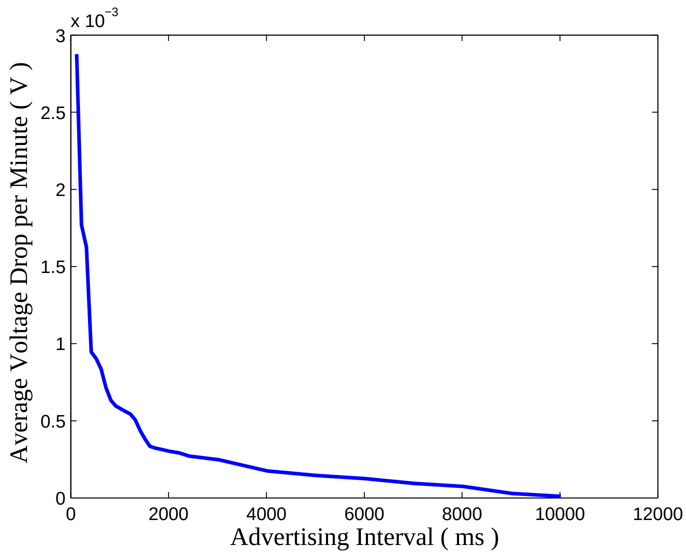

Figure 5.11: Average voltage drop per minute versus advertising intervals, where BLE113 uses all three advertisement channels.

Then, we have connected solar panels to our system and started testing the advertising scheduling we developed. First, we have tried to extract the energy profiles in the test environment. For that purpose, we have recharged the batteries using solar panels. After these, we have achieved harvested energy curves, and using those and the Stretched String Method, we have reached consumed energy curves. Then, using the results in Figure 5.11, we have determined the advertising interval values to be used on each time. After installing these values to BLE113, we rerun the test under the same environmental conditions. Then, we have achieved the graph of number of packets that BLED112 captured with respect to time. We have also monitored the voltage levels of the batteries that powered up BLE113.

In addition, we have determined some consumed energy curves that did not follow a specific pattern except for connecting the start and end points of the harvested energy curves that we achieved. That is, we randomly chose advertising intervals for different time segments during the tests. Then, we have used the advertising interval values that we achieved using these consumed energy curves and called it random scheduling. After that, we have compared the number of captured packets and battery voltage levels we achieved with both scheduling schemes.

We have conducted our first test under florescent light, using one solar panel. The resulting harvested energy curve and consumed energy curve determined according to Stretched String Method are shown in Figure 5.12; whereas, the consumed energy curve corresponding to random scheduling is shown in Figure 5.13. In the scheme in Figure 5.13, consumed energy exceeds harvested energy after 20 minutes. Here, for the first 20 minutes, advertising intervals have been set to 756 ms, and after 20 minutes, they have been increased up to 3606 ms. On the other hand, for Figure 5.12, these values have been 1638 ms for the first hour and 1493 ms for later hours.

Figure 5.14 and 5.15 show the change in the battery voltage level during two different tests. As seen here, the scheduling that is closer to energy neutral operation is the one shown in Figure 5.14. Here, the maximum change in the battery voltage level is around 0.009 V. On the other hand, in Figure 5.15, this change is larger than 0.04 V. That is because, as seen from Figure 5.13, that scheduling tries to consume more energy than the harvested, between 20th and 60th minutes.

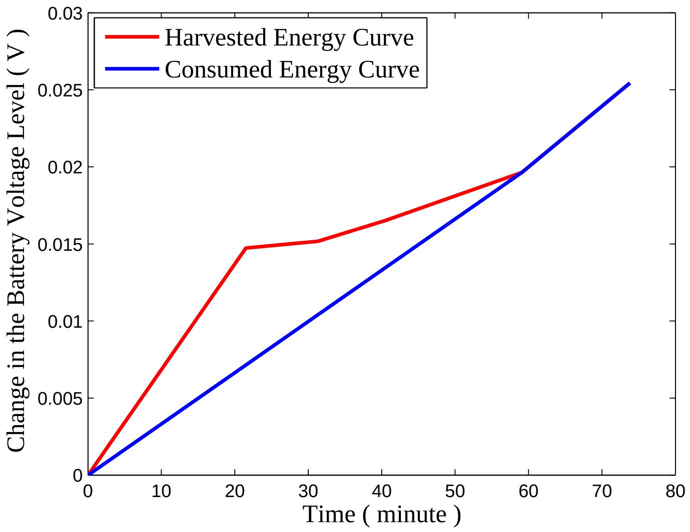

Figure 5.12: Change in the battery voltage level versus time. Harvested energy curve achieved by using one solar panel indoors and the consumed energy curve determined according to Stretched String Method.

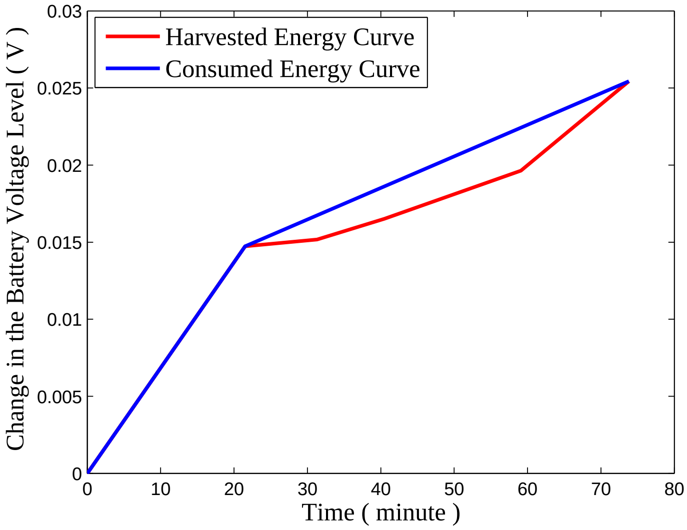

Figure 5.13: Change in the battery voltage level versus time. Harvested energy curve achieved by using one solar panel indoors and the consumed energy curve determined according to random scheduling.

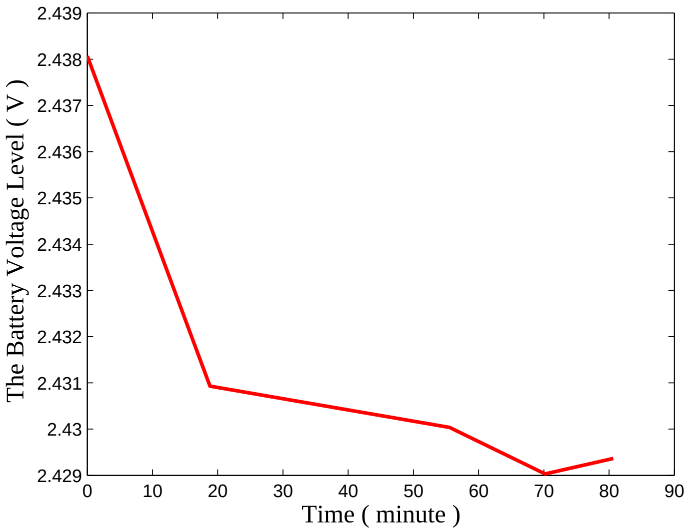

Figure 5.14: Battery voltage level versus time. The effect of the advertising scheduling determined using Stretched String Method on the battery recharged indoors by one solar panel.

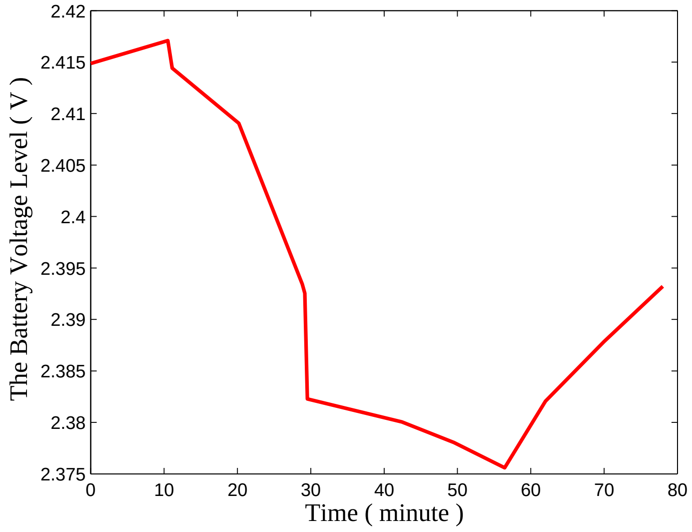

Figure 5.15: Battery voltage level versus time. The effect of random scheduling on the battery recharged indoors by one solar panel.

Figure 5.16 and 5.17 show the effects of these 2 schedulings on throughput. The packets captured by BLED112 are almost equal in two cases. However, in Figure 5.16, we can see that the captured packets are distributed evenly during the test. Whereas, in Figure 5.17, the throughput drops significantly, after consuming excessive energy in the first 20 minutes. This is an important parameter to design reliable applications.

Our second test have been conducted under florescent light, using two solar panels. According to the graph in Figure 5.18, for the first 55 minutes, advertising intervals have been set to 1387 ms, and after that, they have been decreased to 1036 ms. On the other hand, in Figure 5.19, these values have been 4390 ms for the first 75 minutes and 80 ms for the later hours.

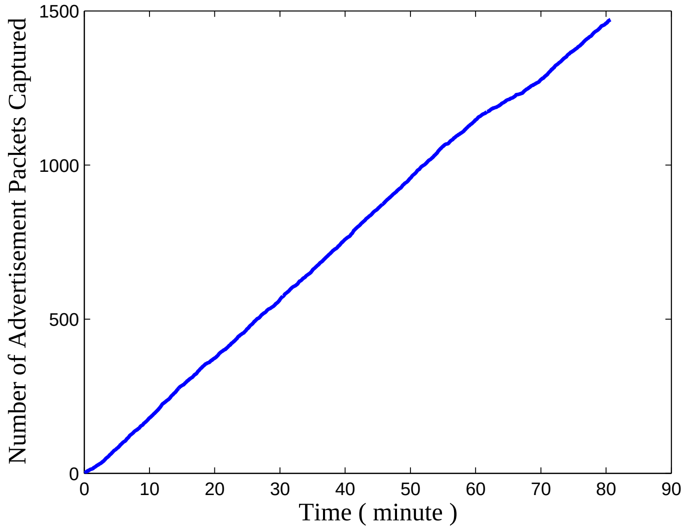

Figure 5.16: Number of advertisement packets captured by BLE112 USB Dongle versus time. The effect of the advertising scheduling determined using Stretched String Method on the throughput of the system powered up indoors by one solar panel.

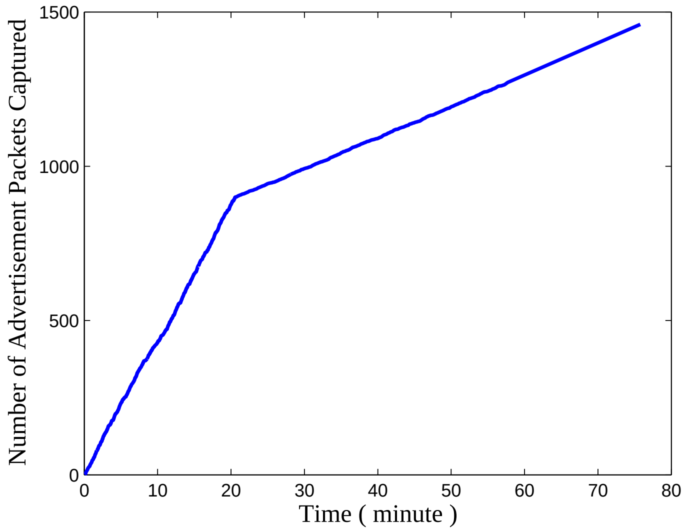

Figure 5.17: Number of advertisement packets captured by BLE112 USB Dongle versus time. The effect of random scheduling on the throughput of the system powered up indoors by one solar panel.

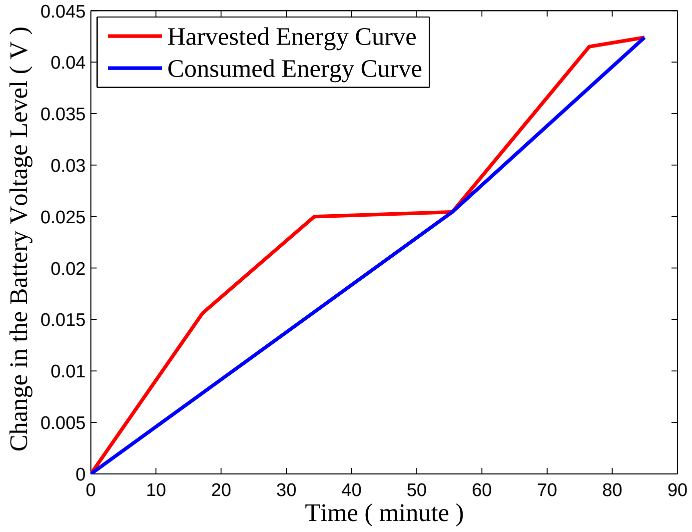

Figure 5.18: Change in the battery voltage level versus time. Harvested energy curve achieved by using two solar panels indoors and the consumed energy curve determined according to Stretched String Method.

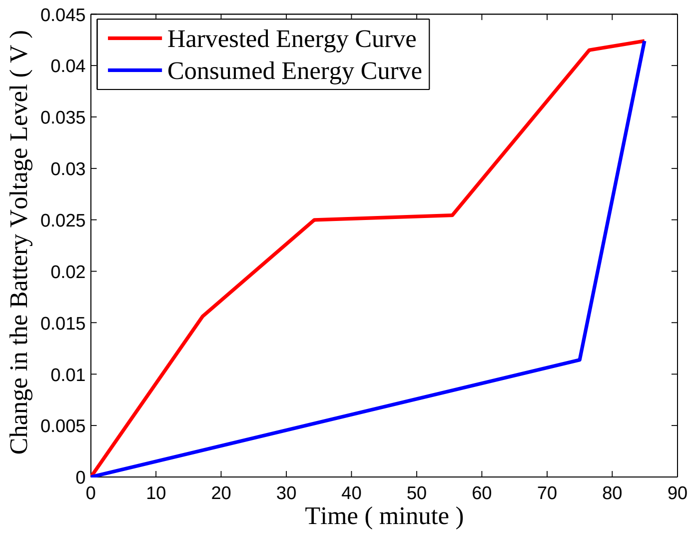

Figure 5.19: Change in the battery voltage level versus time. Harvested energy curve achieved by using two solar panels indoors and the consumed energy curve determined according to random scheduling.

Figure 5.20 shows the monitored battery voltage using the advertising scheduling determined in Figure 5.18. By this graph with the maximum voltage difference of

0.01 V, we can say that we have achieved energy neutral operation. Also, the number of captured packets are shown in Figure 5.22. Here, we see that we have achieve a throughput that is distributed homogeneously on the time axis. If we compare that graph with Figure 5.16, where we used one solar panel, we observe that 500 more packets are transmitted within 80 minutes with two solar panels, and that corresponds to a %50 improvement.

On the other hand, in Figure 5.21, the maximum voltage difference is 0.02 V. From Figure 5.23, we can see that for the first 75 minutes, very little number of packets are captured and the extra energy is stored into the battery. After that time, when we start to transmit packets with advertising intervals of 80 ms, battery voltage drops quickly but the number of captured packets increase significantly. However, for many applications, the throughput in the first 75 minutes may not be adequate.

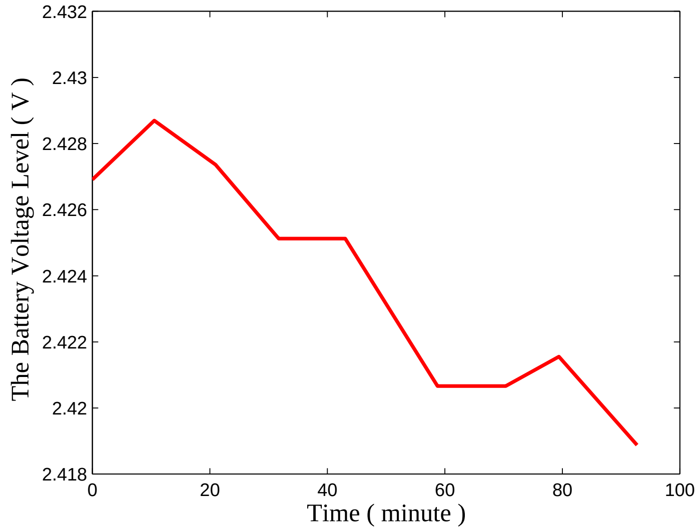

Figure 5.20: Battery voltage level versus time. The effect of the advertising scheduling determined using Stretched String Method on the battery recharged indoors by two solar panels.

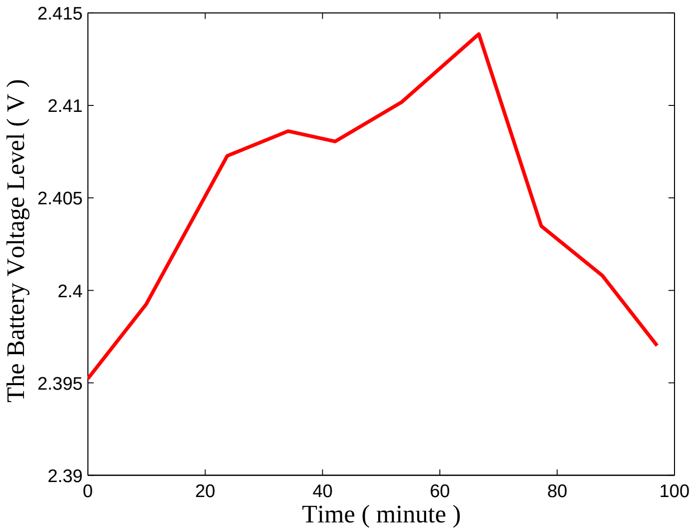

Figure 5.21: Battery voltage level versus time. The effect of the random scheduling on the battery recharged indoors by two solar panels.

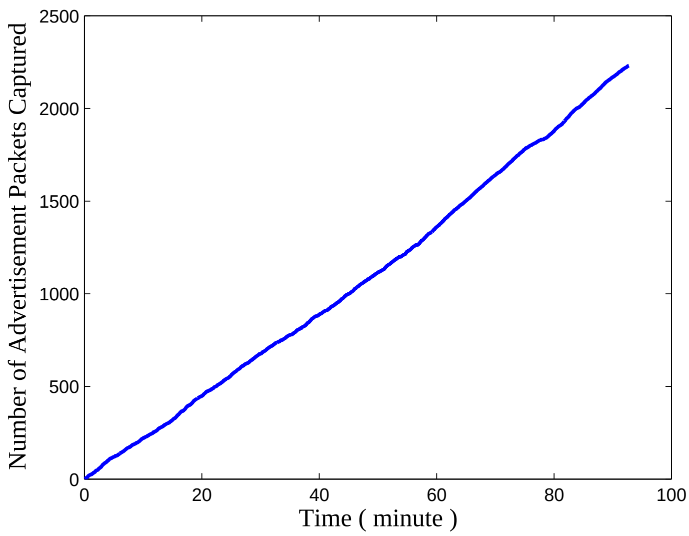

Figure 5.22: Number of advertisement packets captured by BLE112 USB Dongle versus time. The effect of the advertising scheduling determined using Stretched String Method on the throughput of the system powered up indoors by two solar panels.

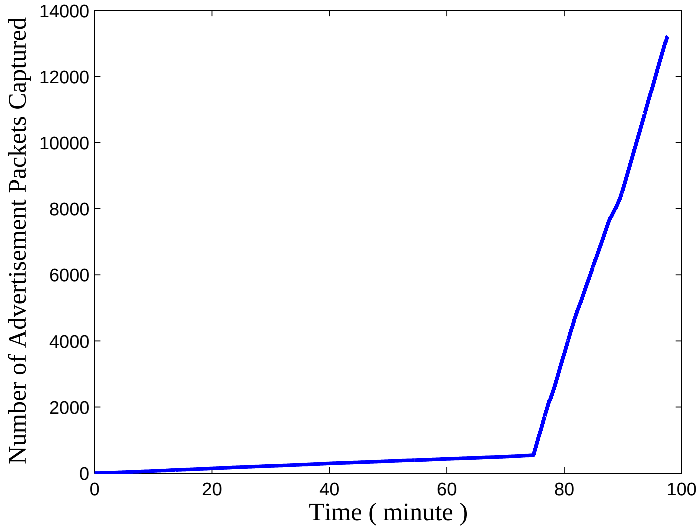

Figure 5.23: Number of advertisement packets captured by BLE112 USB Dongle versus time. The effect of the random scheduling on the throughput of the system powered up indoors by two solar panels.

Our next test has been conducted outdoors. Different from indoors, since we have not use florescent lights that time, energy that can be harvested from the medium can change throughout the test due to weather conditions. In addition, harvestable energy can also change according to the time of the day. For the results to be consistent, we have collected them in consecutive hours.

We have achieved Figure 5.24 from the tests we conducted using 1 solar panel. The voltage level of the battery has increased 1 V after one hour. The advertising intervals determined according to the Stretched String Method have been found as 1570 ms for the first 28 minutes, 655 ms for the next 10 minutes and 25 ms for later on. On the other hand, Figure 5.25 shows the constant rate of consumed energy, determined according to random scheduling. According to random scheduling, advertising intervals have been selected as 335 ms.

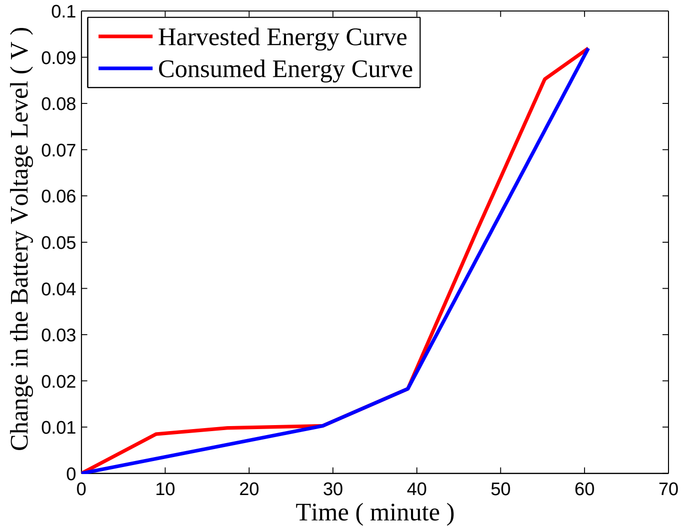

Figure 5.24: Change in the battery voltage level versus time. Harvested energy curve achieved by using one solar panel outdoors and the consumed energy curve determined according to Stretched String Method.

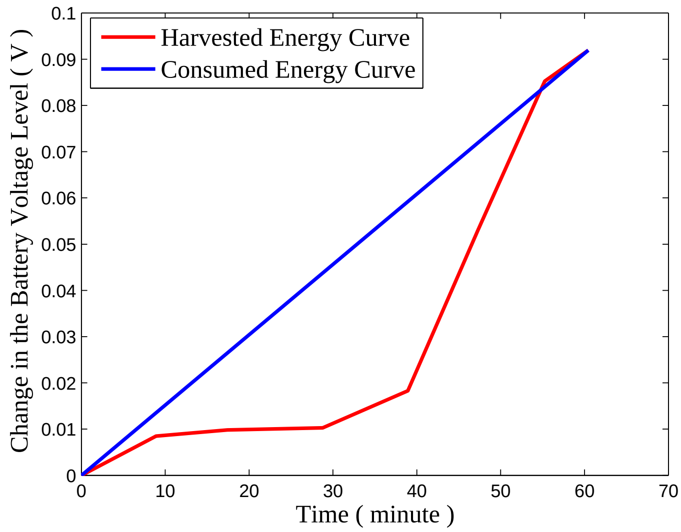

Figure 5.25: Change in the battery voltage level versus time. Harvested energy curve achieved by using one solar panel outdoors and the consumed energy curve determined according to random scheduling.

The results achieved using Figure 5.24 are shown in Figures 5.26 and 5.28. As seen here, the number of captured packets have increased by a factor of ten, compared to indoor tests. Figure 5.26 shows that we have almost provided energy neutral operation. In order to do that, we should transmit packets at a lower rate for the first 40 minutes and then increase this rate as the harvested energy increases. This way, we can maximize the throughput while ensuring energy neutral operation.

On the other hand, the results achieved using random scheduling are shown in Figures 5.27 and 5.29. We can see that the number of captured packets has decreased by %80, compared to the scheduling achieved using Stretched String Method. In addition, Figure 5.27 shows that the battery voltage level has dropped 0.08 V. As a result, we can say that random scheduling is far from providing energy neutral operation.

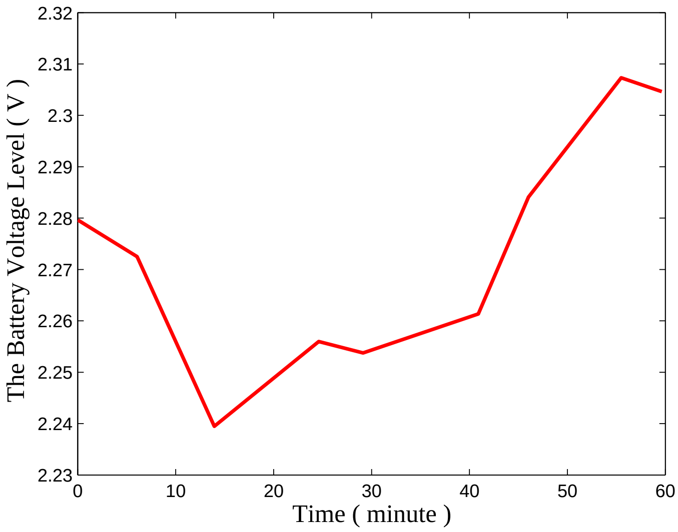

Figure 5.26: Battery voltage level versus time. The effect of the advertising scheduling determined using Stretched String Method on the battery recharged outdoors by one solar panel.

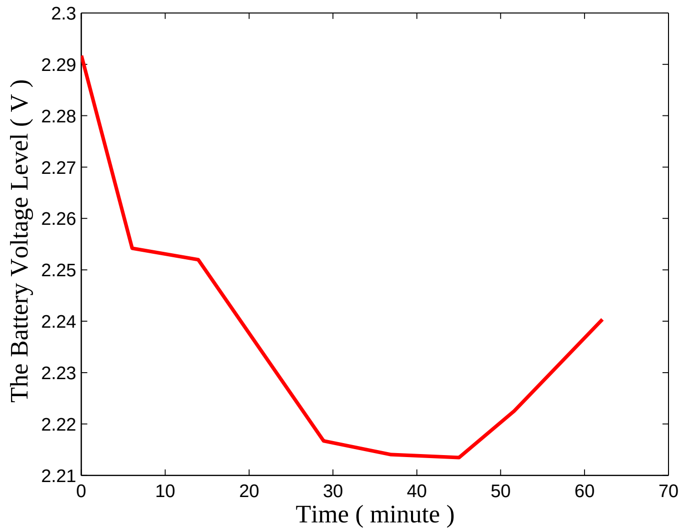

Figure 5.27: Battery voltage level versus time. The effect of the random scheduling on the battery recharged outdoors by one solar panel.

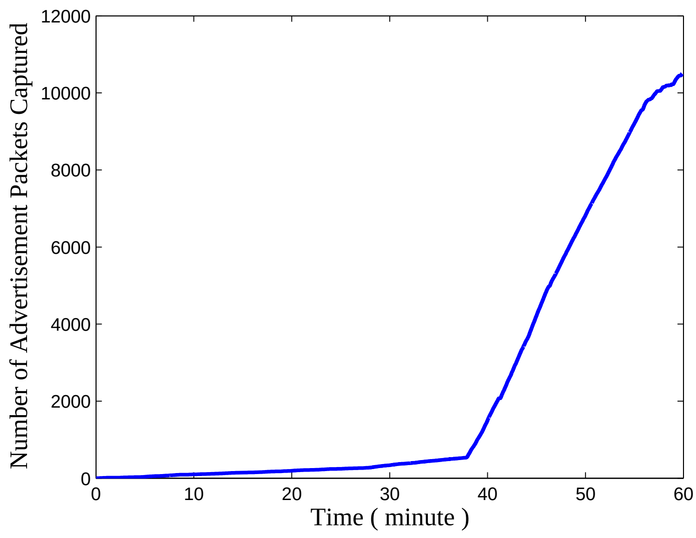

Figure 5.28: Number of advertisement packets captured by BLE112 USB Dongle versus time. The effect of the advertising scheduling determined using Stretched String Method on the throughput of the system powered up outdoors by one solar panel.

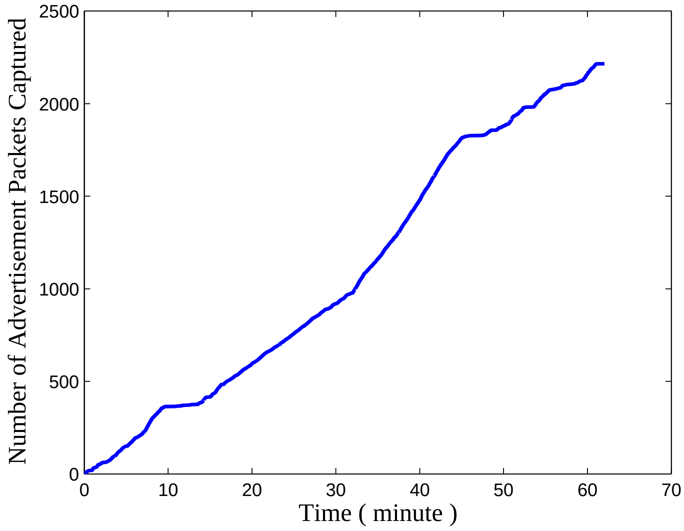

Figure 5.29: Number of advertisement packets captured by BLE112 USB Dongle versus time. The effect of the random scheduling on the throughput of the system powered up outdoors by one solar panel.

After that, we have continued our outdoor tests using 2 solar panels. We have achieved Figure 5.30 from the tests we conducted using 2 solar panels. As seen from Figure 5.18, when we could only increase the battery voltage by 0.045 V in 80 minutes indoors, we were able to increase it by almost 0.3 V outdoors. Using this result, we have decided to use 25 ms as the advertising interval for 60 minutes. On the other hand, Figure 5.31 shows the consumed energy graph determined by random scheduling. According to that, advertising intervals would be 20 ms for the first 26 minutes and then they would increase up to 121 ms.

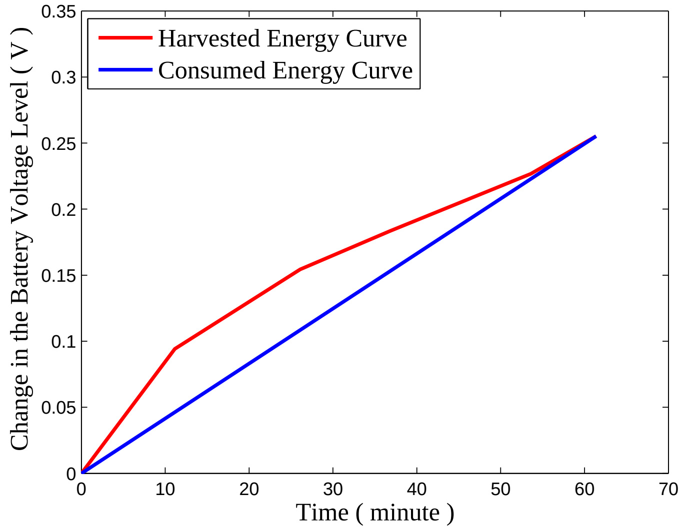

Figure 5.30: Change in the battery voltage level versus time. Harvested energy curve achieved by using two solar panels outdoors and the consumed energy curve determined according to Stretched String Method.

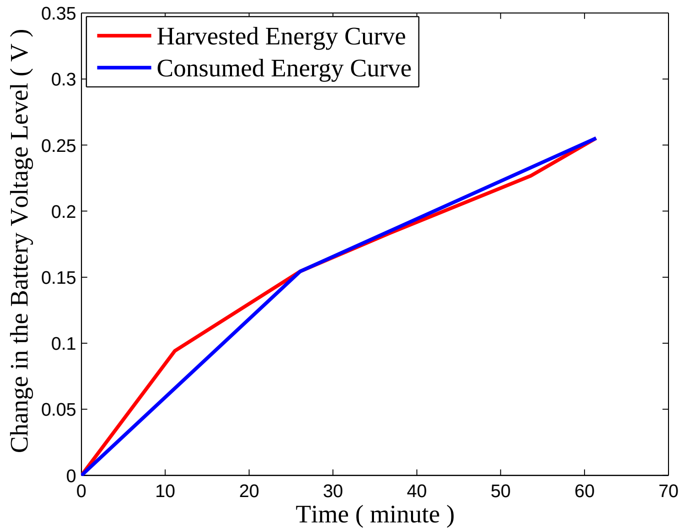

Figure 5.31: Change in the battery voltage level versus time. Harvested energy curve achieved by using two solar panels outdoors and the consumed energy curve determined according to random scheduling.

In the scheduling that we have achieved using Figure 5.30, since we have consumed much less energy than the harvested in the first 20 minutes, the extra energy was stored into the battery as seen from Figure 5.32. However, even though the rate of harvested energy dropped after 20 minutes, we have continued to transmit packets at the same rate, which yielded to a drop in the battery voltage level. On the other hand, since the consumed energy curve determined by the random scheduling have been closer to the harvested energy curve, the difference between the minimum and maximum voltage values in Figure 5.33 is smaller than that of Figure 5.32.

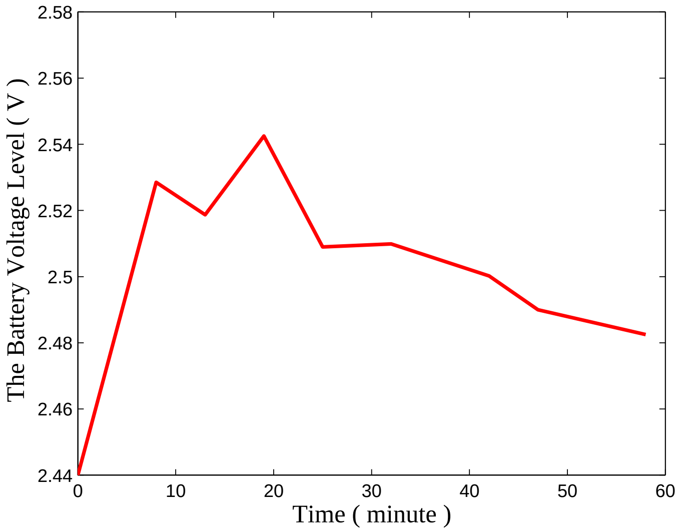

Figure 5.32: Battery voltage level versus time. The effect of the advertising scheduling determined using Stretched String Method on the battery recharged outdoors by two solar panels.

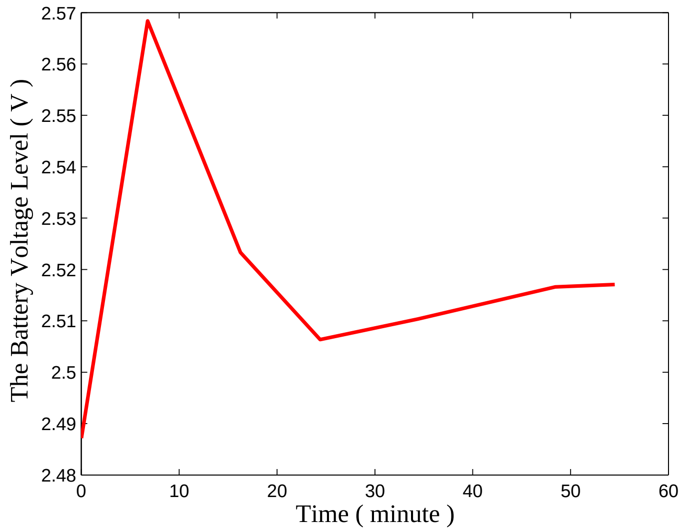

Figure 5.33: Battery voltage level versus time. The effect of the random scheduling on the battery recharged outdoors by two solar panels.

When we compare the two algorithms in terms of the number of captured packets, we observe that the scheduling determined using Stretched String Method outperforms the random scheduling. From Figure 5.34, we can see that we have captured almost 30000 packets using Stretched String Method; whereas, when random scheduling is used, we could only capture 13000 packets, as seen from Figure 5.35. Thus, even though the two algorithms have consumed almost equal energies, the difference between their throughput values is quite significant.

During the tests that we have conducted using two solar panels, we were able to reach the shortest advertising interval value, 20 ms. Thus, we have concluded that using 3 solar panels is unnecessary; since in that case, even if we transmitted at the maximum rate, the battery levels would continue to increase.

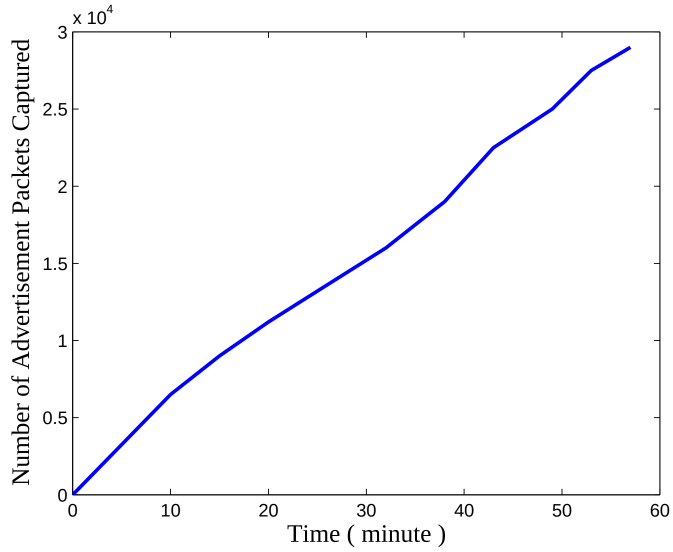

Figure 5.34: Number of advertisement packets captured by BLE112 USB Dongle versus time. The effect of the advertising scheduling determined using Stretched String Method on the throughput of the system powered up outdoors by two solar panels.

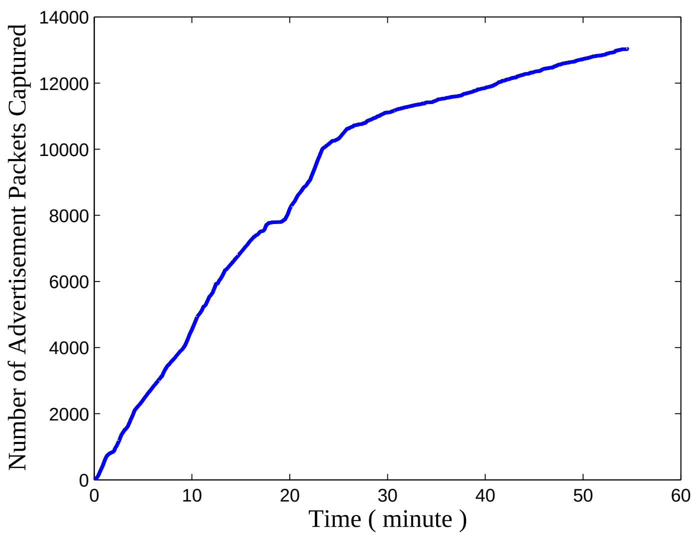

Figure 5.35: Number of advertisement packets captured by BLE112 USB Dongle versus time. The effect of the random scheduling on the throughput of the system powered up outdoors by two solar panels.

All tests that involve solar panels have been conducted once. The reason for that is each test requires 1.5 hours to complete. In addition, since the results of tests depend on the environmental conditions, two schedulings should be compared under the same conditions. However, it is hard to maintain the environmental conditions outdoors, which resulted in conducting the tests only once. That is why, the resulting graphs are not smooth. If these tests are conducted in longer time with more realizations, the reliability of them would increase.

CHAPTER 6

CONCLUSION

In this thesis, we have worked on duty cycle optimization for energy harvesting WSNs. A novel scheduling algorithm, EHPBS, based on energy predictions of wireless sensor nodes has been designed for TDMA based inra-cluster communications. In this approach, CH that uses EHPBS allocates the next time slot to the CM that expects to harvest most at that slot, and the non-transmitting nodes in that slot switch to sleep mode. By this means, the wasted power resulting from non-ideal battery round-trip efficiencies is aimed to be minimized, since some of the harvested energy is wasted during the process of battery charging due to storage inefficiencies. This wasted power is %34 for NiMH batteries, which is found to be the most widely used battery in solar energy applications. If we allow the nodes to use the harvested energy without charging the battery, that power to be wasted can be used for transmission.

Also, simple yet useful energy prediction algorithms, which allow the nodes to predict their future energy profiles, have been developed. Nodes make predictions according to historical energy generation data and then send these predictions to CH, in order for them to be used in EHPBS.

In order to compare EHPBS with the scheduling algorithm proposed in [29], a computer simulation environment has been set. In this environment, nodes are assumed to harvest solar energy. From the results of the simulations that have been run for different topologies, it is observed that EHPBS increases network lifetime. It is shown that EHPBS performs better than the regular algorithm by 3.76% in the simulations with different types of batteries, by 2.38% in the simulations with different types of solar panels and by 2.25% in the simulations with different types of transceivers.

Testing EHPBS in a testbed had not been possible due to hardware limitations. That lead us to focus on optimization of advertising intervals for Energy Harvesting Bluetooth Low Energy, with the help of the research we did on duty cycling. We have inspected [24], which theoretically finds the optimum scheduling that ensures energy neutral operation. Then, we have reached an offline algorithm that uses Streched String Method. The algorithm enables maximum packet transmission by scheduling the duty cycle of the device in advertising mode and also maximizes the packet capturing probability of the device in scanning mode.

We have implemented this offline algorithm, using BLE113 Development Kit and BLE112 USB Dongle by Bluegiga. The software development environment for onboard 8051 microprocessor that is provided by Bluegiga has allowed us to upload the required software on BLE113 Development Kit. First, we have investigated the harvestable energy in the test environment, by using various numbers of solar panels to recharge 2 rechargeable AA batteries. Then, we have collected the required data by powering up the BLE113 Development Kit with our energy harvesting system. After that, we have tracked the number of packets captured by BLE112 USB Dongle and the voltage level in rechargeable batteries.

We have conducted the tests that involve solar panels once. The main reason for that is these tests take considerably long time. Also, energy harvested by the solar panel depends on environmental conditions. Especially in outdoor test, these conditions may change rapidly, which changes the behaviour of the system. If these tests are conducted in longer time with more realizations, the reliability of them would increase.

Then, we compared the implementation results of random advertisement schedule with the advertisement schedule offered by [24]. We have verified that the proposed algorithm maximizes the throughput while ensuring energy neutral operation.

The insight gained in this study is that the lifetime of energy harvesting sensor networks can be increased by adjusting the duty cycles of the nodes. Moreover, the harvested energy is sufficient for a BLE system to run.

6.1 Future Work

As the future work, the offline algorithm used for advertisement scheduling can be replaced by an online algorithm. In order to do that, Bluetooth Low Energy modules should be able to predict the future energy opportunities rather than using previously acquired energy profile predictions. Then, the work proposed in this thesis can be implemented on the modules using the energy predictions of the modules themselves. The BLE113 Development Kit's on-board microprocessor and the software development environment provided by Bluegiga have not made it possible for us to run an energy prediction algorithm on the module. An external microprocessor and a more advanced software program can be used for that purpose.

Another future work is to do research on Bluetooth Low Energy devices in scanning mode. In this thesis, we have focused on optimization of advertising intervals, and in return, we have worked on the devices in advertising mode. However, if we assume that the devices in scanning mode are also capable of harvesting energy, we will face another duty cycle optimization problem. Then, we can develop new algorithms to adjust scanning window and scanning interval values according to harvested energy.

REFERENCES

[1] Bharathidasan, Archana, and Vijay Anand Sai Ponduru. "Sensor networks: An overview." Department of Computer Science. University of California (2002).

[2] Chong, Chee-Yee, and Srikanta P. Kumar. "Sensor networks: evolution, opportunities, and challenges." Proceedings of the IEEE 91.8 (2003): 1247-1256.

[3] Nayyar, Anand. "Conceptual representation and survey of dynamic power management (DPM) in wireless sensor network." International Journal of Advanced Research in Computer Science and Software Engineering 3.3 (2013).

[4] Sudevalayam, Sujesha, and Purushottam Kulkarni. "Energy harvesting sensor nodes: Survey and implications." Communications Surveys & Tutorials, IEEE 13.3 (2011): 443-461.

[5] Younis, Ossama, and Sonia Fahmy. "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks." Mobile Computing, IEEE Transactions on 3.4 (2004): 366-379.

[6] K. Akkaya, M. Younis, A survey on routing protocols for wireless sensor networks, Elsevier Journal of Ad Hoc Networks 3 (3) (2005) 325–349.

[7] M. Younis, M. Youssef, K. Arisha, Energy-aware management in cluster-based sensor networks, Computer Networks 43 (5) (2003) 649–668.

[8] Y.T. Hou, Y. Shi, H.D. Sherali, On energy provisioning and relay node placement for wireless sensor networks, IEEE Transactions on Wireless Communications 4 (5) (2005) 2579–2590.

[9] Y. Xu, J. Heidemann, D. Estrin, Geography-informed energy conservation for ad hoc routing, in: Proceedings of the 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'01), Rome, Italy, July 2001.

[10] M. Adamou, I. Lee, I. Shin, An energy efficient real-time medium access control protocol for wireless ad-hoc networks, in: WIP Session of IEEE Real-time Systems Symposium (RTSS'01), London, UK, December 2001.

[11] T. Wu, S. Biswas, A self-reorganizing slot allocation protocol for multi-cluster sensor networks, in: Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN 2005), April 2005.

[12] Jolly, Gaurav, and Mohamed Younis. "An energy efficient, scalable and collision-free MAC layer protocol for wireless sensor networks." Wireless Communications and Mobile Computing 5.3 (2005): 285-304.

[13] K. Dasgupta, K. Kalpakis, and P. Namjoshi. An efficient clustering–based heuristic for data gathering and aggregation in sensor networks, in: Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC, 2003), New Orleans, LA, March 2003.

[14] Jiang, Xiaofan, Joseph Polastre, and David Culler. "Perpetual environmentally powered sensor networks." Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on. IEEE, 2005.

[15] Kansal, Aman, et al. "Power management in energy harvesting sensor networks." ACM Transactions on Embedded Computing Systems (TECS) 6.4 (2007): 32.

[16] Akgun, Berk, and Irmak Aykin. "EHPBS: energy harvesting prediction based scheduling in wireless sensor networks." Energy Aware Computing Systems and Applications (ICEAC), 2013 4th Annual International Conference on. IEEE, 2013.

[17] "Fast Facts". Bluetooth.com. Retrieved 10 December 2013.

[18] "Bluetooth traveler". hoovers.com. Retrieved 9 April 2010.

[19] Zanchi, Marta Gaia. "Bluetooth Low Energy." Litepoint, June 2012. Web. 12 May 2014. <http://www.litepoint.com/wp-content/uploads/2014/02/Bluetooth-Low-Energy_WhitePaper.pdf>.

[20] "Nickel Metal Hydride (NiMH) Article and Application Manual." . Energizer, Jan 2010. Web. 10 May 2014. <http://data.energizer.com/PDFs/nickelmetalhydride_appman.pdf>.

[21] Specification of the Bluetooth System, Covered Core Package, Version: 4.0; The Bluetooth Special Interest Group: Kirkland, WA, USA, 2010.

[22] Gomez, Carles, Joaquim Oller, and Josep Paradells. "Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology." Sensors 12.9 (2012): 11734-11753.

[23] Liu, Jia, Canfeng Chen, and Yan Ma. "Modeling Neighbor Discovery in Bluetooth Low Energy Networks." Communications Letters, IEEE 16.9 (2012): 1439-1441.

[24] B. Tan Bacinoglu, Fatih M. Ozcelik, Elif Uysal-Biyikoglu, "Bluetooth low energy cihazlarda duyuru cizelgelemesi", SIU 2014, 22. IEEE Sinyal Isleme, Iletisim ve Uygulamalari Kurultayi, Trabzon, Türkiye, 23-25 Nisan 2014.

[25] B. Tan Bacinoglu, Fatih M. Ozcelik, and Elif Uysal-Biyikoglu, "Advertisement scheduling in Bluetooth Low Energy devices", SIU 2014, 22nd Signal Processing and Communications Applications Conference, Trabzon, Turkey, Apr. 2014.

[26] Devillers, Bertrand, and D. Gunduz. "Energy harvesting communication system with battery constraint and leakage." GLOBECOM Workshops (GC Wkshps), 2011 IEEE. IEEE, 2011.

[27] M. A. Zafer and E. Modiano, "A calculus approach to energy-efficient data transmission with quality-of-service constraints," IEEE/ACM Trans. on Networking, vol. 17, no. 3, pp. 898–911, June 2009.

[28] J. Hsu, S. Zahedi, A. Kansal, M. Srivastava, and V. Raghunathan, "Adaptive Duty Cycling for Energy Harvesting Systems," in Proc. 2006 International Symposium on Low Power Electronics and Design. ACM, 2006, pp. 180–185.

[29] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Microsensor Networks,"IEEE Trans. Wireless Commun., vol. 1, no. 4, Oct. 2002, pp. 660–70.

[30] K. Kinoshita, T. Okazaki, H. Tode, and K. Murakami, "A Data Gathering Scheme for Environmental Energy-Based Wireless Sensor Networks," in 5th IEEE Consumer Communications and Networking Conference., Jan. 2008, pp. 719–723.

[31] C. Vigorito, D. Ganesan, and A. Barto, "Adaptive Control of Duty Cycling in Energy-Harvesting Wireless Sensor Networks," 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks., pp. 21–30, June 2007.

[32] 0. Younis and S. Fahmy, "Distributed Clustering in Ad Hoc Sensor Networks: A Hybrid, Energy-Efficient Approach," Proc. IEEE INFOCOM, Hong Kong, Mar. 2004; an extended version appeared in IEEE Trans. Mobile Comp., vol. 3, no. 4, Oct.–Dec. 2004, pp. 366–79.

[33] A. Kansal, et. al., "An Environmental Energy Harvesting Framework for Sensor Networks," ACM International Symposium on Low Power Electronics and Design, pp. 481–486, 2003.

[34] W. Ye, J. Heidemann, and D. Estrin, "An Energy-efficient MAC Protocol for Wireless Sensor Networks," in INFOCOM 2002. Proc. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies., vol. 3, 2002, pp. 1567–1576.

[35] A. Rowe, R. Mangharam, and R. Rajkumar, "RT-Link: A Global Time-Synchronized Link Protocol for Sensor Networks," Ad Hoc Networks, vol. 6, no. 8, pp. 1201–1220, 2008.

[36] T. van Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in SenSys '03: Proc. 1st International Conference on Embedded Networked Sensor Systems. New York, NY, USA: ACM, 2003, pp. 171–180.

[37] Y. Sun, O. Gurewitz, and D. B. Johnson, "RI-MAC: A Receiver-Initiated Asynchronous Duty Cycle MAC Protocol for Dynamic Traffic Loads in Wireless Sensor Networks," in SenSys '08: Proc. 6th ACM Conference on Embedded Network Sensor Systems. New York, NY, USA: ACM, 2008, pp. 1–14.

[38] J. Polastre, J. Hill, and D. Culler, "Versatile Low Power Media Access for Wireless Sensor Networks," in Proc. 2nd International Conference on Embedded Networked Sensor Systems. ACM, 2004, pp. 95–107.

[39] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: A Short Preamble MAC Protocol for Duty-cycled Wireless Sensor Networks," in Proc. 4th International Conference on Embedded Networked Sensor Systems. ACM, 2006, pp. 307–320.

[40] Siekkinen, M.; Hiienkari, M.; Nurminen, J.K.; Nieminen, J., "How low energy is bluetooth low energy? Comparative measurements with ZigBee/802.15.4," Wireless Communications and Networking Conference Workshops (WCNCW), 2012 IEEE , vol., no., pp.232,237, 1-1 April 2012

[41] Jia Liu; Canfeng Chen; YanMa, "Modeling Neighbor Discovery in Bluetooth Low Energy Networks," Communications Letters, IEEE , vol.16, no.9, pp.1439,1441, September 2012

[42] "Bluetooth Low Energy Technology." BLE113 Bluetooth Smart Module. Bluegiga, 19 Sept. 2013. Web. 20 May 2014. <https://www.bluegiga.com/en-US/products/bluetooth-4.0-modules/ble113-bluetooth–smart-module/documentation/>.

[43] Özkan, C. Implementation Of The Heed Clustering Protocol With Sleep Scheduling In Tinyos 2 On A Wireless Sensor Network Testbed. Thesis, Middle East Technical University. 2013.

[44] Baghaee, S., et al. "Demonstration of Energy-Neutral Operation on a WSN Testbed Using Vibration Energy Harvesting." European Wireless, May 2014.

[45] Rajesh, Ramachandran, Vinod Sharma, and Pramod Viswanath. "Information capacity of energy harvesting sensor nodes." Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on. IEEE, 2011.

[46] Ceran, E. T. Dynamic Allocation of Renewable Energy through a stochastic knapsack problem formulation for an access point on the move. Thesis, Middle East Technical University. 2014.

[47] Shakiba-Herfeh, M. Optimization of feedback in a multiuser MISO communication downlink with energy harvesting users. Thesis, Middle East Technical University. 2014.

[48] Uçtu, G. Optimal transmission scheduling for energy harvesting systems and implementation of energy efficient scheduling algorithms on software defined radio. Thesis, Middle East Technical University. 2014.

[49] Gül, Ö. M. A low-complexity, near-optimal scheduling policy for solving a restless multi-armed bandit problem occurring in a single hop wireless network. Thesis, Middle East Technical University. 2014.