EFFICIENT RESOURCE ALLOCATION IN ENERGY HARVESTING WIRELESS NETWORKS

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

BY

NEYRE TEKBIYIK ERSOY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

DECEMBER 2012

Approval of the thesis:

EFFICIENT RESOURCE ALLOCATION IN ENERGY HARVESTING WIRELESS NETWORKS

submitted by **NEYRE TEKBIYIK ERSOY** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy 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. İsmet Erkmen
Head of Department, **Electrical and Electronics Engineering** _____

Assoc. Prof. Dr. Elif Uysal-Bıyıkoğlu
Supervisor, **Electrical and Electronics Engineering Dept., METU** _____

Examining Committee Members:

Prof. Dr. Yalçın Tanık
Electrical and Electronics Engineering Dept., METU _____

Assoc. Prof. Dr. Elif Uysal-Bıyıkoğlu
Electrical and Electronics Engineering Dept., METU _____

Assoc. Prof. Dr. Cüneyt F. Bazlamaçcı
Electrical and Electronics Engineering Dept., METU _____

Assoc. Prof. Dr. Tolga Girici
Electrical and Electronics Engineering Dept., TOBB _____

Assoc. Prof. Dr. Çağatay Candan
Electrical and Electronics Engineering Dept., METU _____

Date: _____

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: NEYRE TEKBIYIK ERSOY

Signature :

ABSTRACT

EFFICIENT RESOURCE ALLOCATION IN ENERGY HARVESTING WIRELESS NETWORKS

Tekbıyık Ersoy, Neyre
Ph.D., Department of Electrical and Electronics Engineering
Supervisor : Assoc. Prof. Dr. Elif Uysal-Bıyıkoğlu

December 2012, 186 pages

This thesis presents various studies on energy efficient design of wireless networks. It starts with a survey on recent shortest path based energy efficient routing algorithms developed for ad hoc and sensor networks, making a comprehensive classification for these algorithms. In addition to energy efficient design, sustainable and environmentally friendly deployment of wireless networks demands increased use of renewable energy. However, this calls for novel design principles to efficiently utilize the variation in the availability of the energy. The thesis continues with an investigation of state-of-the-art resource management and scheduling algorithms developed for energy harvesting wireless sensor networks. Building on the state-of-the-art, the main contribution of this thesis is to formulate and solve a utility maximizing scheduling problem in a multiuser broadcast channel with an energy harvesting transmitter. The goal is to determine the optimal power and time allocations to users between energy arrivals. The structural properties of the problem are analyzed, and its biconvexity is proved. A Block Coordinate Descent (BCD) based algorithm is developed to obtain the optimal solution. Two simple and computationally scalable heuristics, PTF and ProNTO, which mimic the characteristics of the optimal policy, are proposed. Finally, an online algorithm, PTF-On,

that will bypass the need for offline knowledge about the energy harvesting statistics, is developed. PTF-On uses a Kalman filter based energy harvesting prediction algorithm, developed in this thesis, to predict the energy that will arrive in the future.

Keywords: Energy harvesting, optimization, block coordinate descent, biconvex, proportional fairness.

ÖZ

ENERJİ HARMANLAYAN KABLOSUZ AĞLARDA ETKİN KAYNAK PAYLAŞTIRIMI

Tekbıyık Ersoy, Neyre
Doktora, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi : Doç. Dr. Elif Uysal-Bıyıkoğlu

Aralık 2012, 186 sayfa

Bu tez, kablosuz ağların enerji verimli tasarımı üzerine yapılan çesitli çalışmalar sunmaktadır. Tez, kablosuz tasarsız ağlar ve algılayıcı ağlarda, ağ ömrünü uzatabilecek, yakın zamanda literatüre katılan, en kısa yol atama tabanlı enerji verimli yol atama algoritmalarının araştırılması, ve bu algoritmalara yönelik yeni bir gruplandırma yönteminin sunulması ile başlamaktadır. Enerji verimli tasarımın yanısıra, günümüzde birçok uygulamanın sürdürülebilir ve çevre dostu olması gerekmekte, ve, bu da yenilenebilir enerji sistemlerinin kullanımının artması ile mümkün olmaktadır. Fakat, yenilenebilir bir enerji kaynağının varlığı, enerji miktarlarındaki anlık değişimleri etkin bir şekilde değerlendirecek yeni tasarım prensiplerini gerektirmektedir. Bu dogrultuda, tezin devamında enerji harmanlayabilen kablosuz algılayıcı ağlarda kullanabilecek en yeni kaynak paylaştırma ve çizelgeleme algoritmalarının araştırılması konu edilmiştir. En son gelişmeler ışığında, bu tezin başlıca katkısı, çok kullanıcılı bir tümegönderim kanalında, enerji harmanlayabilme özelliğine sahip bir göndericinin bulunduğu durumda, fayda enbüyütme amaçlı bir çizelgeleme probleminin formüle edilmesi ve çözülmesidir. Amaç, enerji harmanları arasındaki zaman dilimleri için, optimal güç ve kullanıcılar arası zaman paylaştırımını belirlemektir. Problemin yapısal özellikleri incelen-

miş ve iki yüzü dışbükey (biconvex) olduğu kanıtlanmıştır. Optimal çözümü elde etmek için Blok Koordinat Alçalma (BCD) tabanlı bir algoritma geliştirilmiştir. Basit ve sayısal olarak ölçeklenebilen ve optimal çözümün karakteristik özelliklerine uyum sağlayan iki buluşsal yöntem, PTF ve ProNTO, tasarlanmıştır. Son olarak, online bir algoritma olan ve gelecekte harmanlanacak enerji miktarlarının önceden bilinmesini gerektirmeyen, PTF-On algoritması tasarlanmıştır. PTF-On, bu tezde geliştirilen bir Kalman filtre tabanlı enerji harman tahmin algoritması kullanarak ileride harmanlanacak olan enerji harman miktarlarını kestirebilmektedir.

Anahtar Kelimeler: Enerji harmanlama, eniyileştirme, blok koordinat alçalma, iki yüzü dışbükey, orantısal adil.

To my parents, my brother and my husband

ACKNOWLEDGMENTS

At last, I feel very glad to write this page, and, thank the many people who made this thesis possible. First of all, I wish to express my sincere gratitude to my supervisor Assoc. Prof. Elif Uysal-Bıyıkoğlu for her guidance, encouragement, and, belief in me throughout the progress of my thesis study. I feel privileged to work with such a mentor and role model. I would like to thank her for challenging me to think and work outside my comfort zone, telling me how many tasks she has to complete during the time I need to complete my own task, and, inspiring me to be the best version of me.

I would also like to thank my primary school teacher Mustafa Arpalıklı, who told my mother; “no matter which school she is in, as long as she keeps studying like this, she will achieve the success she wants” when we told him how upset we were as a family, because I had missed the special intermediate school entrance exam that I had studied so hard, since I was ill. I will keep remembering his words everytime I accomplish something...

I am very grateful to Prof. Dr. Kemal Leblebicioğlu and Assoc. Prof. Dr. Tolga Girici for their help and suggestions on my research. Without their help, this thesis would not have been possible.

I would also like to take this opportunity to thank to my thesis defense committee members, Prof. Dr. Yalçın Tanık, Assoc. Prof. Dr. Cüneyt F. Bazlamaçcı, and, Assoc. Prof. Dr. Çağatay Candan for their important remarks, suggestions and corrections which were useful and essential in the improvement of this thesis.

I would like to thank TÜBİTAK for supporting the work in this thesis through projects 106E119 and 110E252.

I wish to express my debt to all present and former colleagues in Telecommunications Lab, especially Tuğcan Aktaş, Gökhan Güvensen and Seçil Özdemir, who have provided me with inspiration, and advice, and who have so generously shared their knowledge and expertise with me. I would also like to thank my dear friend Ebru Gökalp, for always being there for

me. She has been the perfect friend to me, so supportive, helpful, and, considerate...

My deepest gratitudes go to my husband Ahmet Ersoy for his endless support, love and invaluable patience without which this thesis would not have been possible. I am grateful to him for standing by me through all the forcible experiences of this period, being so supportive even the times he had to be alone because I needed to study hard, and, being able to make me smile even with a headache caused by that study.

Finally, I am thankful to my family: My parents Kadriye and Hasan, and my brother Murat, for their love, support and belief in me throughout all these years. My parents have made lots of sacrifices for me to have a better future. They have always been proud of me for simply stepping up to take on a challenge. I hope I can make them even more proud in the future...

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

CHAPTER 1

INTRODUCTION

1.1 Motivation and Related Work

1.1.1 Sustainable Machine to Machine Networks

The progress in short range networking, growth of wireless mobile networks, and advances in device networking have allowed the development of a new technology, Machine to Machine communications (M2M) [10], [11], which has recently received considerable attention. M2M is mainly a combination of three common technologies: wireless sensors, the Internet and personal computers. In M2M, a field node or a group of field nodes[1] (in the vicinity of an event) gather data and send it wirelessly through a network (e.g. wireless sensor network), where it's routed, often over the Internet, to a server or cloud of servers. At that point, an application program (or a software agent), translates the data into meaningful information (e.g., an accident has occurred, items need to be restocked, etc.) based on information from the sensors and a set of rules, and then, sends commands to controllers or actuators, which issue the electrical signals necessary to make machines take action. Hence, M2M technology enables the next generation of wireless sensor networks and global connectivity to billions of processes, devices, and machines through the Internet. There is an immense potential for future applications of this technology. Some of the currently envisioned applications are building automation and structural control, transportation and logistics, healthcare, public safety and surveilance, and, environmental and utilities monitoring. We summarize some of the major existing and potential M2M applications in Figure 1.1. The vast majority of

[1] Field nodes can be sensors of properties such as temperature, humidity, flow measurement, and position finding system, as well as controllers and actuators for machines such as air conditioners, elevator pumps, traffic lights, etc.

M2M applications - perhaps 70 percent - have intrinsic environmental benefits that promote environmental sustainability [12].

Figure 1.1: Current and future M2M applications (Photo Credits: 1 [2], 2 [3], 3 [4], 4 [5], 5 [6], 6 [7])

With increasing awareness of the potential harmful effects to the environment caused by CO_2 emissions and the depletion of non-renewable energy sources, there is a growing consensus on the need to develop more energy-efficient systems and networks [13]. In recent years, researchers and industry analysts have pointed out that application of M2M can result in a "greener" ICT infrastructure. In particular, it is suggested in [14] that a combination of M2M and dematerialization services (such as eliminating paper bookkeeping in favor of electronic records) could reduce Europe's energy bill by at least 43 billion Euros and reduce CO_2 emissions by at least 113M tons per year in 2020 across 25 EU countries (EU-25).

However, the authors estimate that a billion mobile connections would be required to achieve these savings, 87% of which are Machine-to-Machine (M2M). As the use of M2M networks increases, the information generated by end users will also increase. The reliable distribution of this content will require increasing investments in infrastructure and maintenance, and a matching electricity bill to run the underlying ICT (Information and Communication Technologies). These could significantly offset the potential savings offered by M2M networks

unless innovative energy efficient technologies are employed in the design of these networks.

Today, ICT equipment and services are responsible for about 8% of all electrical power consumption in the EU, and about 2% of carbon emissions. As M2M applications and deployments continue to increase, the ICT sector will inevitably grow. This growth, together with increasing energy costs and the need to reduce greenhouse gas emissions make it important to design M2M networks as energy efficiently as possible on every stage from computation and storage to communication. Yet, despite the growing research interest in M2M, its energy efficient design per se has not been sufficiently addressed in the literature.

With this in mind, Chapter 2 is devoted to overview an array of state-of-the-art techniques and technologies that may be incorporated in various stages and layers of M2M design.

1.1.2 Energy Efficient Routing in Machine to Machine Networks

Although still in its nascent phase, M2M is a promising technology. M2M networks make use of multi-hop routing in order to route data in a wireless network. Since each node in a wireless multi-hop network acts as a router, one of the main issues in multi-hop routing is energy efficiency. In the last decade many energy efficient routing algorithms have been developed for use in multi-hop networks, such as ad hoc and sensor networks. Ad-hoc networks are dynamically formed multi-hop networks that can be deployed without the need for any fixed infrastructure, such as base stations. The nodes configure themselves into a network and cooperatively maintain network connectivity. Some works in which treat the problem of energy by reserving the network connectivity are [15], [1]. Sensor networks often have nearly an ad-hoc structure, except perhaps having a central data collection unit or sink.

The growing interest in sensor applications has created a need for protocols and algorithms for large-scale self-organizing ad-hoc networks, consisting of hundreds or thousands of nodes. Hence, in the past decade, wireless sensor networks (WSNs) have been the topic of considerable research effort due to their potential for civilian and military applications and their ability of being incorporated in M2M networks. Although M2M networks do not only consist of sensors, WSNs are a key component of machine-to-machine (M2M) communication. Therefore, sometimes sensor networks are referred to as M2M networks [16]. WSNs are made up of a large number of small sensors that are networked via low power wireless communications. A

sensor network enables cooperation, coordination, and collaboration among sensor nodes and thus it differentiates itself from a mere collection of sensors. The three main functions in a sensor network are sensing, controlling and actuating. These functions could be on separate nodes or co-located on the same physical node. Thus, there are three types of nodes; sensors, controllers and actuators [17]. These nodes are generally equipped with data processing and communication capabilities which are used for collecting and disseminating environmental data. Figure 1.2 shows a schematic diagram of sensor node components.

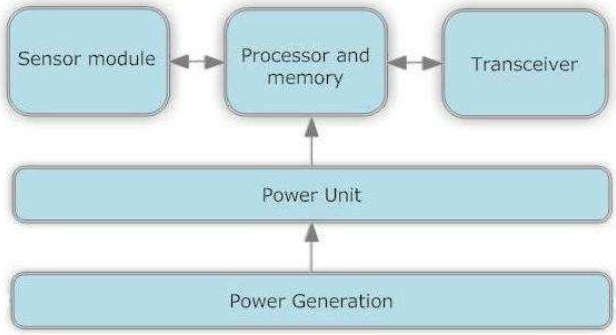

Figure 1.2: Basic components of a typical sensor node

The sensing circuitry measures ambient conditions related to the environment surrounding the sensor and transforms them into an electric signal. Processing such a signal reveals some properties about objects located and events happening in the vicinity of the sensor. The sensor sends such collected data, either to other sensors or back to an external base station (BS). A base station may be a fixed or mobile node (sink) capable of connecting the sensor network to an existing communications infrastructure or to the Internet where a user can have access to the reported data.

Due to the inherent multihop characteristic, routing is important in WSNs. Since a WSN consists of energy-constrained sensor nodes, the resources of the network (such as energy supply and bandwidth) should be used wisely. Therefore, at the network layer, it is highly desirable to use energy aware and energy conserving routing algorithms for routing and relaying of data from the sensor nodes to the BS. In developing energy aware and energy conserving routing techniques, WSNs are modeled as graphs. Then, a shortest path routing (SPR) algorithm is applied on this graph. Therefore, once an appropriate link metric has been defined, the optimal shortest path routes can be computed in polynomial time and in a distributed fashion [18].

There have been previous surveys on the characteristics, applications, and communication protocols in WSNs [19], [20]. The surveys in [19], [20] addressed several design issues and techniques for WSNs describing the physical constraints on sensor nodes, applications, architectural characteristics, and the protocols proposed in all layers of the network stack. In [21], a comprehensive list of recently proposed routing protocols is presented, and routing algorithms used in WSNs were classified as data-centric, hierarchical and location-based. As can be observed from [19] and [21] early literature on wireless networking addressed the design of efficient routing algorithms without optimization of the energy required to send the messages. A comprehensive survey of routing techniques proposed for wireless sensor networks is presented in [22]. The techniques mentioned in the survey have the common objective of trying to extend the lifetime of the sensor network while not compromising data delivery, as well as addressing routing challenges and design issues that may affect the performance of routing protocols in WSNs.

The motivation and scope of Chapter 3 differ from those of previous related surveys in that the focus is on link-cost based shortest path routing alternatives and mechanisms proposed for WSNs as well as other ad-hoc networks are considered.

1.1.3 Resource Management and Scheduling in Energy Harvesting Industrial WSNs

The collaborative nature of wireless sensor networks (WSNs) brings several advantages over traditional wired systems, including self-organization, rapid deployment, flexibility, and intelligent processing capability [23]. Recently, WSNs have found their way into a wide variety of applications and systems with varying requirements and characteristics [24], [25]: ocean water monitoring and bathymetry, avalanche rescue, object tracking, remote monitoring of oil and gas reservoirs, and, preventive and predictive maintenance (PdM)[2], which is considered to be an important example of the class of industrial WSN applications that provide measurable value in real deployments.

Resource management is just as critical in the industrial environment as in other deployment scenarios. This may seem counterintuitive since most industrial plants have ample power supplies and distribution systems. However, as also discussed by Krishnamurthy et. al. in [25],

[2] Predictive maintenance is a general term applied to a family of technologies used to monitor and assess the health status of a piece of equipment (e.g., a motor, chiller, or cooler) that are in service [25].

operating and safety regulations call for each piece of equipment to have a dedicated power circuit, thus requiring separate power connections for sensor nodes. Hence, to reduce installation costs, the WSN must either be battery powered (and employ aggressive resource management) or make use of energy harvesting. Since industrial WSNs are expected to be deployed in harsh or inaccessible environments for long periods of time, recently, employing energy harvesting (via ambient energy sources such as solar [26], vibrational [27], [28], wind [29] and thermal energy [30]) to replace/supplement batteries that power WSNs, has earned much interest. Detailed information about different types of energy harvesting approaches can be found in [30], which is a useful source that investigates the current energy harvesting WSN applications in several areas, and, provides examples of leading development enterprises.

The most popular source of the ambient energy is the sun. Solar energy is becoming widely used, due to its high power density compared to other sources of ambient energy [31]. Consequently, numerous researchers have designed energy harvesting circuits to efficiently convert and store solar energy [32], [33], [34], and, most of the studies mentioned in Chapter 4 focuses on solar energy harvesting. As claimed by Yu and Yue [35], solar energy harvesting is a comparatively fledged technology for WSNs used for outdoor applications. However, for indoor applications, it is not suitable since the efficiency of photovoltaic cell is very low under low indoor light luminous intensity. For indoor applications, one may prefer the micro-scale indoor light energy harvesting system developed in [35], or, the energy optimized sensor node [36] designed to harvest the energy from indoor light, for building climate control application.

Moser et. al. argue in [37] that, depending solely on energy harvesting gives rise to new challenges and will trigger the revision of conventional resource management. If, e.g., the size of a solar cell limits the available power/energy of an electronic device, decisions such as when to provide how much power, rate, service, etc. have to be made in order to satisfy the needs of the user as well as possible. Successful demonstration of perpetual operation with indoor EH WSNs have already been made, that are drawing attention to the importance of routing and scheduling mechanisms that are aware of the energy harvest process. For instance, in an indoor office environment, Hande et al. [38] used monocrystalline solar cells to scavenge energy from 34 W flourescent lightbulbs in order to supply (via supercapacitors) the routers of a WSN. The routers, operating in pairs, achieved virtually perpetual operation by resource-aware operation. The authors in [38] stressed that in scenarios with mobility, resource management mechanisms for other forms of energy scavenging (such as vibration-

based or thermal) should be investigated in future work.

Resource (energy) management in WSNs equipped with energy harvesting capabilities is substantially and qualitatively different from resource management in traditional (battery-powered) WSNs. As stated by Mao et. al. in [39], conservative energy expenditure in energy harvesting networks, may lead to (i) missed recharging opportunities because the battery buffer is full and, (ii) long delays because the energy is not being fully used to transmit at high enough data rates. On the other hand, aggressive usage of energy may result in reduced coverage or connectivity for certain time periods, not to mention complete battery discharges that could make the nodes temporarily incapable of transferring time-sensitive data. In industrial applications, this may lead to loss of production and may sometimes create hazardous situations [40]. Thus, new resource allocation and scheduling schemes need to be designed to balance these contradictory goals, in order to maximize the network performance. This is the main motivation for Chapter 4.

1.1.4 Proportional Fair Resource Allocation on an Energy Harvesting Downlink

Management of energy consumption is vital for the sustainability of many wireless communication systems. Therefore, especially in the past decade, energy efficient scheduling policies have been investigated [41, 42, 43]. Due to recent advances in energy harvesting technologies, emerging communication devices have been powered by rechargeable batteries which are capable of harvesting energy through solar cells, vibration absorption devices, thermoelectric generators, wind power, etc. Although energy harvesting allows sustainable and environmentally friendly deployment of wireless networks, it requires efficient utilization of time-varying energy. Hence, the focus should be shifted from minimizing energy expenditure to optimizing it over time.

It is well known (*e.g.*, [44], [45], [46]) that optimization of a broadcast channel (*e.g.*, the downlink) shared by many users calls for different choices of rate and power allocation to different users depending on the gains, channel conditions, demands of these users, and most importantly, the objective of the optimization.

There has been considerable recent research effort on optimizing data transmission with an energy harvesting transmitter. In [47], the authors develop a packet scheduling scheme that

minimizes the time by which the energy harvesting transmitter delivers all packets to the receiver of a single-user communication system. In [48], the authors extend this work to the multi-user case and, propose an iterative approach that reduces the two-user broadcast problem into a single-user problem as much as possible, and then, utilizes the single-user solution in [47]. [49] treats the time minimization problem for the two-user broadcast channel differently, as it proposes an iterative solution technique by considering two energy arrival slots at a time. These approaches are extended by [50] and [51] to the case of a transmitter with a finite capacity battery. [52] extends [47] one step further to propose the directional water-filling algorithm, which is able to find the optimal energy management schemes for energy harvesting systems operating in fading channels, with finite capacity rechargeable batteries. Both [51] and [52] investigate the following dual *offline* problems; maximizing the number of bits transmitted with a given deadline constraint, and minimizing the transmission completion time with a given number of bits to transmit.

Unlike the broadcast related studies mentioned above, [53] investigates the dual problems in a multiple access communication system. By using the generalized iterative backward waterfilling algorithm [53], the transmission completion time minimization problem can be simplified into convex optimization problems, and solved efficiently. [54] solves the short-term throughput maximization problem for a battery-limited energy harvesting transmitter in a single link topology.

In [55], the authors consider the problem of energy allocation over a finite horizon for point to-point wireless communications, taking into account a time varying channel and energy source, so as to maximize the throughput. In [56], Gatzianas et. al. consider an infinite-horizon *online* throughput maximization problem for a rechargeable sensor network. The authors propose a queue stabilizing transmission policy with decoupled admission control and energy allocation to maximize a function of the long term rate achieved per link. Chen et. al. [57] claim that infinite-horizon based solutions can be highly inefficient, especially in the context of networks with energy replenishment. Hence, unlike [56], [57] investigates the finite-horizon throughput maximization problem for a rechargeable sensor network.

The work in Chapters 5 and 6 differ from the previously mentioned studies particularly in their aim to maximize the throughput in a proportionally fair way, taking into account the inherent differences of channel quality among users. Due to characteristics of the utility function, the

problem presented is a *biconvex* problem[3] which is nonconvex, and has multiple optima. This allowes us to decompose the problem into two parts (power allocation, time allocation) and present a Block Coordinate Descent based optimization algorithm, BCD, that converges to a partial optimal solution. Although BCD is guaranteed to converge to a partial optimal solution and thus the partial optimal utility, it is computationally expensive and when there are tens of users and energy arrivals, forming invertible hessian matrices (needed for the optimization of the power variables) may be computationally excessive. Hence, we next restrict our general case assumption to the case where energy interarrival times are equal, in Chapter 6, so that we can analytically derive the characteristics of the optimal solution, and then, build on those to develop simple heuristics, PTF and ProNTO that closely track the performance of the BCD solution. Note that, not all generality is lost, since harvest amounts are arbitrary and the absence of a harvest in a certain slot can be expressed with a harvest of amount zero for the respective slot. Periodic sampling is consistent with practice as in many energy harvesting systems, transmitters have supercapacitors that can store the harvested energy and supply in every predetermined time window.

1.1.5 Prediction Based Proportional Fair Resource Allocation for Industrial Wireless Sensor Networks

As previously discussed in Chapter 4, resource management is just as critical in the industrial environment as in other deployment scenarios. In many industrial WSN applications, an area needs to be covered with a wireless sensor network (WSN), or multiple WSNs, monitoring different parameters, or different locations. Often, these subnetworks of simple devices send data via gateway nodes (or cluster heads) to a remote base station located at a central office, where the signal processing to produce strategic decisions runs on a more powerful computer. It is often also the central computer that maintains the health of the network by regularly recomputing the network topology. It is then necessary for the base station to broadcast certain network details and commands to the gateway nodes. Sustainable and environmentally friendly development of such industrial applications requires increased use of renewable energy, i.e., solar, wind, etc. Thus, in chapter 7, we address the case where the base station is supplied with solar energy harvesting. However, unlike what we have done in Chapters 5 and

[3] The problem of optimizing a biconvex function over a given (bi)convex or compact set, where a function $f : X \times Y \to \Re$ is called biconvex if $f(x, y)$ is convex in y for fixed $x \in X$ and is convex in x for fixed $y \in Y$ [58].

6, this time we focus on a more realistic scenario and thus, complicated problem; the *online* problem, in which the energy arrival amounts within a frame are not known apriori.

1.2 Contributions of This Thesis

The contributions of this thesis study can be stated as follows:

- An overview of the state-of-the-art techniques and technologies that may be incorporated in various stages and layers of sustainable M2M design:
  - Despite the growing research interest in M2M communications and the importance of energy efficiency in M2M networks, we are not aware of a detailed study on energy efficiency in M2M networks. By this means, we believe that the reserach presented in Chapter 2 will be a useful reference for those who are interested in energy efficient design of M2M networks.
- A detailed survey on link-cost based shortest path routing alternatives, and a comprehensive classification of the discussed algorithms, summarized in Table 3.2.
  - Several surveys on both ad hoc and sensor networks exist in the literature. However, the motivation and scope of the study presented in Chapter 3 differs from those of previous related surveys since the focus is on link-cost based shortest path routing alternatives and mechanisms. To our knowledge, this is the first study/survey devoted to the shortest path based energy efficient routing algorithms. It should be noted that, this survey is a published work [59].
- An overview of the state-of-the-art resource management and scheduling algorithms, developed for energy harvesting WSNs:
  - This study will soon be available as a book chapter [60].
- Proportional fair resource allocation related contributions:
  - The proportional fair resource allocation problem proposed for an energy harvesting downlink.
  - Derivation of the structural properties of the proposed problem.

  - An optimization algorithm that converges to an optimal solution, BCD.
  - Derivation of the optimal solution based characteristics.
  - Simple and efficient heuristics that closely track the performance of the optimal algorithm.

  - Although there has been a considerable recent research effort on optimizing data transmission with an energy harvesting transmitter, we are not aware of a study on an energy harvesting downlink that takes into account the inherent differences of channel quality among users to maximize the throughput in a proportionally fair way. The study in Chapter 5 entails not only the optimal power/rate allocation to users between energy arrivals, but also the optimal time allocation that will maximize a proportionally fair utility function. Chapter 6 provides simple, practical, and close-to-optimal algorithms that can operate in order of seconds, and by this means, it can be considered as a useful source for those who desire forming fast and proportionally fair allocations in an energy harvesting broadcast system. The studies in Chapters 5 and 6 are combined to form a journal paper which is accepted for publication [61]. Parts of these studies are presented in 27th International Symposium on Computer and Information Sciences (ISCIS 2012) [62], [63].

- A Kalman-filter based energy prediction algorithm, K-SEP, and an *online* proportional fair resource allocation algorithm, PTF-On, that can closely track the performance of the *offline* PTF algorithm.

1.3 Organization of The Thesis

In Chapter 1, the motivation of the thesis study is stated. The chapter also provides introductory information about the studies included to this thesis, along with their related works.

In Chapter 2, we overview an array of state-of-the-art techniques and technologies that may be incorporated in various stages and layers of M2M design. We start by describing a typical M2M architecture and its components, as well as issues related to their design. Next, we discuss possibilities for the energy conservation in the design of these components.

In Chapter 3, we focus on the problem of energy efficient routing and its significance in wireless networks, followed by a definition of link cost based Shortest Path Routing (SPR). Then, we describe various shortest path based energy efficient routing algorithms designed for wireless ad-hoc or sensor networks, and, the candidate link cost metrics that could be used in accordance with shortest path based algorithms.

Chapter 4 overviews a selection of state-of-the-art resource management and scheduling algorithms, developed for energy harvesting WSNs, selected in particular with respect to their suitability to the industrial WSN environment. The treatment includes an explanation of the operating principles, as well as the design settings for these algorithms. The drawbacks, advantages, and possible application areas of these algorithms are also discussed.

Chapters 5 and 6 are devoted to the proportional fair resource allocation problem on an energy harvesting downlink. In Chapter 5, we describe the system model, and, make the problem statement precise to study the mathematical structure of the proposed problem. The proposed BCD algorithm is described in this Chapter, followed by a detailed analysis and discussion of the nature of the solution found by BCD. We also test the insight gained from analysis about convergence and the nature of the solution, by running the algorithm on numerical examples.

Chapter 6 discusses the structure and properties of the optimal solution. Depending on these properties, PTF and ProNTO heuristics are proposed in this Chapter. The numerical and simulation results are also presented.

Chapter 7 presents our latest study, i.e., the *online* version of the problem proposed in Chapter 5. The chapter explains how we leveraged the PTF heuristic, to propose a stand-alone algorithm, PTF-On, that can predict the base station's energy arrival profile throughout the day, and then, act upon this energy arrival profile to maximize the throughput in a proportionally fair way.

Finally in Chapter 8, the summary of this study, and, a discussion about the possible future directions is presented.

CHAPTER 2

TOWARDS SUSTAINABLE MACHINE TO MACHINE NETWORKS

M2M communications (M2M) is a rapidly growing technology that automates the communication among heterogeneous groups of devices. M2M has received increased attention recently due to its potential for reducing the energy consumption and greenhouse gas emissions caused by current communication technologies and human activities. In order not to offset the potential savings by the overall energy consumption of computation, storage and communications needed to realize M2M networks, it is necessary to design these networks with utmost energy efficiency at every layer of their architecture.

This chapter starts by laying out the components of an M2M architecture, and continues by a wide scoped consideration of solutions from recent literature that can be employed toward energy efficient design of each of these components. The array of solutions surveyed include, energy efficient hardware and protocols for wireless sensor and actuator networks and, performance-improving techniques for other wired and wireless networks that serve to support M2M networks.

2.1 M2M System Architecture and Design Issues

2.1.1 M2M Architecture

M2M is a combination of various heterogeneous electronic, communication and software technologies. A typical M2M system, illustrated in Figure 2.1, comprises of the following basic components [64], [65]:

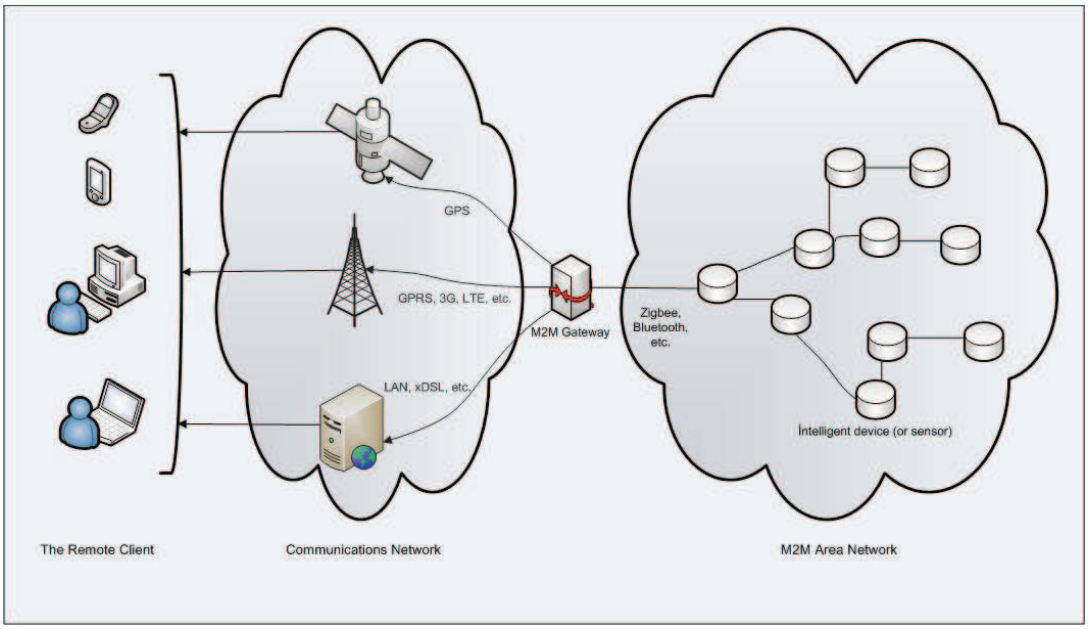

Figure 2.1: M2M architecture

- Intelligent and communication enabled modules and devices: These devices include sensors, actuators, RFID tags, PLCs (Programmable Logic Control), I/O modules and any other device, machine or appliance that incorporates a communications module. The wireless communications module can be built into the system or integrated as an add-on component. These devices are programmed to read, and sometimes react to, actions and conditions such as motion, pressure, or temperature. We will refer these devices as intelligent devices.

- M2M Area Network: This network provides connectivity between intelligent devices and M2M Gateways. Examples of M2M Area Networks include: Local networks such as M-BUS and Wireless M-BUS, and, Personal Area Network (PAN) technologies such as IEEE 802.15, SRD (Short Range Device), UWB, Bluetooth, Zigbee, sensor networks, etc.

- M2M Gateway: An M2M gateway is responsible for extracting raw data from an intelligent device and preparing it for the network. The gateway uses a proprietary protocol or device driver to interact with the intelligent device, and translate the data into a format that another device, application or human can understand. Mainly, an M2M gateway facilitates communication among the various devices and provides a connection to a backhaul that reaches the Internet. The M2M gateway can have many different embod-

iments [66]; it needs to support one or more of the local network protocols as well as the backhaul connection to the Internet. The backhaul connection may be Ethernet, cable, DSL, fiber, or cellular. We refer the interested reader to [66] for information about the issues related to the design of M2M gateways. Although the focus is on home M2M networks in [66], the converged M2M gateway structure proposed by the authors is a single product that uses advanced protocols to control all of the home network devices and thus, it provides insight into how an M2M gateway should be designed.

There can be several gateways in an M2M network. Every gateway is capable of analyzing a certain amount of data transmitted from various types of intelligent devices. According to Kim and Choi [67], when an M2M application queries the sensing information of a certain geographical region to an M2M platform, the platform should find out the devices that cover the requested region and select the M2M gateways to route the request to those devices. Since the M2M platform knows geographical locations of the registered devices and M2M gateways only, the M2M platform should speculate about the coverage of M2M area network (M2M-AN) which consists of the M2M gateway[1] and associated devices including both registered and unregistered devices. In this case, determining the smallest feasible M2M gateway set (list) whose collective coverage spans the target region is important for minimizing the routing overhead, transmission power and processing resources. We refer the interested reader to [67] for details of four algorithms proposed by Kim and Choi in order to select the M2M gateway list for geographical region based query from certain M2M applications.

- Communications network: The Communications network in an M2M application is the central connection component between an intelligent device and a remote client. It provides communications between the M2M Gateways and the remote clients (or software applications). Examples of Communications networks include: xDSL, IEEE 802.11, Local Area Networks (LAN), GERAN (GSM EDGE Radio Access Network), UTRAN (UMTS Terrestrial Radio Access Network), W-LAN, WiMAX (Worldwide Interoperability for Microwave Access), and the cellular communication technologies like GSM (Global System for Mobile communications), GPRS (General Packet Radio Service), EDGE (Enhanced Data rates for GSM Evolution), 3G, LTE (Long Term Evolution), CDMA, etc. A large number of M2M applications transmit relatively low data

[1] In some M2M applications, M2M gateways can be included to the M2M-AN structure.

volumes, so the potential transmission speed has little significance for them. Among many choices mentioned above, the pricing model of GPRS and EDGE makes them attractive for M2M applications. Some gateways also support GPS (Global Positioning System) technology in order to transmit location data by using satellite communication. Communications network allows the system to send information to a back-end server, which processes the data and sends it via the Internet to the facility that monitors and controls the machine, via protocols such as TCP/IP and the short message Peer to Peer Protocol.

- The remote client or application: This is the destination of the information. The remote client can be either a hardware or a software that receives the data. More specifically, clients can be cell phones, web browsers (e.g., Internet Explorer, Firefox, or Google Chrome, etc.), email clients and smart messaging (SMS) devices, among others. These clients use a software application which allows the received data to be analyzed, reported, and acted upon.

2.1.2 M2M Design Strategies

There are basically three infrastructure design strategies in M2M implementations: push strategy, pull strategy and push/pull (hybrid) strategy. In push strategy, the intelligent device initiates the communications and sends data through an M2M gateway over the network to a remote client. In order to perform this operation, the device recognizes pre-defined conditions and triggers itself to send alarms, alerts, e-mails, data and commands to an M2M gateway. Then, the gateway connects over the network on an as-needed basis to send data to the remote client. For instance, if the temperature level in a room rises above a predetermined threshold, the intelligent device sends a command via the M2M gateway to a PLC or I/O unit to enable power to a cooling device. Depending on the device's capabilities, it may also be possible to send an email to a technician's cell phone. In this scenario, the gateway connects over the network only when the temperature threshold is exceeded. Pull strategy, on the other hand, requires a server and an always-on LAN, or Internet connection to ensure continuous reporting. Hence, it is more expensive than the push strategy. In the Pull strategy, there is a server on the network which is tasked with polling (periodically) the intelligent devices for data. The server is capable of broadcasting the data throughout the enterprise, sending alerts, alarms,

messages, or commands to other devices. An advantage of this strategy is that it enables the user to check whether the server is consistently establishing communications with the intelligent device or not. The hybrid strategy is a combination of the push and pull strategies. The hybrid method is used when the user demands to access the intelligent device at any time and rely on the the device to take action when an event is triggered. However, if the M2M gateway has an Internet connection and, acts as a Web server, the hybrid method introduces many issues [64] such as the need for a static IP address for each intelligent device which necessitates greater security precautions such as individual firewalls.

Having laid out the typical architectural components, we are ready to discuss the construction of these for sustainable operation of the whole M2M network.

2.2 Requirements for M2M Network Design

Despite the promising real-time monitoring applications and tremendous benefits, M2M communications is still in its infancy and faces many challenges, namely, meeting the following requirements [68], [69] are met.

1. Scalability: Scenarios with hundreds or thousands of intelligent devices are easily foreseable, considering, for example, the smart grid. Network protocols that scale well are thus needed.

2. Sustainability / Energy Efficiency: To reduce the need for human intervention for true automated operation, sustainable operation for months or years at a time is desirable. Usually, a mass of intelligent devices are deployed in M2M networks. Hence, as supported by Lu et. al. [70], M2M communications should focus on energy efficiency by optimizing M2M nodes' sensing, processing, and transmissions, and ultimately prolong the lifetime of the whole M2M communications. Protocol architecture that support sustainable, low power operation with adaptive duty cycling, sleep/wakeup scheduling, etc. are thus needed.

3. Reliability: Low reliablity in sensing, processing, and transmission, leading to high error rate in monitoring, data loss or long delays will certainly reduce the applicability of M2M infrastructures. Hence, M2M infrastructures need to be designed to ensure certain reliability requirements at every layer.

4. Mobility: In certain applications where intelligent devices are built on moving machinery, for example, or are intrinsicly mobile or portable, it will be required that the network provide seamless mobility.

5. Priority / Identification: Management of services with different priority classes is likely to be a requirement. Interrupting routine traffic to transmit important or urgent messages or serve a flow with higher priority is expected in many automation scenarios. To accomplish this, the network needs to support identification of intelligent devices or groups of intelligent devices by permanent or temporary IDs, location or combination thereof (e.g. URIs or IMSI).

6. Heterogeneity: Often, the M2M System should be able to support a variety of different device types, e.g. active intelligent devices and sleeping intelligent devices, upgradable / not upgradable intelligent devices, and, M2M gateways. Hence, the M2M network should be capable of interfacing heterogeneous M2M Area Networks.

7. Security/Privacy: M2M network should be designed in a way that prevents unauthorized use of the intelligent devices and the M2M Gateway. It should be capable of protecting privacy.

Recently, a considerable attention has been paid to the deployment of architecture and software challenges in M2M networks not only from the IT industry but also from academia [70]. However, the energy efficiency related issues in M2M networks have not been well explored. On the other hand, energy efficient design can increase the applicability of M2M and overall impact to global energy efficiency [70]. In the next section we focus on methods and technologies that can be used for designing sustainable and energy efficient networks.

2.3 Sustainable M2M Networks

This section will make an overview of energy efficient design at various components of an M2M network.

2.3.1 Energy Efficient Solutions for Intelligent Devices

Intelligent devices form the core of an M2M network, and naturally the availability of the devices themselves is critical for the reliability of the network. Intelligent devices used by M2M applications are often wireless sensor nodes or communication modules that incorporate sensors. These are typically powered by batteries of finite capacity which need to be replaced after a certain period of use. However, for many current M2M applications such as Environmental Monitoring and Structural Health Monitoring (SHM), battery replacement is impractical - nodes may be crammed into hard-to-reach nooks or distributed over wide areas. Hence, in order for M2M networks to operate successfully and efficiently, M2M applications must overcome the maintenance problem, probably through using nodes that can automatically replenish their energy from the environment. This concept, sometimes called ambient energy harvesting, has attracted great interest from the research community in recent years.

Today, the main sources of ambient energy considered suitable for use in wireless sensor networks are solar, mechanical (vibration or strain) and thermal energy [71]. Solar power is one of the most common and mature forms of energy harvesting, succesfully used in many current implementations. One of the most poignant examples is a plug-and-play solar energy harvesting module (Heliomote3 [72]) which is capable of powering commonly used sensor nodes, including Crossbow's Mica2 and MicaZ, Moteiv's Telos, Yale's XYZ, Intel's Stargate, and ISI's PASTA.

Harvesting of vibrational, kinetic and mechanical energy generated by movements of objects is particularly suitable around roads, bridges, and rail tracks. One method of harvesting vibrational energy is through the use of a piezoelectric capacitor while kinetic energy can be harvested using a spring-loaded mechanism. For example, traffic sensors can be powered by the short duration vibrations generated by a vehicle driving over them. Recently, Torah et al. developed an autonomous and energy aware wireless condition monitoring sensor system (ACMS) powered by harvested vibrational energy [73]. This system employs a kinetic energy generator (EG) which converts mechanical energy, in the form of vibrations, into electrical energy using an electromagnetic transduction mechanism. Despite the small volume of the EG ($150mm^3$), it produces sufficient power for the entire microcontroller subsystem from a very low vibration source of $0.2ms^{-2}$ with no additional battery or power supply.

Table 2.1: A comparison of possible wireless technologies for M2M-ANs

		Zigbee	Bluetooth	Wi-Fi	UWB (Ultra Wide Band)
Supporting Organization		ZigBee Alliance	Bluetooth SIG	Wi-Fi Alliance	UWB Forum and WiMedia Alliance
RF Frequency		868 MHz (Europe) 915 MHz (USA & Australia) 2.4 GHz (Worldwide)	2.4 GHz	2.4 & 5 GHz	3.1-10.6 GHz
Data rate		Up to 250 Kbits/sec	1 Mbits/sec	11 & 54 Mbits/sec	100-500 Mbits/sec
Topology		Mesh, Star, Tree	Star	Star	Star
Range		70-300 meters	10 meters	50-100 meters	4-20 meters
Max. number of nodes		65536	8	32	128
Power Consumption	Transmission Power	≈ 70-80 mW	≈ 100-110 mW	≈ 770-780 mW	≈ 780-790 mW
	Reception Power	≈ 80-90 mW	≈ 100-110 mW	≈ 760-770 mW	≈ 770-780 mW
Typical Applications		Industrial control & monitoring, WSNs, building control & automation, etc.	Wireless connectivity between devices such as phones, PDAs, laptops, etc.	Wireless LAN connectivity, broadband Internet access	Streaming video, home entertainment applications
Battery Lifetime		Months to years	Days to weeks	Hours	Hours to days

Thermal energy harvesting, on the other hand, exploits the property of certain semiconductors to generate electricity from temperature differences or gradients (e.g. between the human body and the surrounding environment). The practical challenge here is to harness sufficient energy from small gradients occurring in daily life, such as between a person's skin and the surrounding air. Chandrakasan and Ramadass [74] have demonstrated devices which can exploit differences of just one or two degrees Celcius to produce tiny (about 100 microwatts) yet usable amounts of electric power. A system powered with such thermal energy harvesting devices could, for example, enable 24-hours-a-day monitoring of heart rate, blood sugar or other biomedical data, through a device worn on a patient's arm.

2.3.2 Energy Efficiency in M2M Area Networks

2.3.2.1 MAC Standards for M2M-ANs

M2M-ANs can employ various MAC standards (wireless networking technologies) such as UWB, Bluetooth, WiFi, etc. The chosen technology can dramatically impact the rate, range, capabilities and the cost of the network. Table 2.1 presents a comparison of some of the popular wireless networking technologies appropriate for M2M networks.

UWB (IEEE 802.15.3a) uses an extremely wide band of RF spectrum to transmit data, and thus, it can transmit more data in a given period of time than the more traditional technologies (Bluetooth, WiFi, Zigbee, etc.). The shortcoming of this technology lies in its high power requirements. Also, the IEEE 802.15.3a task group was dissolved in 2006, making further support for UWB unattainable in the future, if UWB is selected as the communication technology in an M2M-AN. WiFi (IEEE 802.11) also, is suited for high-data-rate applications over large areas. As discussed in [75], this technology enjoys an enormous infrastructure for residences and support for IPv6 addressing. The main shortcoming of WiFi is similar to that of UWB: the high power requirement. Hence, WiFi is considered impractical for M2M communications. Bluetooth (IEEE 802.15.1) technology supports IP addressing and is well suited for low-power/low-data-rate applications. However, Bluetooth networks (known as *piconets*) support up to only eight devices communicating simultaneously. To provide scalability, M2M-AN requires a number of piconets (each consisting of eight devices), being able to communicate with one another via their master devices. This, however, leads to increased communication latency. Another drawback of Bluetooth is that it requires periodical waking up ($\approx$ 3 secs) and synchronization with the master device of the piconet.

When energy efficient operation and longevity (rather than high rate, sophisticated features, or a long communication range) are the main concerns, the Zigbee standard stands out among currently available technologies. ZigBee is an open global standard that operates on the IEEE 802.15.4 physical radio specification, in unlicensed bands including 2.4 GHz, 900 MHz and 868 MHz. ZigBee was developed particularly for wireless devices, ensuring low power and long life time. It supports power-saving modes, battery-optimized network parameters, and application configurations to address the needs of low-cost, low-power multi-hop wireless networks. We refer the interested reader to [76] for details about how to plan and develop Zigbee networks.

2.3.2.2 Protocols and Algorithms for M2M-ANs

Today, the most widely used M2M area network (M2M-AN) paradigm is a WSN. The lifetime of a WSN, which is used as an M2M-AN, directly affects the performance of the whole M2M network. Thus, the problem of efficient utilization of a finite energy resource on nodes is of great importance. Over the last decade, various solutions have been proposed for this problem

in the research community. A large part of this effort targeted energy conserving protocols at the network and link layers. A comprehensive survey of promising new solutions for WSNs, such as techniques for energy efficient data acquisition, can be found in [77]. We refer the interested reader to [78] for a recent survey of MAC and routing protocols that have been proposed for WSNs with the goal of prolonging the network lifetime. In the MAC domain, they list protocols that conserve energy by reducing collisions, idle-listening, overhearing and excessive overhead. In the routing domain, the authors discuss two major categories of routing protocols, namely data-centric routing[2] and hierarchical routing[3]. Due to the inherent multihop characteristic, energy efficient (or energy aware) routing is very important in WSNs. Among energy aware routing techniques, shortest path based approaches focus on modeling WSNs as graphs. The main aim in doing this is to apply a shortest path based routing algorithm on the network graph to determine the most energy efficient route. In Chapter 3 and [59], we survey shortest path based energy efficient routing alternatives that can be used in M2M networks. A novel comprehensive classification of these algorithms is also available in Chapter 3 and [59].

In the previous section, we discussed energy harvesting methods that can be used to power the sensors used in WSNs. Energy management in wireless networks equipped with energy harvesting capabilities is substantially different from energy management in traditional (battery-powered) networks [39]. Thus, new resource allocation schemes should be designed to balance these contradictory goals, in order to maximize the network performance. In Chapter 4, we focus on resource allocation schemes that could be used for sustainable operation of Industrial WSNs. These shemes can easily be used in M2M-ANs, as Industrial WSNs are often used as M2M-ANs.

Another typical M2M-AN is the Wireless Sensor and Actuator Network (WSAN), a heterogeneous network that consists of sensor and actuator nodes. As the actuating task is typically a more complicated and energy-consuming activity than the sensing task, actuators are resource-rich (i.e, line-powered) nodes equipped with better processing capabilities, higher transmission powers, and longer battery life. There are two basic architectures for data processing in WSANs: Automated architecture (AA) and Semiautomated architecture (SA). In

[2] Protocols in this category employ data-aggregation and sometimes caching to reduce the number of transmissions and amount of data to sink.

[3] Protocols in this category divide the network into clusters where clusterheads gather data from field sensors, aggregate and transmit the data.

AA, sensor nodes sense the environment and report the data to actuator nodes which then initiate appropriate actions based on the received data. In SA, sensor nodes route sensed data to a sink which may then issue action commands to actuator nodes. As transmitting the sensed data to the sink typically results in fast energy depletion of nodes around the sink, this type of architecture tends to have a disadvantage in terms of network lifetime. In AA, however, sensed data is reported to actuators and different actuators may be triggered based on different events, resulting in a more evenly distributed communication load and thus more even draining of energy.

2.3.3 Energy Efficiency in Wide Area Communication Networks

2.3.3.1 Energy Efficient Solutions for Wireless Connectivity

Considering the part of an M2M architecture from the gateway to the end user, the power consumed on the access network is often the dominant component of the overall power consumption of the M2M architecture. Indeed, [79] reports that the energy consumption of wireless access networks account for more than 55% of the whole communication sector. Today the most profitable wireless access network choices for M2M networks include radio access networks (RANs) and cellular networks. According to [13], as of 2008, the energy consumption of the cellular communication systems corresponded to 60 billion kWh of electricity usage annually, and, about 40 million metric tons of CO_2 emissions each year which is equivalent to annual greenhouse gas emissions from about 8 million cars. Hence, in this section, we will mainly focus on energy consumption of RANs and cellular networks. Today, the most attractive alternatives for reducing the energy consumption of RANs and cellular networks are: varying the cell size, switching-off some BSs, and, sometimes, switching between operators [80], [81]. Therefore, the energy efficient solutions mentioned in this section will be related to the above-mentioned three alternatives.

a) Cell Size: Large vs. Small: Cell sizes are categorized into macro-, micro-, pico-, and femto-cells[4]. A small-sized cell based approach requires many base stations (BSs) with low transmit power level. A large cell based approach, on the other hand, requires a few BSs with a high transmit power level. This short section is devoted to the effect of

[4] Femtocells enable connecting miniaturized, lower power base stations to wired backhauls such as home digital subscriber lines or cable modems [82].

cell sizes on the energy consumed by the network, and, the selection of the appropriate cell size for M2M networks.

As cell size is reduced, energy efficiency and system capacity can significantly increase. According to Leem et. al. [83], if the path-loss exponent is taken as 4 and the per-energy capacity of a macro-cell is normalized to 1, the per-energy capacities of a micro-, pico-, and femto-cell are 16, 10^4, and 10^8 in downlink and 64, 10^6, and 10^{12} in uplink, respectively. Therefore, for the setting in [83], even though many BSs are required in small-cell based approaches, using small cells leads to significant reduction in energy consumption. Moreover, the amount of CO_2 emission of one BS transmitter is approximately 181 kg in a year, if the cell radius is 1 km, and, if the cell radius is reduced to 500 m, 100 m, and 10 m, the amount of CO_2 emission in a year reduces to 45.25 kg, 1.81 kg, and 18.1 g, respectively [83]. Hence, it is evident that small-cell based communication systems not only reduce energy consumption but also help protecting the environment. Bhaumik et. al., however, claim in [84] that the optimal cell size from an energy perspective depends on a number of factors, including base station technology, data rates, and traffic demands (explained in [84]). This claim is relevant to M2M networks as the technology, traffic demands, and, data rate requirements change from application to application. It is the service requirements of each M2M application that leads to the definition of the media (and thus the cell size) used for transmission of M2M data. M2M applications have traditionally only required relatively low data rate connections, for which cellular 2G technologies, such as GPRS, EDGE and CDMA have been perfectly adequate. In 2006, however, high rate wireless M2M applications, such as, remote information display, and in-vehicle camera systems, began to be deployed. While high speed wireless M2M will likely only represent a small proportion of total M2M connections over the course of the foreseeable future, these applications will continue to exist, increasing diversity in M2M network deployments.

The appropriate cell-size for an M2M network depends on the type of the application it is used for. An M2M network with macro cells is effective in providing area coverage for voice and low-speed data traffic, but limited in providing high data rates per unit area. Hence, larger cells are needed for energy saving when data rate is low. If an M2M application require high data rates, however, small-cell based approaches are more appropriate since they can be very effective in accommodating high data rates with low

energy consumption. For some hybrid applications, a hierarchical cell structure, consisting of cells with different sizes and ranging from macro to micro cells, can also be used. We refer the interested to [85] for an energy-aware hierarchical cell configuration framework that has theoretical results as well as practical guidelines on how wireless network operators should manage their BSs.

When traffic demands, rather than the data rate, is the primary concern for an application, one should prefer approaches/architectures that consider traffic demands in choosing cell sizes. A multi-layer cellular architecture that automatically detects traffic demands at different locations and selects the best cell size from a few alternatives has been proposed in [84]. It was shown that the simple 2-layer architecture developed in [84] is compatible with current cellular standards (GSM, UMTS, WiMAX), and, can save up to 40% power compared to a traditional static cellular network. Instead of choosing the cell size among a few alternatives, one may prefer to adapt the cell size to traffic load. In such a case, the *cell zooming* approach [86] which adaptively adjusts the cell size according to traffic load, user requirements and channel conditions seems to be the best choice. Cell zooming can not only solve the problem of traffic imbalance[5], but also reduce the energy consumption in cellular networks. Instead of providing further details about the cell zooming approach, we refer the interested reader to [86] in which the implementation issues of cell zooming, including techniques, benefits, and challenges, are discussed in detail.

b) BSs: Switch-off or Sleep: In a recent document [87], it is reported that the Radio Base Station (RBS)[6] energy consumption is the dominant part of total energy consumption of a wireless access network. Oh et.al. agree with [87], reporting that the base stations contribute to 60% - 80% of the celluar networks' total energy consumption. Thus, the introduction of sleep modes and switching-off mechanisms in the operations of base stations are today considered as the most promising approaches to reduce the energy consumption of cellular access networks. Sleep modes are particularly effective at the network periphery, where the degree of traffic aggregation is low, and where the network is less vulnerable to possible failures, service degradation, since the number of affected

[5] For a cellular network in a city, the traffic load in the daytime is relatively heavy in office areas and light in residential areas, while the opposite things happen in the evening. For a static cell deployment, this causes some cells to be under light load, while others are under heavy load.

[6] RBS is also referred to as the base transceiver station (BTS), node B (in 3G Networks) or, simply, the base station (BS).

users is limited [80]. The sleep mode mechanisms proposed by [88], enable savings in energy (for 2G and HSPA (High Speed Packet Access) systems): the dynamic scheme switches resources ON and OFF as a function of the instantaneous change of the load in the system, which in turn follows the users arrivals and departures. The semi-static one, on the other hand, allows the resources to remain active for a relatively longer time interval, in order to minimize the number of activation/deactivation commands. It was shown in [88] that the dynamic approach is mostly efficient when the load in the network is high, and, the semi-static approach is as good when the traffic load is low. Thus, semi-static one is preferred for low traffic since it requires fewer activation and deactivation procedures. Another sleep mode based approach, which can be useful in M2M networks, has been presented in [89]. By introducing a feature that dynamically puts low loaded cells into sleep mode, during which BS's radio equipment is effectively powered down, the authors report a daily energy saving of 33% over a period of 12 hours. At this point it should be noted that, this is achieved without any degradation to user satisfaction and thus network performance [89]. However, this comes with a reduction in average user data rate. Until now, we discussed the benefits of using sleep mode in which the BS experiences a litle or no activity. However, in order to achieve significant energy savings, what is called for is a more carefully designed approach that will allow the system to shut entire BSs and transfer the corresponding load to neighboring cells during periods of low-utilization [13]. [90] is one of the most well-known studies that deal with this problem. [90] studies the BS switching strategy using a simplified analysis and shows simulation results for several switching-off BS ratios. However, [90] does not analyze the dominant factors for minimizing the energy consumption based on the BS switching, the traffic profile and the BS density. The traffic profile during day time is higher than that during night time. Moreover, there is a difference in the traffic profile observed on a normal weekday and on weekends. However, BSs are planned to support the day time traffic. Therefore, infrastructures of access networks are under utilized during the night time and the holiday period. This necessitates the use of effective dynamic BS switching strategies. A basic dynamic BS switching strategy which considers the time varying characteristic of the traffic profile has been proposed in [91]. The first-order analysis performed in [91] reveal that the amount of energy saving is dependent upon the traffic ratio of mean and variance and the BS density. The results of [91] provide a guideline on how to manage the BS resources so as to obtain energy

saving. Another dynamic BS operation approach has been proposed in [92]. Litjens and Jorguseski [92] apply a well-structured approach for the derivation of the potential energy savings that can be achieved by switching off UMTS/HSDPA sites in off-peak hours, while maintaining a prespecified requirement on the average or cell edge performance. The authors report possible energy savings of up to about 40% depending on the network operator's performance target and the specifics of the energy consumption model. However, the results presented in [92] are valid only for UMTS/HSDPA networks and thus, can not be generalized. A generalized first-order approximation of the percentage of power saving one can expect by turning off BSs during low traffic periods has been derived in [13]. The most important contribution of [13], however, is not the approximation itself, but the discussion of related challenges. We refer the interested reader to [13] for a useful discussion on challenges and potential solutions in switching approaches, including maintaining coverage, enabling cooperation between operators, and providing E911 service.

Recently, telecommunication operators, alongside researchers, have started paying attention to energy issue, and begun to study green solutions. From the perspective of the operators, reducing energy consumption is not only a matter of being green and responsible, it is also very much a cost issue. Many service providers and cellular network operators have been exploring ways to increase energy efficiency in all components of access networks. The following is a summary of recent technologies, developed by different service providers and operators:

- Ericsson: Ericsson has developed a number of energy optimization innovations [93] that reduce the total cost of ownership, while at the same time improving the environmental performance of mobile network growth worldwide. One of these innovations, Base Transceiver Station (BTS) Power Savings feature, works by putting the radio resources of the network that are not being used into standby mode during periods of low traffic. It is reported that, depending on network traffic patterns, the feature can reduce energy consumption by up to 25 percent in the radio access network. According to the company, if all of its currently installed GSM base stations had this feature, CO_2 emissions would be reduced by one million tons per year.

- Motorola: Motorola has unveiled a new design[7] for its WiMAX base stations (BSs)

[7] The new design was showcased at the Global Green Telecom in November 2009 in Dubai.

that costs less to build and operate. According to Motorola, build costs for these new designs are 35% lower than traditional BSs and power consumption is reduced by 60%. Moreover, Motorola estimates that maintenance and power costs for a hybrid site (a combination of solar and power generators) over a five-year period are 157, 774 USD, which compares favorably with the maintenance cost of 228, 098 USD for a shelter-based WiMAX BS driven by two generators.

- Nokia Siemens Networks (NSN): NSN's Flexi Base Station, which has been recognized as the world's most progressive mobile network technology at the annual GSMA Global Mobile Awards 2009, can be deployed with WCDMA/HSPA and upgraded to LTE with software alone. It is reported that a typical Flexi BS running WCDMA/HSPA consumes over 70% less than previous generations, without impacting performance. Moreover, Flexi can work without external air conditioning, typically bringing a 30% reduction in site energy consumption.

- Vodafone: Vodafone reports in [94] that by 2020 it will reduce its CO_2 emissions by 50% against its 2006/7, primarily by improvements in energy efficiency and increased use of renewable energy such as solar power, wind power and fuel cells.

Controlling or reducing the energy consumption in the telecommunication network equipment and related infrastructure is another way of improving the efficiency of communications networks (CNs). [95] presents a collection of ideas from operators and manufacturers on methods of reducing their operational energy use. The document focuses on telecommunication equipment and infrastructure equipment (power station, air cooling, control of equipment, etc.).

Until now, we have considered the energy consumption of access networks. To verify the effectiveness of the mentioned technologies and methods, it is important to accurately measure the energy efficiency of the network. Metrics need to be specified to adequately compare different configurations and to evaluate the efficiency from various perspectives. We refer the interested reader to [96], [97] and, [98] for such energy efficiency metrics. [96] characterizes a network's power consumption in Watts per unit area for given coverage and spectral efficiency requirements. Other metrics like Energy Consumption Ratio (ECR) and Energy Consumption Gain (ECG) is discussed in [97]. A cell's energy efficiency and area energy efficiency measured by *bit per Joule* and *energy efficiency per unit area* are investigated in [98].

2.3.3.2 Energy Efficient Solutions for Wired Connectivity

While perhaps secondary nowadays to access networks, the energy consumption in the wired network part of an M2M architecture is increasing in importance, especially as applications increasingly depend on data centers reached over the Internet. Research and development efforts include efficient hardware, energy efficient Ethernet [99] and, more recently, optimizing network traffic for energy conservation. Virtualization of load through cloud computing is clearly a very promising technology for scalable energy use in wired networks. In the rest, we overview these different avenues of research.

a) Energy Efficient Hardware: By using rapid heat-dissipating raw materials, highly efficient power supplies, intelligent cooling systems, and advanced silicon solutions, some switch vendors (e.g., 3Com, D-Link) have gained significant equipment-level improvements in energy efficiency. For example, D-Link's line of gigabit Ethernet switches for small and home offices can reduce power consumption by both hibernating unused ports and adjusting signal strength based on cable length. It is claimed that the DSG-2208 desktop switch can achieve up to 80% power savings, compared to conventional switches by the same manufacturer. It is worth noting that power supplies and processors are also typical heavily power consuming components of wired networks. Different processor manufacturers employ different strategies and technologies to conserve energy [100], [101].

b) Improving Energy Efficiency by Optimizing Network Traffic: As mentioned in the previous section, in order to improve energy efficiency, some vendors work from the bottom up and use the most efficient components available for their equipment. Other vendors, on the other hand, working from the top-down, seek to first optimize network traffic across the network to reduce the stress on equipment. According to [102]: "In the high-end market, the focus includes improving operating efficiency by controlling individual hardware components, and optimizing the efficiency of network-attached devices by using technologies such as virtualization". Virtualization is a key technology for energy-efficient operation of servers in data centers. It partitions computational resources and allows the sharing of hardware. Many services often need only a small fraction of the available computational resources of a data-center server. If such services are virtualized and run within a virtual machine, depending on their utilization,

many virtual machines can exist on a single physical server resulting in consolidation of resources and significant energy efficiency [103]. These virtual machines can be moved, copied, created, and deleted depending on management decisions. Due to the server consolidation, less hardware investment is needed overall, further reducing the energy wasted for cooling.

A good example of virtualization is the Cloud Computing, which allows services to run remotely in a ubiquitous computing cloud that provides scalable and virtualized resources. This technique allows peak loads to be moved to less populated parts of the cloud and it can provide higher utilization of the hardware through aggregation of a cloud's resources. It also facilitates the identification of the main sources of energy consumption, and the significant trade-offs between performance, QoS and energy efficiency. In fact, it is pointed out in [103] that cloud computing provides insight into the manner in which energy savings can be achieved in large scale computer services that integrate communication needs.

Gelenbe and Silvestri [104] have focused on optimizing wired network traffic in order to improve energy efficiency, proposing and studying a dynamic approach where link drivers and/or nodes are turned on or off in response to traffic load in the network, with ensuing changes in the paths followed by the traffic so as to meet the QoS (Quality of Service) needs of the flows. Optimization in the context of network routing is carried one step further by monitoring the current flows and predicting the future flows in the network. The authors also considered the design and implementation of the Energy Management System Middleware (EMS), a software component with the following tasks:

1) Observe ongoing traffic flows, monitor the status and power consumption of nodes.
2) Select a network configuration that offers sufficient QoS to ongoing and predicted flows, with lower energy consumption.
3) Manage and sequence dynamic changes in links and nodes, and reroute traffic accordingly.

The results of experiments ran on a network testbed [104] indicate that EMS can effectively reduce the energy consumption in wired networks, while guaranteeing an acceptable level of QoS.

c) Energy Efficient Ethernet: For data center managers and equipment vendors looking for greener alternatives, the IEEE 802.3az initiative offers further reducing the power consumed by Ethernet equipment. IEEE 802.3az, or the Energy-Efficient Ethernet (EEE) standard [99] offers a series of enhancements to the twisted-pair and backplane Ethernet networking standards that allow for less power consumption during periods of low link utilization. The goal of EEE is to reduce power usage by 50% or more while remaining fully backward compatible with already deployed equipment. Christensen et. al. claim in [99] that the adoption of the new IEEE 802.3az EEE standard will result in large energy and economic savings likely exceeding $400 million per year in the US alone.

2.4 Challenges and Open Issues

Having extensively discussed the above topics, we provide our view on various challenges and research opportunities in the area of M2M networks. As an M2M-AN is the core of the M2M network, we first list the M2M-AN related research issues, and then, discuss the general issues that need attention for sustainable and reliable operation of M2M networks.

2.4.1 Open Research Issues for M2M-ANs

1. **Channel modelling and link quality characterisation:** Due to the area of deployment, M2M-ANs may be subject to strong RF interference, and may be damaged due to harsh physical environmental conditions such as corrosion, cold, heat, and high humidity caused by weather, or the area of operation (factory, or home, etc.), not to mention the malicious acts of human beings. The above-mentioned conditions may cause the network topology and wireless connectivity to change when certain intelligent devices fail or the measurements are not suitable for drawing good conclusions. Hence, efficient wireless channel modelling and link quality characterisation methods need to be developed so that the system designers can employ to predict the performance of the network.

2. **Energy Efficiency:** The availability of energy harvesting capabilities change from application to application. Today, there exists numerous M2M applications that operate on limited battery power, requiring communication protocols, and resource allocation

schemes, developed for high energy efficiency. Energy efficient protocols such as routing solutions are needed where M2M-ANs are usually expected to function over years without having to change the battery. In [59], we present a comprehensive survey on wireless unicast routing alternatives for M2M-ANs. However, as also discussed by ETSI in [68], an M2M network should support all of the following communication modes; anycast, unicast, multicast and broadcast. This way, whenever possible, a global broadcast can be replaced by a multicast or anycast in order to minimize the load on the communication network. Hence, routing techniques needs to be developed for above-mentioned communication modes, by taking into account the long sleep cycles, changing radio environment, change of topology, and the limited power.

3. **QoS (Quality of Service):** There exist various M2M pplications that require different levels of QoS. Therefore, as also stated by [69], in order to enable efficient prioritisation of certain M2M applications that have some critical requirements to meet, such as those belonging to protection and control functions, algorithms that will help M2M-ANs to support different levels of quality of service (QoS) are required. For example, in intrusion detection based M2M applications, an alarm notification for the system requires immediate attention, and thus a realtime communication, where applications based on periodic reporting activities require reliable communications.

4. **Dynamic Environment:** As the topology of the intelligent devices (mostly sensor nodes) in an M2M-AN changes due to several reasons such as sleep mode schedule, and mobility or node failure, there is a need for dynamic protocols that can adapt to changing network topologies to make sure that the network functions as normal.

5. **Data Aggregation:** With the large scale development of M2M networks, and thus M2M-ANs, there are a large amount of information collected over time. Hence, researchers need to focus on methods of combining or aggregating, fusing or inferring data intelligently, so that a conclusion, on what action is needed or how to configure the parameters in the system for optimum functionality, can be drawn.

2.4.2 Open Research Issues for M2M Networks in General

1. **Standardization:** As shown in this thesis, M2M communications require an integration and convergence among various different communications systems. Therefore,

apart from individually designed protocols, standardization of a unified M2M architecture is highly demanded to promote rapid development and application of M2M networks. Recently, there has been a new and exciting standardization effort in ETSI. ETSI has constituted a technical committee, ETSI M2M committee, with the purpose to develop an end-to- end architecture for M2M communications. Moreover, according to [105], mobile operators around the world have been constructing platforms to integrate M2M services with infrastructure networks and launching M2M projects (e.g., GSM Association's Embedded Mobile Initiative) to accelerate the adoption of wireless interconnectivity of different M2M components.

2. **Protocol Re-design:** The current leading transmission protocols of the Internet used by the communication networks, TCP/IP, are reported to be inefficient [105] for M2M networks' traffic, due to the redundant and energy-wasting overhead compared to the low data volume needing to be transmitted. Hence, in near future, the researchers need to seek the ways of designing new transmission protocols that reflect the special needs of M2M communication.

3. **Security:** Last but not least, security is an essential requirement that needs to be met in all sub-layers of an M2M network, in order to ensure that the whole system functions smoothly and safe from any sorts of attack and intrusion. As, due to their need for dispersed and decentralized methods, conventional security models are not directly applicable to the highly distributed and low-cost devices used in M2M networks, new security models and methods that can protect privacy/security in every layer of an M2M network, needs to be explored.

CHAPTER 3

ENERGY EFFICIENT WIRELESS UNICAST ROUTING ALTERNATIVES FOR MACHINE-TO-MACHINE NETWORKS

Energy efficiency is one of the important design objectives for machine-to-machine network architectures that often contain multi-hop wireless subnetworks. Constructing energy-efficient routes for sending data through such networks is important not only for the longevity of the nodes which typically depend on battery energy, but also for achieving an environmentally friendly system design overall, which will be imperative as M2M networks scale in number of nodes as projected. The objective of this chapter is to provide a comprehensive look into energy-efficient routing alternatives to provide a reference for system designers as well as researchers.

By considering the fact that energy efficient routing is required in newly evolving M2M networks and, the fact that ad hoc and sensor networks are usually components of M2M networks, most of the energy-efficient routing algorithms surveyed in this thesis will be those that appeared within the literature on ad-hoc and sensor networks, primarily in the last decade.

This chapter provides a detailed account of energy-efficient unicast routing alternatives, with a particular focus on those based on additive link cost. One of the main contributions of this chapter and the thesis is a detailed and comprehensive classification summarized in Table 3.2. We believe that Table 3.2 could be useful as a reference to the reader who needs to quickly look up an algorithm with specific properties, rather than read the complete chapter. Note that, the work in this chapter is also available in [59].

3.1 Energy Efficient Routing

In wireless ad hoc and sensor networks, the problem of routing has received more attention than any other design and operation problem. Many wireless routing algorithms have been proposed in the last couple of decades. Flooding and broadcast routing is often necessary during the operation of the wireless network, such as to discover node failure and broadcast some information. Multicast routing, on the other hand, is very common in wireless networks, and it is used to communicate in a one-to-group fashion. Moreover, it involves wireless multicast advantage (WMA) [106] which means that if a node transmits a packet by spending high power, it is possible that more than one node receive its transmission. Finally, unicast is always in an end-to-end fashion and it is the most common kind of routing in networks. The case of unicast routing, although a special case of multicasting, involves no wireless advantage, however, choosing a good path from source to destination requires knowledge of node and link states. This is especially the case when battery lifetime maximization is an objective. Given a selected route, nodes on this route between the source and destination who act as routers deplete their energies with each packet they forward. Of course, there are other energy-consuming tasks, as discussed in detail by Ephremides in [107], in particular, often idle listening, or actively receiving data require significant current consumption, depending on the type of receiver. For example, the reference values of transmission power, P_t, receiving power, P_r, and power consumed in listening, P_l, for a Lucent silver wavelan PC card (802.11) are 1.3 W, O.9 W, and 0.74 W respectively [108]. FreeScale MC 13192 SARD (802.15), on the other hand, uses; $P_t = 0.1404W$, $P_r = 0.1404W$ and $P_l = 0.0018W$ [109]. In the transmitting mode of operation, an ad hoc battery operated node consumes energy in two ways [107]:

1. In the front-end amplifier that supplies the power for the actual RF transmission (the radiated energy as well as the internal heat losses in the antenna and the amplifier itself)
2. In the node processor that implements all the signal generation, formatting, encoding, modulation, memory access, and other signal processing functions.

The first one is known as the transmission energy and the second one is the processing energy. In receiving mode, the consumed energy is only of the processing type and includes the low-noise amplifier that boosts the output of the receiving antenna to levels suitable for

demodulation, decoding, buffering, etc. Finally, in the listening mode, a node typically listens but is not actively receiving. Hence, the energy consumed is again of the processing type but also possibly of some transmission type. The reason for this is a possible network protocol requiring a listening device to emit periodic beacon signals. As distance over which point-to-point transmission is made increases, transmission energy is bounded to be the dominant component of energy consumption. It is for this reason that many studies related to multihop wireless networks (e.g. [110]) have focused on transmit energy.

When the above-mentioned ad hoc node is capable of sensing as well (sensor nodes used in M2M-ANs), there is one more way of energy consumption; namely, the information collection (or sensing) based energy consumption. Although, this type of energy consumption seems to be important in designing energy efficient routing mechanisms, most of the algorithms mentioned in scope of this chapter do not consider this type of energy consumption. We refer the interested reader to DAPR protocol [111] (in Section 3.3.3.3) which is the only algorithm using this type of energy consumption in its link cost definition.

Gupta and Hirdesh argue in [112] that network technologies used in M2M networks should ensure availability, reliability and cost effectiveness, and that the mesh network backbones that can be used in M2M networks should be structured for optimized communication and energy-efficiency. There can be different definitions of energy efficiency for a routing algorithm [113]. For example, consider a sequence of packets that need to be sent from one source to a given sink. Minimizing the energy consumed for each packet transmission is an obvious solution that optimizes locally the energy consumption. In [114] some other objectives such as minimizing the variance in each node's battery power level, or the maximum node cost are discussed. However, focusing on individual nodes in the system instead of the system as a whole might quickly lead the system to a state the network is disconnected, although most nodes have high residual power. Shah and Rabaey suggest that [17] a more meaningful metric for routing protocol performance is network survivability. Network survivability depends on how well a routing protocol (or algorithm) is designed to use the energy of each node in the network efficiently and thus elongate the time that the network stays connected. This is also known as the network lifetime and it is defined as the time until network dies or becomes partitioned. Yet, it may not be so straightforward to define wireless network lifetime, as argued in [107]: Is it when the first node runs out of energy? Is it when a fraction of them does? Or is it when all nodes do? In networks where the nodes need to work collaboratively (such as

ad-hoc and sensor networks) after death of the first node, other nodes are loaded more heavily and deaths occur much faster [115], [116], [117]. Therefore, maximizing the time until the death of the first node seems to be key. Indeed, many studies in the scope of this chapter define network lifetime as the time until the first node runs out of battery power.

3.2 Link Cost Based Shortest Path Routing

In Shortest Path Routing (SPR), the goal is to send packets over a network in such a way that the path cost from the source to the destination is minimized. Modeling a network as a directed graph G, one assigns to each directed edge (n_i,n_j) in G a real number $d_{i,j}$ which represents the cost of using a particular edge in the network. In the case of an undirected graph, one can set $d_{i,j} = d_{j,i}$. If an edge does not exist between node i and node j, $d_{i,j} = \infty$. Let $(n_i,...,n_l)$ denote a path. Accordingly, its length is defined as

$$d_{1,2} + d_{2,3} + ... + d_{l-1,l} \tag{3.1}$$

The goal of shortest path routing applied on G is to find the minimum length path from n_i to n_l. The network graph mentioned above is an accurate depiction of the network topology if the nodes are interconnected with dedicated wired lines. However, it should be noted that in the wireless case, the notion of a link between the nodes, say n_i and n_j depends on the transmit power, channel variations, as well as other factors, and can be dynamic. Hence, it requires a separate treatment [107]:

In its simplest form, considering interference as noise, and taking a constant channel code rate, the criterion for successful reception can be written as follows:

$$SINR > \gamma \tag{3.2}$$

where γ is a threshold that depends on the detector structure, modulation/demodulation, and coding/decoding used and $SINR$ is the received Signal-to-Interference-plus-Noise Ratio at the receiver. $SINR$ depends on the channel characteristics, transmit and receive antennas, RF transmission power (P) and transmission rate (R), and, interference caused by other users. P and R determine the amount of signal energy packed in each symbol and because of this, they

are highly adjustable. Especially, P influences the amount of energy consumed on a link and determines which links are feasible and hence which paths can be used for routing to the final destination. Hence, in wireless networks, whether a link exists or not depends on the chosen values of P and R.

Shortest path routing requires using a link metric which defines the properties of an arbitrary link. In a wireless network, different link quality metrics can be defined: link bit error rate, delay, transmission energy, and residual energy. In principle, the metric of choice should map to the usual global objectives such as the total delay, throughput, blocking probability, and total energy consumption. In this thesis, we focus on the studies that define metrics in order that the resulting routing algorithm choosing the optimal paths will achieve a well defined global objective. We will revisit the topic of link quality metrics in Section 7.

Once link costs that map to global objectives are determined, the routing algorithm makes a straightforward implementation of SPR. At the basis of most methods described in this chapter lie the two well-known methods for calculating shortest paths, Bellman-Ford and Dijkstra algorithms, and it may be worthwhile to briefly describe these two algorithms before proceeding.

The Bellman-Ford algorithm computes shortest paths from each node to a given destination node by iterating on the number of hops. Let the destination node be 1. Let D_i denote the length of the shortest path from node i ($i \in [1, ..., n]$) to node 1, and let D_i^h denote the shortest path from node i to node 1 that contains at most $h \geq 0$ arcs. The algorithm for computing D_i is given by the following steps [18] :

1. Initial conditions. Set $D_1^h = 0, \forall h \in [0, 1, ...]$, $D_i^0 = \infty, \forall i \in [2, ..., n]$ and set $h = 0$.

2. Evaluate

$$D_i^{h+1} = min_j[d_{i,j}, D_j^h], \forall i \neq 1, \tag{3.3}$$

   and let $h = h + 1$.

3. If $D_i^h = D_i^{h-1}$ $\forall i$, stop. Let $D_i = D_i^h$ $\forall i$. Otherwise go to step 2.

The above computation assumed synchronous operation. In a distributed environment, a distance vector approach is taken to compute shortest paths, and some additional mechanisms

are included to cope with erroneous or outdated information, or link failures.The centralized Bellman-Ford algorithm terminates in at most n steps and has a worst-case complexity of $O(n^3)$. To determine all shortest paths in a network, it must be run n times, where n represents the number of destination nodes.

When the link costs (the arc lengths $d_{i,j}$) are nonnegative, an alternative to this method is Dijkstra's algorithm. Dijkstra's algorithm iterates on path length. To illustrate, let the destination node be 1, and let D_i, $i \in [1, ..., n]$, denote the shortest path length of the i^{th} closest node to node 1. Finally, Let P be a set of *permanently labeled* nodes for each of which the shortest path distance to node 1 has been determined. Then, apply the following steps [18]:

1. Initialization. Set $P = 1$, $D_1 = 0$ and $D_i = d_{j,1}$, for $j \neq 1$.

2. Find the next closest node. Determine $i \notin P$ such that

$$D_i = min_{j \notin P}[D_j]. \tag{3.4}$$

   Add node i to the set of permanently labeled nodes, i.e., $P = P \cup i$. If P contains all nodes, then stop; the algorithm is complete.

3. Updating of labels. For all $j \notin P$ set

$$D_j = min_{j \in P}[D_j, d_{j,i} + D_i] \tag{3.5}$$

   Go to step 2.

Dijkstra's algorithm terminates after $n - 1$ iterations and has a smaller worst-case complexity, $O(n^2)$, than that of the Bellman-Ford algorithm. We refer the interested reader to [118] and [18] for more detailed information about distributed implementation of these basic methods.

3.3 Shortest Path Based Energy Efficient Routing Algorithms

3.3.1 Energy Aware Routing Algorithms

Energy aware routing algorithms, in general, have the common objective of maximizing network lifetime by considering the residual battery energy when performing routing [114],

[119], [120], [121], [122], [123], [124], [125], [126]. These seek to perform well with respect to the objective of finding paths that consume minimum energy as well as the objective of finding paths which do not rely on nodes that are significantly depleted, without compromising either of these two conflicting objectives. In this section, we present a detailed overview of these algorithms.

3.3.1.1 MMBCR and CMMBCR Algorithms

Toh et al. [120], [127] proposed the online algorithms MMBCR (min-max battery cost routing) and CMMBCR (conditional MMBCR) to select energy-efficient source-to-destination paths. The MMBCR algorithm uses a min-max route selection technique. It chooses a path P for which the minimum of the residual energies of the nodes on P is maximum. Since MMBCR tries to avoid routes with nodes having the least battery energy among all nodes in all possible routes, the battery energy of each node is depleted more evenly as compared to previous schemes like MTPR (Minimum Total Transmission Power Routing) [128], MTE (Minimum Total Energy) routing [129], [127] and MBCR (Minimum Battery Cost Routing) [130], [127]. However, since MMBCR does not try to minimize the total transmission energy along a path, it may also lead to a high overall consumption.

Recognizing that to maximize network lifetime one needs to achieve some balance between the energy consumed by a route and the minimum residual energy at the nodes along the chosen route, a conditional variant of the MMBCR algorithm was also proposed in [120]. In this scheme (called CMMBCR), we look for a minimum energy source-to-destination path in which no node has residual energy below a threshold. Hence, the algorithm uses minimum-energy routing when there is at least one candidate path, where the remaining battery power (energy) in all the constituent nodes is above the battery protection threshold γ. Once one or more of nodes on all possible paths falls below γ, CMMBCR switches to MMBCR which equitably distributes the battery consumption among the different nodes. Thus, it protects against the early exhaustion of a few nodes. These algorithms are among the earliest solutions for extending the life span of ad hoc networks. For more detail about them, we refer the interested reader to the survey in [131].

3.3.1.2 Max-min zP_{min} Algorithm

In [121], the authors propose an online message routing algorithm, max-min zP_{min} , for the network lifetime maximization problem. Here, P_{min} is the energy required by the minimum energy path, and z is a parameter. The algorithm selects a path that uses energy at most $z.P_{min}$, while maximizing the minimum residual energy fraction. The residual energy fraction (energy remaining after route/initial energy) of node i after sending a message to j is defined as $R_{i,j} = (E_i(k) - w_{i,j})/E_i$ where $E_i(k)$ represents the current residual energy of node i and E_i is the initial energy level of node i. The authors use a general metric setting for $w_{i,j}$ where the energy consumption for a transmission depends on the distance between the sending and the receiving nodes:

$$w_{i,j} = k.d_{i,j}^{\alpha} \tag{3.6}$$

where k and α are constants for the specific wireless system (usually $2 \leq \alpha \leq 4$). The algorithm works as follows:

1. Find the minimum transmission energy path (the total transmission energy on this path is P_{min}) by using $w_{i,j}$ metric in Dijkstra's algorithm.
2. Find the minimum $R_{i,j}$ on that path and let it be R_{min}.
3. Find all edges whose $R_{i,j} \leq R_{min}$ and remove them from the graph.
4. Find the minimum transmission energy path on the new graph.
5. If the energy consumption $> z.P_{min}$ or no path is found then the previous shortest path is the solution, stop. Otherwise, go to step 2.

The authors in [121] provide a competitive analysis of their algorithm by comparing its performance to the optimal solution obtained by linear programming. However, it should be noted that in the analysis, it is assumed that the messages are generated cyclically, or in each interval of time the set of messages are the same. As the competitive bound depends on the amount of residual energy left over in the network as well as the periodicity of the messages, it is not clear how good the bound is. Despite the mentioned drawbacks, this approach has inspired many studies. The algorithm has the disadvantage of being centralized and requiring

knowledge of the power level of each node in the system. A the distributed version of it algorithm is proposed in [114]. The distributed version of max-min zP_{min} uses the distributed Bellman-Ford algorithm and requires n message broadcasts for each node when there is no clock synchronization. When all clocks are synchronized, only one message broadcast is needed.

3.3.1.3 Zone Based Routing Algorithm

In [122], Zone based routing was proposed. Zone based routing can be defined as the modified and scalable version of max-min zP_{min} algorithm. It uses a hierarchical approach in which the area covered by the sensor network is partitioned into small groups of sensors. Each group of sensors in geographic proximity are clustered together as a zone. Each zone is treated as an entity. Zone based routing algorithm which was improved and discussed in more detail in [114] consists of three small algorithms:

- An algorithm for estimating the power level of each zone
- An algorithm for computing the best path for the message within each zone (modified max-min zP_{min})
- An algorithm for computing a path for each message across zones

The algorithm mainly works as follows: The sensor nodes in a zone autonomously direct routing inside the zone. While doing this, they also participate in estimating the zone power level. Using this zone power level estimate (P_{est}) each message is routed across the zones. A global controller (the node with the highest power) manages the zones. If the network can be divided into a smaller number of zones, the scale for the global routing algorithm is reduced. The global information needed for sending each message across is summarized by P_{est} of each zone. A graph, called zone graph, is used to represent connected neighboring zone vertices. A link in this graph means that the current zone can go to the next neighboring zone in that direction. Each zone vertex has a unit power level. Each zone direction vertex is labeled by its P_{est} computed by a modified Bellman-Ford algorithm.

3.3.1.4 Energy Aware Routing Algorithm for Low-Energy Networks

Energy Aware Routing Algorithm proposed in [17] (called EAR-Low in this thesis) tries to ensure the survivability of low-energy networks. EAR-Low scheme uses sub-optimal paths occasionally to provide substantial gains. EAR-Low is a reactive protocol (such as the Ad-hoc On-demand Distance Vector Routing (AODV) [132] and directed diffusion [133].) It is different from previous reactive protocols in that instead of finding a single optimal path and using it for communication, it keeps a set of good paths and then, chooses one of them based on a probabilistic fashion. Choosing among multiple sub-optimal paths ensures that the optimal path does not get energy drained and the network degrades gracefully as a whole rather than getting partitioned. In order to achieve this goal, multiple paths are found between source and destinations. Then, depending on the energy metric, each path is assigned a probability of being chosen. When data needs to be sent from a source to a destination, one of the previously found paths is randomly chosen depending on the probabilities. By having paths that differ in time, the energy of any single path will not drain quickly. Longer network lifetime (longer network connectivity) is achieved as energy is dissipated more equally among all nodes.

The proposed Energy Aware Routing (EAR-Low) protocol consists of three phases [17]: The first phase is the *setup phase* in which localized flooding is used to find all routes between a source-destination pair and their costs. In second phase, *data communication phase*, the data is sent from sources to destinations by using the paths which are chosen probabilistically according to the energy costs (metrics). The last phase is the *route maintenance phase* in which localized flooding is performed by the destination node to keep the paths alive. The energy metric that is used to evaluate routes is an energy aware metric that has been proposed by Chang and Tassiulas [134]:

$$C_{ij} = e_{ij}^{\alpha} R_i^{\beta} \tag{3.7}$$

where C_{ij} is the cost metric between nodes i and j and, e_{ij} represents the energy used to transmit and receive on link (i , j). Here, R_i is the residual energy at node i normalized to the initial energy of the node and α and β are the weighting factors. These factors can be chosen appropriately to favor either the minimum energy paths or the paths with nodes having the most energy.

Compared to directed diffusion [133], this protocol provides an overall improvement of 40 percent in network lifetime. Moreover, difference in energy usage among nodes is lesser as compared to diffusion and, this results in 21.5 percent less average energy consumption. However, the approach necessitates gathering location information and setting up the addressing mechanism. Hence, comparing to directed diffusion, the route setup becomes complicated.

3.3.1.5 EAR and DEAR Protocols

The authors in [117] proposed EAR (Energy Aware Routing) and DEAR (Device and Energy Aware Routing) protocols for a heterogeneous wireless ad hoc network where there exist different classes of nodes. EAR is the implementation of the Distributed Bellman-Ford (DBF) routing protocol [18] which uses the following metric as the link cost function:

$$C_j = \sum f_i(R_i) \tag{3.8}$$

C_j is the cost of sending packet j from node n_1 to node n_k via intermediate nodes $n_2, ..., n_{k-1}$ and $f_i(R_i)$ denotes the cost or weight of node i. Since f_i represents a node's reluctance to forward packets, it was chosen as:

$$f_i(x_i) = \frac{1}{R_i} \tag{3.9}$$

where R_i represents the residual energy of node i. By using the reciprocal of residual energy as link cost, as the energy of a node decreases, the cost of using that node increases. Hence, that node is not chosen as a forwarding node and its energy is not depleted. The benefit of EAR protocol comes from the dynamic load balancing among different nodes. The authors in [117] compared the performance of EAR protocol with a few conventional protocols (AODV, DBF, DSR [135], WRP [136]) using the system lifetime as the performance measure. Simulation results have shown that the system lifetime was the highest when EAR was used and the percentage increase in system lifetime with respect to DBF also increased with an increase in the number of nodes. Moreover, the system lifetime increased as the edge density increased since the load was balanced over a larger number of routes. However, the authors mention that the savings are valid only for a static ad hoc network. When nodes move independently with respect to one another, it is not necessary that there will be multiple paths from a source to a destination at all times. Hence, the savings obtained by using EAR is small (or even

zero) due to the lack of multiple routes. By considering both the advantages and drawbacks of EAR protocol, the authors in [117] proposed DEAR protocol for such a heterogeneous ad hoc network where there exist two different classes of nodes; battery-powered nodes and externally powered nodes.

DEAR protocol is less dependent on the availability of the multiple paths; rather, it makes use of device awareness to enhance the routing. With both energy and device awareness, the system lifetime is further increased by taking advantage of the extra capability and resources of externally powered nodes while at the same time balancing traffic among the battery-powered nodes. DEAR actively redirects the packets to the (externally) powered nodes for power-saving operations. The device-aware redirect scheme is designed as follows:

Each node maintains a routing table and an additional redirect table. Unlike conventional routing tables, the routing table includes an additional field named as the device type (binary field in which 0 indicates battery powered and 1 indicates externally powered). The redirect table consists of the destination address and the address of the node to be redirected. We refer the interested reader to [117] for more information about the formation of these tables and the algorithms used to update the routing table and the redirect table. Whenever a routing table update is received, a node updates its routing table by using the update algorithm. After updating its routing table, the node browses through its routing table and determines the minimum cost to reach any externally powered device. Once a battery powered node receives a packet to be forwarded, it extracts the destination address from the header and looks at the corresponding entry in the redirect table. According to the redirect table entry, the node either forwards the packet to the next hop or redirects it to a particular node. Whenever an externally powered node receives a packet, it checks if the destination of the packet is one of its neighbors. If so, it unicasts the packet to that particular destination. If not, it boosts its transmit power to cover the entire network and then it unicasts the packet to its destination.

Reportedly, [117] the DEAR protocol achieves better system lifetime as compared to other considered energy efficient routing solutions from the literature, as well as EAR. Of course, this performance superiority increases with increasing number of powered nodes that this protocol exploits. Finally, as expected, it was shown that when these powered devices cover a larger area, DEAR can reduce the number of hops per route and the percentage increase in system lifetime compared to EAR increases.

3.3.1.6 EERP protocol

In [137], the authors proposed an Energy-Efficient Routing Protocol (EERP) to maximize the lifetime in sensor networks. The proposed protocol is similar to directed diffusion in certain ways. Both the Directed diffusion and EERP are sink-initiated and reactive routing protocols. Multiple paths are maintained from source and destination (sink). However, Directed diffusion sends data along all the paths at regular intervals, while EERP uses only one path at all times; the path that expends minimum energy at all intermediate nodes. The main aim of EERP is to find the route (path) with lowest energy from the source to the sink depending on the energy metric and data sent on that path.

The protocol has three phases [137]: Interest Propagation Phase in which localized flooding occurs to find all the routes from source to destination and their energy costs, Data Communication Phase in which the paths from source to destination are chosen according to the energy costs, and, Route Maintenance Phase in which localized flooding is performed intermittantly from destination to source to keep all the paths alive. EERP protocol is similar to the one proposed in [17] in that both protocols have the three phases mentioned above. However, these two protocols differ in Data Communication Phase. In Data Communication Phase of the protocol described in [17]: "The paths are chosen probabilistically according to the energy costs and each of the intermediate nodes forwards the data packet to a randomly chosen neighbor in its forwarding table, with the probability of the neighbor being chosen equal to the probability in the forwarding table". However, EERP does not consider any probability assignment for choosing neighbors. It uses merely the idea of shortest path (cost) routing. The data communication phase of EERP is realized by modifying the Bellman-Ford Algorithm to compute the least energy path from the source to the sink. The modified algorithm uses the link cost metric which was defined in eq(3.7) and keeps a list of tentative shortest paths, which are then iteratively refined. The algorithm operates as follows: Initially, all vertices (on the network graph) are unmarked and the path costs to each node are either the weights on the edges (C_{ij} in eq(3.7)) or ∞. The algorithm then marks the node that is head of the path from the source node whose cost is minimal among those paths whose heads are unmarked. The corresponding tentative path is declared final. Then, the algorithm updates the other path costs by computing the minimum between the previous path costs and the sum of the (final) path cost to the newly marked node plus the costs on the edges from that vertex [137]. The

procedure mentioned above continues until all vertices are marked.

Simulation results reportedly show that EERP consistently performs well with respect to energy-based metrics, e.g. energy consumption and network lifetime and hence increases the throughput. EERP protocol reduces the energy consumption per node as compared to Directed Diffusion Protocol (DDP) and provides an increase of 11.4 percent in network lifetime.

3.3.1.7 Flow Augmentation Algorithm

The flow augmentation (FA) algorithm [138] is a minimum cost path routing algorithm which uses link costs that reflect both the communication energy consumption rates and the residual energy levels at the two end nodes. The algorithm has received considerable attention as it can achieve a lifetime close to the optimal network lifetime obtained by solving the linear programming problem.

FA algorithm is mainly an extension to what has been presented in [134]. The main difference between two algorithms is that in [138], the problem is extended to include the energy consumption at the receivers during reception. Moreover, the problem is formulated for fixed information-generation rates as well as for some arbitrary information-generation process. In fixed information-generation case, the amount of information to be generated within a certain time interval is known a priori. The algorithm aims to find the flow that maximizes the system lifetime under the flow-conservation condition[1]. On the contrary, in arbitrary information-generation case, the amount of total information generated in some time interval is not known a priori but FA algorithm makes routing decisions on the fly as new information is generated.

Chang and Tassiulas [138] observe that the flow augmenting path should avoid nodes with small residual energy since the main aim is to maximize the minimum lifetime over all nodes. By taking this into account, the FA algorithm uses a new link cost metric which combines the above mentioned parameters in one:

$$C_{ij} = (e_{ij}^{t})^{x_1} \underline{E}_i^{-x_2} E_i^{x_3} + (e_{ij}^{r})^{x_1} \underline{E}_j^{-x_2} E_j^{x_3} \tag{3.10}$$

where

[1] Flow-conservation condition: At node i for each commodity c, the sum of information-generation rate and the total incoming flow must equal the total outgoing flow

$$e_{ij}^{t} = e^{T} + \epsilon_{amp} d_{ij}^{4} \quad and \quad e_{ij}^{r} = e^{R} \tag{3.11}$$

where e^T, e^R, ϵ_{amp} and d_{ij} are the energy consumed in the transceiver circuitry at the transmitter and the receiver, respectively, the energy consumed by the output amlifier, and, the distance between node i and node j. x_1, x_2, and, x_3 in eq. (3.10) are the weighting factors for each item and, E_i, E_j, and, $\underline{E}_i$, $\underline{E}_j$ represent the initial and the residual energy levels of nodes i and j respectively. The transmission energy consumed at node i to transmit a data unit to its neighboring node is denoted by e_{ij}^{t} whereas the energy consumed by the receiver is denoted by e_{ij}^{r}.

In fixed information-generation case, if there is enough residual energy for a packet, the path cost in FA algorithm is computed by the summation of the link costs ,$\sum C_{ij}$,on the path. After running the Bellman-Ford algorithm in order to determine the shortest cost paths, if any of the commodities cannot find a path to its destination, then, the FA algorithm stops. Otherwise, the algorithm augments $\lambda Q^{(c)}$[2] on each shortest cost path of its commodity and update the residual energy accordingly. In arbitrary information-generation case, on the other hand, instead of $\lambda Q^{(c)}$ of flow, all packets generated in between the routing information updates are assigned the available shortest cost path.

FA algorithm provides significant improvement over others such MTE, CMMBCR [120] and Max-min zPmin [121] in terms of maximizing the system lifetime, (or the amount of information transfer between the source and destination nodes) under limited energy resources. Simulation results are claimed to show that, in the fixed information-generation case, the average gain in the system lifetime obtained by FA algorithm can be about 50-78 percent compared to MTE, whereas in the arbitrary information-generation case the lifetime obtained by FA algorithm can be more than three times longer than that of MTE. Simulation results with both fixed and arbitrary information-generation process models also indicate that the FA algorithm can achieve network lifetime that is very close to the optimal network lifetime obtained by solving the linear programming problem.

[2] λ is the augmentation step size which is equivalent to the amount of information routed between routing information updates and, $Q^{(c)}$ is the information generation rate for commodity c

3.3.1.8 CMAX Algorithm

The network lifetime competitive ratio results presented in [121] motivated the study reported in [123]. The algorithm proposed in [123], CMAX, is known as a very competitive online algorithm which then inspired many other studies. The main objective of CMAX is maximizing the total number of messages that can be successfully sent over the network (network capacity) without any information regarding future message arrivals or message generation rates. CMAX uses a combined cost metric (the energy consumed and the residual energy). Hence, it needs knowledge of residual battery energy at each node. Moreover, the algorithm uses only one shortest path computation. It works as follows:

1. Consider routing message k on the network *G=(N,A)*. Eliminate all links *(i,j)* $\in A$ for which $E_i(k) < e_{ij}$ and form a reduced network.
2. Associate weights w_{ij} with each link *(i,j)* in the reduced graph.
3. Find the shortest path from s_k to d_k in the reduced graph with link weights w_{ij}.
4. Let γ_k be the length of the shortest path found in Step 3 ($\gamma_k = \infty$ if no path was found). If $\gamma_k \leq \sigma$, route the message along the shortest path, otherwise reject it.

The authors use the following metric for w_{ij}:

$$w_{ij} = e_{ij}(\lambda^{\alpha_i(k)} - 1) \tag{3.12}$$

where

$$\alpha_i(k) = 1 - \frac{E_i(k)}{E_i} \tag{3.13}$$

Here, λ and γ are two constants chosen appropriately, $\alpha_i(k)$ is the fraction of node i's energy that has been used at the time message k arrives and e_{ij} is equivalent to the metric defined in eq. (3.6). Unlike [134] and [121] which aim to maximize the network lifetime, CMAX algorithm was designed to maximize the network capacity. Despite the choice of different objective, simulation results show that the CMAX algorithm outperforms other algorithms proposed before with respect to optimizing both lifetime and capaticy. Certainly, the most attractive part of the study is the part in which the authors show that the algorithm achieves a logarithmic competitive ratio. In order to obtain the competitive ratio result, the authors permit admission control (by using step 4 of the algorithm) so that the algorithm can occasionally

reject the messages that will overuse the network resources. However, it was shown in [123] that even if the CMAX algorithm is run without the admission control option, its performance is excellent. Distributed version of CMAX was also proposed in [123] and simulation results has shown that by an appropriate choice of the broadcast distance and broadcast frequency, D-CMAX can achieve performance close to CMAX.

3.3.1.9 OML Algorithm

Park and Sahni improved the idea used in [123] and they proposed OML (Online Maximum Lifetime) algorithm [119]. OML is an online heuristic whose main objective is to maximize the network lifetime. Although both algorithms use exponential metrics as well as other similar ideas, OML differs from CMAX in many ways. First, OML performs two shortest path computations to route each message (where CMAX uses only one). Secondly, OML performs two pruning operations. Third, OML algorithm uses a weighting function which assigns a high weight to edges whose use on a routing path cause a node's residual energy to become low. More specifically, OML works as follows:

1. Consider routing message k on the network $G=(N,A)$. Eliminate all links $(i,j) \in A$ for which $E_i(k) < e_{ij}$ and form a reduced network (G'=(N',A')).
2. Find the minimum energy path P_i' from s_k to d_k in G'. If there is no such path P_i', the routing request fails, stop.
3. Compute the minimum residual energy, *minRE*, for nodes other than d_k on P_i'.
4. Eliminate all links $(i,j) \in A'$ for which $E_i(k) - e_{ij} < minRE$ and form a reduced network (G"=(N",A")).
5. Compute the weights w_{ij} with each link (i,j) in G".
6. Find the shortest path P_i' from s_k to d_k in G" with link weights w_{ij}.
7. Route the message along the shortest path, otherwise reject it.

The authors use the following metric for w_{ij}:

$$w_{ij} = (e_{ij} + \rho_{ij})(\lambda^{\alpha_i(k)} - 1) \tag{3.14}$$

where

$$\alpha_i(k) = \frac{minRE}{E_i(k)} \tag{3.15}$$

and

$$\rho_{ij} = \begin{Bmatrix} 0 & E_i(k) - e_{ij} > minRE \\ c & otherwise \end{Bmatrix} \tag{3.16}$$

Here, λ and c are two constants chosen appropriately, e_{ij} is equivalent to the metric defined in eq. (3.6). $E_i(k)$ is the current energy at node i just before the route and, $\alpha_i(k)$ is the fraction of node i's initial energy. As it can be seen from the equation above, OML and CMAX algorithms have $(\lambda^{\alpha_i(k)} - 1)$ in common. Although, both algorithms have $(\lambda^{\alpha_i(k)} - 1)$ term in the edge weighting function, the two algorithms use different $\alpha_i(k)$ functions. In the case of CMAX, $\alpha_i(k)$ (3.13) function discourages the use as relays of nodes that have depleted a large fraction of their initial energy. However, these nodes might still have a large amount of energy remaining. OML algorithm prevents such a scenario by discouraging the use as relays of nodes whose current energy is low. The authors perform various analyses by changing transmission radius and node density and, simulation results show that OML outperforms CMAX with respect to network lifetime, and its performance is less sensitive to the selection of heuristic parameters. However, it should be noted that although OML performs better than CMAX, OML has few drawbacks. The main drawback is that the network lifetime competitive ratio results presented in [123] does not hold for OML. Another drawback is the extra complexity added by comparisons and the second shortest path selection. Despite these drawbacks, one should know that OML is superior to previously published heuristics for lifetime maximization. Distributed version of OML was also proposed in [119].

3.3.1.10 E-WME Algorithm

In [124], the authors presented a routing framework in which they formulated and solved the problem of energy-aware routing with energy replenishment. In contrast to the previously discussed studies in which the resources used were never recovered and receiving energy was not considered, in this study, the receiving energy of each node was considered and the resources of the network were allowed to be replenished by per-node processes. The authors developed an energy model to characterize the performance of such network in the presence

of energy constraints. The energy model captures heterogeneous energy sources (different replenishment rates, battery sizes, etc.) and allows routing in an energy-opportunistic way. By using this energy model the authors proposed E-WME (Energy-opportunistic Weighted Minimum Energy) algorithm.

The proposed algorithm is shown to achieve a competitive ratio that is asymptotically optimal with respect to the number of nodes in the network. E-WME algorithm leverages the work in [123]. If no nodes with renewable energy sources exist, the E-WME algorithm reduces to that of [123]. E-WME is easy to implement since it requires local short-term energy replenishment information and assumes no knowledge about the statistical information on the packet arrivals. The authors describe E-WME algorithm for two different cases; case of constant replenishment rate (constant case) in which the rate of energy replenishment of each node is constant (in time) and the general case in which time-varying replenishment rate was allowed in each node. The authors proposed two similar metrics for these two cases. For the sake of simplicity, we will only mention the metric for the first case (constant case). We refer the interested reader to [124] for information about the metric used for the second case and more detailed information about the analysis of the algorithm. For the constant case, the algorithm uses the following cost metric associated with each node *n*:

$$w_n(j,R) = \frac{u_n}{\gamma_n log\mu}(\mu^{\lambda_n(j)-1})l(j)e_n(R(j)) \tag{3.17}$$

where

$$\lambda_n(j) = 1 - \frac{P_n(j)'}{u_n} \tag{3.18}$$

Here R is the path from the source to the destination, u_n is the battery capacity of node n, $\lambda_n(j)$ is the fraction of the maximum storable energy used up to node n and $P_n(j)'$ is the residual energy at node n when considering routing request j. Lastly, $l(j)e_n(R(j))$ represents the energy requirement for packet j of length $l(j)$ and μ is an appropriately chosen constant. The algorithm works as follows:

1. Consider routing message j on the network $G=(N,A)$.

2. For an incoming routing request j, check if the least cost route R from s_j to d_j satisfies

$$Cost_R(j) = \sum w_n(j,R) \leq \rho(j) \tag{3.19}$$

where $\rho(j)$ is the revenue gained by forwarding the j^{th} packet.

3. If yes, accept the request and route the packet on *R(j)*.

4. Otherwise, reject the request.

By defining the cost metric as an exponential function in node residual energy, the E-WME is capable of closely adapting to the changes in the network energy profile and provides a clear guidline of how to balance the importance of residual energy, the transmit and receive energies and the quality of replenishment. Simulation results showed that E-WME has better throughput than other algorithms such as MBCR [130], CMMBCR [120] and Max-min zPmin [121]. E-WME has several advantages: It is optimal in the sense of minimizing the competitive ratio and strikes a balance between saving communication cost and distributing the network load. Moreover, it can be integrated with distance-vector-like proactive and on-demand routing protocols. However, by considering the fact that many sensor nodes are battery powered and can not be replenished, this algorithm becomes application specific. Moreover, we believe that the performance of this algorithm should be compared with CMAX and OML which appear to outperform MBCR, CMMBCR and Max-min zPmin algorithms.

3.3.1.11 SWP Algorithm

The authors in [125] suggested that a good energy-aware routing technique should balance two different goals: choosing a path with maximal residual energy and choosing a path with minimal energy consumption. Hence, they proposed a two-phased energy-aware routing strategy that balances these conflicting objectives by transforming the routing problem into a multi-metric widest path problem. The authors claim that the proposed Shortest Widest Path (SWP) algorithm outperforms the best known online algorithm in literature (OML). SWP algorithm uses two different metrics: The first metric was defined as the residual energy along a path (the minimum energy level of any node in the path). The second metric was the energy consumed along a path (the sum of the weights on the edges along the path). The residual energy of a path is a concave metric, whereas the energy consumed along a path is an additive metric. In [125], the authors presented a polynomial time combinatorial technique which can provide a good balance between these metrics by first maximizing the concave metric and then minimizing the additive metric. The auhors justify in their study why this order of

optimization - concave first, additive second - is better than the other possible order - additive first, concave second. The SWP algorithm consists of two different phases in which it uses these two metrics. The algorithm works as follows:

1. Consider a network graph *G=(V, E)*.
2. Modify *G* into an energy graph *EG = (V, E')* as follows: Leave the vertices intact but replace each single undirected edge in *G* with two directed edges such that the weight of one directional edge will be equal to the difference between the originating node's energy level and the transmission cost along the edge.
3. Given a source node *s* and a destination node *t*, run the two-phased routing algorithm on *EG* to find a suitable path between *s* and *t*.
4. Phase I: Apply a variant of the Dijkstra's algorithm (described in [125]) to find a path with the maximum residual energy and let it have a residual energy of B (there could be several paths in the network between *s* and *t* with a residual energy of B).
5. Let E" be the set of edges whose residual energy is less than B. Prune those edges from *EG* and form a pruned graph, *EG'*. (Pruning operation was explained in detail in [125])
6. Phase II: Use Dijkstra's algorithm to find the least energy cost path on *EG'*.
7. Restore all edges back in *EG* and go to step 3.

The simulation results showed that the performance of the proposed technique, shortest-widest path (SWP), is superior than that of the best known routing approach proposed in the literature (OML heuristic proposed by Park and Sahni [119]). However, unlike the lifetime definition used for OML, the results were obtained under the assumption that the lifetime was the number of packets that could be transferred in the network until *s* session failures occur (s is a parameter to be set by the network manager). According to the simulation settings used in [125], SWP outperforms OML with respect to both network lifetime and the average residual energy in the nodes for several values of transmission radii. Moreover, SWP consistently outperforms OML as the node density increases. The authors mention that the proposed approach can be easily combined with other QoS metrics such as delay, which can be beneficial for resource constrained networks.

3.3.1.12 SWCRP and SFWP Algorithms

In addition to the shortest widest path (SWP) discussed earlier, Mohanoor et al. proposed two other online energy aware routing algorithms: SWCRP and SFWP [126]. These algorithms too are two-phased strategies and are derivatives of the SWP algorithm. As mentioned earlier, SWP first maximizes the concave metric (the residual energy of a path) and then minimizes the additive metric (energy consumed along a path). The derivatives of SWP aim to provide a good balance between the concave and the additive metrics. The first algorithm, which is called the Shortest Width Constrained Path or Shortest Width Constrained Residual Path (SWCRP), finds paths with a suitably high residual energy, and then minimizes the energy consumed along such a path. By sacrificing the high residual energy of the absolute widest path, the algorithm uses a path with a slightly lesser (compared to maximum) residual energy but which consumes less energy along the path. In order to do this, the edges whose widths are below a certain lower bound are removed (temporarily) from the graph. Then, the minimum energy path is computed on the remaining edges, and this path is used for routing.

The second algorithm, Shortest Fixed Width Path (SFWP), is similar to the first algorithm in the sense that it finds a minimum energy path among the paths that have a high residual energy. However, unlike SWCRP, SFWP does not change the residual energy with each route calculation. Instead of this, it fixes the width (residual energy) of the path at a certain value. Then, it prunes the edges with residual energy which is less than the fixed value, and finds the minimum energy path on the pruned graph. The algorithm repeats this procedure until no path can be found for the given width, at which point the width is decreased (by a constant factor) and so on, until the source and destination get disconnected. In [126], the authors compare the performance of the proposed algorithms with the on-line maximum life-time (OML) heuristic proposed by Park and Sahni [119] and the max-min zP_{min} algorithm proposed by Li et al. [121]. Previously, we had mentioned that OML was the best known routing approach proposed in the literature. The simulation results show that using the widest path approach usually improves the network lifetime and the widest path (SWP) and its derivative algorithms (SWCRP and SFWP) are able to send more packets at fewer energy cost. Because of this, there is more residual energy available for the nodes compared to other algorithms. Hence, the performance of the proposed algorithms is superior to both the max-min zP_{min} algorithm and OML algorithm.

By considering the fact that it is highly desirable to have a distributed implementation of any routing algorithm, Mohanoor et al. also developed distributed versions of their algorithms. The distributed algorithms work as follows: A rooted spanning tree is constructed on the wireless network graph. Then, the global information is collected in such a way that each node sends all the information it has to its parents all at once. Once the complete network information is collected at the root, the root transmits this information to all the nodes (descendants) using the links on the spanning tree. The performance of the distributed versions of SWP, SWCRP and SFWP algorithms with varying transmission radius and varying node densities are presented in [126]. Similar to the centralized case, simulation results show that the distributed implementations of SWP and its derivatives lead to better network lifetimes than the distributed implementations of OML and max-min zP_{min}.

3.3.1.13 OLSRE Protocol

OLSR (Optimized Link State Routing), [139], is a proactive routing protocol where nodes periodically exchange topology information in order to establish a route to any destination in the network. OLSR uses multipoint relays (MPRs)[3] to minimize flooding of control traffic and thus reduce the number of retransmissions of broadcast messages. The OLSR routing protocol has been standardized by IETF [139]. In [140], Mahfoudh and Minet extended the OLSR routing protocol to respect energy efficiency concerns. This extended version is called energy efficient routing based on OLSR (OLSRE) (later called EOLSR in [141]). Instead of using the number of hops to compute the shortest path (in case of OLSR), OLSRE uses a new metric that takes into account the energy consumption. According to this new metric, the cost of transmitting a packet is computed as follows:

$$cost_{transmission}(i) = E_{trans} + n * E_{rcv} \tag{3.20}$$

where n is the number of non-sleeping nodes belonging to the interference zone of the transmitter i, and, E_{trans} and E_{rcv} represent the energy dissipated in transmitting and the energy dissipated in receiving a packet respectively. The authors use the following definitions for E_{trans} and E_{trans}:

[3] Using MPRs reduces the size of the control messages because, rather than declaring all its links in the network, a node declares only the set of links with its neighbors that have selected it as MPR

$$E_{trans} = P_{trans} * Duration \tag{3.21}$$

$$E_{rcv} = P_{rcv} * Duration \tag{3.22}$$

where *Duration* is the transmission duration of a packet, and, P_{trans} and P_{rcv} represent the transmission power and reception power respectively. After defining the link cost as mentioned above, the cost of an end-to-end transmission on any path P is computed as follows:

$$Cost(P) = \sum_{i \in sender(P)} cost_{transmission}(i) \tag{3.23}$$

The goal of OLSRE is to find the path that minimizes this cost. Hence, in OLSRE, each node computes its route of minimum energy towards any other node in the network, using the Dijkstra algorithm, with $cost_{transmission}(i)$. The above-mentioned routing strategy used by OLSRE is called (one hop-by-hop energy efficient routing) RE in the thesis.

As it can be understood from the discussion above, the link cost used in RE does not consider the residual energy (energy-awareness). The main reason for including this protocol in this category (Energy aware routing algorithms) is that the residual energy is taken into account in MPR selection phase. Unlike OLSR, which takes intermediate nodes as MPRs, in OLSRE MPRs are selected according to the residual energy of themselves and their one hop neighbors. These new MPRs are called (Energy MPRs) EMPRs. To avoid frequent route changes and assure load balancing, the selection of EMPRs is changed only when the topology changes or the residual energy decreases over a given threshold. Simulation results presented in [140] show that this protocol uses the minimum energy consumption path, and, is more efficient than multipath routing. Moreover, compared to OLSR, OLSRE maximizes both network lifetime and user data delivered. Indeed, the authors claim that RE prolongs the network lifetime of 50% compared to OLSR for a network of 200 nodes. At this point, it should be noted that, the network lifetime evaluated in [140] differs from the one (the first node failure) used by most of the studies mentioned in this chapter. The network lifetime presented in the results is the time to the first network partitioning. As soon as the network is no longer connected, vital information can no longer be transferred to its destination.

3.3.1.14 FML and FMO Algorithms

Fuzzy maximum lifetime (FML) algorithm and fuzzy multi-objective (FMO) algorithm are two efficient online routing algorithms developed by Minhas et. al. [142]. FML attempts to maximize the network lifetime objective whereas FMO strives to simultaneously optimize lifetime as well as the energy consumption objectives. The distinguishing aspect of these algorithms is the novel use of fuzzy membership functions and rules in the design of link cost functions. Among these two algorithms, FML is the most appealing one since, as claimed by the authors, it outperforms the OML heuristic both in terms of the network lifetime and the average energy consumption. FML algorithm operates as follows: When a routing request is initiated, a fuzzy lifetime membership is computed for each edge using the following equation:

$$\mu_{lt}^{ij} = \begin{cases} 1-(\frac{1-\gamma}{1-\alpha}.(1-\frac{re(v_i)}{\sigma})) & if\ \alpha.\sigma<re(v_i)\leq\sigma \\ \frac{\gamma}{\alpha.\sigma - TX_{ij}}.(re(v_i) - TX_{ij}) & if\ TX_{ij} < re(v_i) \leq \alpha.\sigma \end{cases} \tag{3.24}$$

where α and γ are algorithmic parameters, and, σ is the initial energy level which is the same for all nodes. TX_{ij} denote the energy expended in transmission of a k-bit packet and is modeled as defined in [142]. Finally, $re(v_i)$ and ce_{v_i} represent the residual energy and current energy of node v_i respectively.

As it can be seen from the expression above, the membership function strongly discourages the inclusion, on the selected routing path, of those intermediate nodes that have depleted their energy beyond a certain threshold value. The next step is to assign a weight, w_{ij}, to each link in WSN using the following expression:

$$w_{ij} = 1 - \mu_{lt}^{ij} \tag{3.25}$$

Following the weight assignment, the maximum lifetime path between a source and a destination is found by using Dijkstra's algorithm. Experimental results presented in [142] show that with rising node density, FML shows a consistent increasing trend in the obtained lifetime regardless of the transmission radius used, whereas OML is not able to show a similar increasing trend at higher transmission radii. It also seems that FML algorithm has a complexity advantage over the OML algorithm since it requires only one shortest path computation. FMO

algorithm, on the other hand, is more complex than FML algorithm since it tries to simultaneously optimize two routing objectives: maximizing network lifetime and minimizing energy consumption. FMO offers a flexible control over choosing a desired balance between the two routing objectives. The FMO algorithm operates in a similar manner as the FML except the computation of two additional parameters that need to be used in assigning weights for each link; namely, *fuzzy minimum energy membership* and *multiobjective membership* [142]. Therefore, rather than providing further detail about this algorithm, we refer the interested reader to [142]. A multipath extension to [142] is also available. The multipath version is called FML-MP [143], and, strives to maximize the network lifetime metric by distributing the source-to-sink traffic for a given routing request along a set of paths.

3.3.1.15 Keep-Connect Algorithms

Most of the algorithms explained in this chapter consider the network lifetime as the time the first node failure happens. However, Pandana and Liu claim in [1] that in many practical sensor applications, the death of the first node may not influence the information collection task, and therefore, the network lifetime should be defined as the time until there is no route from any source to any destination (the time until the network becomes disconnected/disintegrated). Using this definition as the network lifetime, Pandana and Liu argue that the network connectivity is an important criterion that needs to be considered in a routing algorithm. Thus, the authors propose a class of routing algorithms called keep-connect algorithms that use computable measures of network connectivity in determining how to route packets. The algorithms embed the importance of a node when making the routing decision. The importance of a node is quantified by the Fiedler value[4] of the remaining network graph when that particular node fails. The keep-connect algorithms are; MHKC (Minimum Hop while Keeping Connectivity), MMREKC (Max-Min Residual Energy while Keeping Connectivity), MMKC (Max-Min remaining Connectivity) and MTEKC (Minimum Total Energy Keeping Connectivity algorithm). MHKC, MMREKC and MTEKC are modified versions of the Minimum Hop (MH) [144], MMRE [145] and MTE [145] algorithms respectively, and, MMKC is a special case of MMREKC algorithm in which the residual energy of nodes is set to 1. As the MTEKC algorithm outperforms the other proposed algorithms in [1], we only provide the link

[4] The Fiedler value qualitatively represents the connectivity of a graph in the sense that the larger the Fiedler value is, the more connected the graph will be.

cost and SPR (shortest path routing) method related details of these algorithms in Table 3.1 and refer the interested reader to [1] for more detail about these algorithms.

Table 3.1: Link costs and SPR methods used in MHKC, MMREKC and MMKC algorithms [1]

Algorithm / Method	Link cost
MH / Standard Dijkstra	$c(u,w) = 1$
MHKC(y) / Standard Dijkstra	$c(u,w) = \frac{1}{2}[W(u)^y + W(w)^y]$
MMRE / Modified Dijkstra	$c(u,w) = \min\{\underline{\mathcal{E}}_u(t), \underline{\mathcal{E}}_w(t)\}$
Variant of MMRE / Modified Dijkstra	$c(u,w) = \underline{\mathcal{E}}_u(t) + \underline{\mathcal{E}}_w(t)$
MMREKC(y) / Modified Dijkstra	$c(u,w) = \underline{\mathcal{E}}_u(t)W(u)^y + \underline{\mathcal{E}}_w(t)W(w)^y$
MMKC(y) / Modified Dijkstra	$c(u,w) = W(u)^y + W(w)^y$

In table 3.1, $\epsilon_u(t)$ and $\epsilon_w(t)$ represent the residual energy at time t for transmitting node u and receiving node w, respectively. $c(u,w)$ is the link cost between nodes u and w, $W(.)$ is the connectivity weight calculated by using the *Keep-Connect* algorithm (keep-connect using Fiedler value) proposed in [1], and y determines how important the connectivity weight should affect the routing cost. Being different from the above-mentioned algorithms, the MTEKC algorithm uses the following link cost:

$$c(u,w) = e_t(u,w)W(u)^y + e_r(u,w)W(w)^y \tag{3.26}$$

where $e_t(u,w)$ and $e_r(u,w)$ are the transmit and received energy for delivering a packet from node u to node w. MTEKC operates as follows:

1. For any source-destination pair, find the MTE path with edge cost $c(v_i, v_j)$ by using the Dijkstra algorithm.

2. If a node dies, recompute the alive nodes' connectivity weight using *Keep-Connect* algorithm. Go to Step 1.

MTEKC algorithm minimizes the total transmit energy while trying to keep the remaining network as connected as possible. Extensive simulations performed in [1] indicate that MTEKC not only achieves 10%-20% better network lifetime and total delivered packets compared to MTE algorithm, but also, is more robust in terms of algebraic network connectivity. However, it should be noted that the algorithm never achieves both the best and energy efficient and the

most robust connectivity route. A distributed version of MTEKC, that nearly achieves the performance of the centralized version, is also proposed in [1]. The distributed MTEKC is based on the distributed reinforcement learning routing algorithm [146], and, can be characterized as a version of distributed Bellman-Ford algorithm that performs its path relaxation step asynchronously and online with edge cost defined as weighted energy required to transmit packet in that hop. We refer the interested reader to [1] for more information about the distributed version of MTEKC.

3.3.2 Performance Analysis of CMAX and OML Algorithms

Having observed that CMAX and OML are among the best early competitors for lifetime maximization (with lower complexity compared to those with superior lifetime), and have frequently used as benchmarks in the literature, we shall devote this section to a comparison of these two algorithms. We performed simulations by using the same settings used in [119], on a randomly generated simulation topology containing 20 sensors deployed on a 10×10 grid. Each node has the same initial energy. Moreover, for each topology, different realizations of packet streams are randomly generated. Average network lifetime over 10 different networks is plotted with respect to λ in Figure 3.1. In addition to this, we tested the algorithms on some other network topologies and we have seen that the performances of the algorithms are dependent on changing network topologies. However, as mentioned in [119], OML performed better than CMAX in all scenarios. Moreover, as it can be seen from figure 3.1, OML algorithm was less sensitive to the selection of λ and sent more packets until the first node died (within the network lifetime).

Figure 3.2 and 3.3 illustrate the average residual energy levels of 20 nodes after the first node dies, in the cases of using CMAX and OML algorithms. Every line in the figures is a measured average residual energy level (over 10 networks and 10 routing sequences) of the nodes, for the corresponding λ value. When λ increases, the residual energy levels decrease which means that the algorithm sends more packets by using nodes' energies efficiently. As it can be seen from the figures, in the case of using OML algorithm, the average residual energy levels of the nodes are much lower than the ones for CMAX algorithm. Moreover, OML is less sensitive to changing λ value. This means that OML algorithm sends more packets within the lifetime of the network and hence it spends more energy. Moreover, the OML

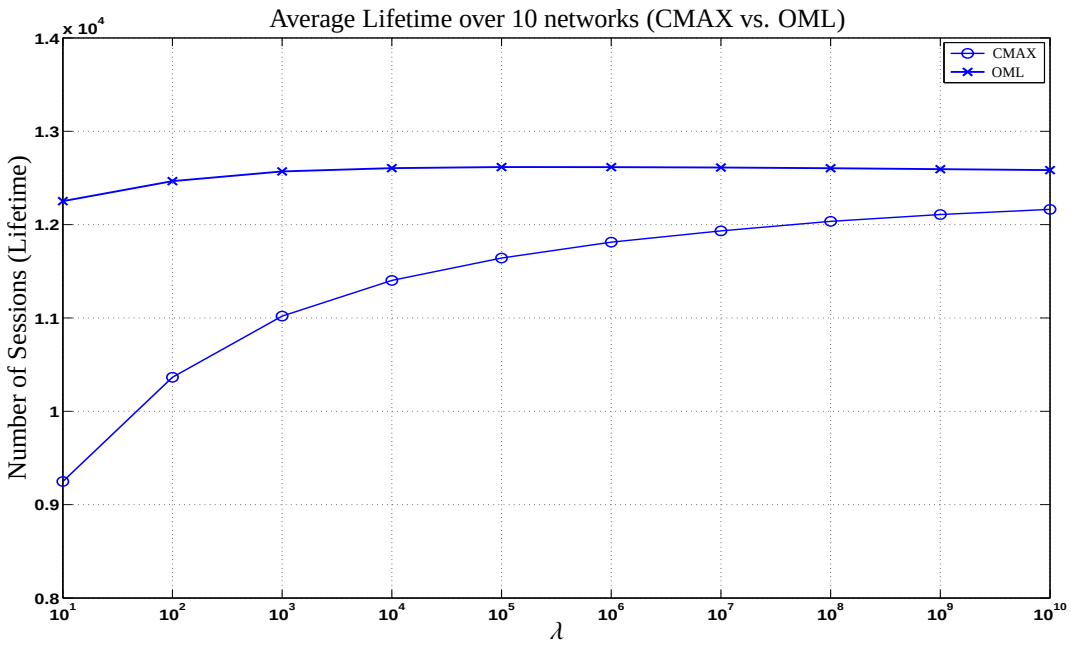

Figure 3.1: Average network lifetime over 10 different networks (topologies consisting of 20 nodes) each of which use 10 different routing sequences. In each network and routing instance, the nodes start with an initial energy of 30. The lifetime values plotted indicate the average value of the network lifetime obtained by averaging over $10 \times 10 = 100$ instances for that specific λ value.

results show that after the death of the first node, other nodes in the network have similar and very low energy levels. In contrast, residual energy levels in CMAX exhibit a quite high variation after the death of the first node. Hence, although many nodes have enough energy levels to send more packets, because the first node dies early, they can not send their packets within the lifetime.

3.3.3 Delay Optimization and QoS Related Energy Efficient Routing Algorithms

In the design of communication as well as networking techniques, there is a fundamental tradeoff between energy and delay, such that, optimizing energy-efficiency inevitably will come at the expense of increasing packet delay in the network. However, low latency is an important requirement in many application scenarios, and applications have varying delay tolerances. Consider, for instance, an M2M security system which is supposed to send an alarm to a control center when an intrusion, a fire, etc., is detected. Obviously, sending the alarm packet within tolerable delay is imperative, whatever the energy cost may be. So, while

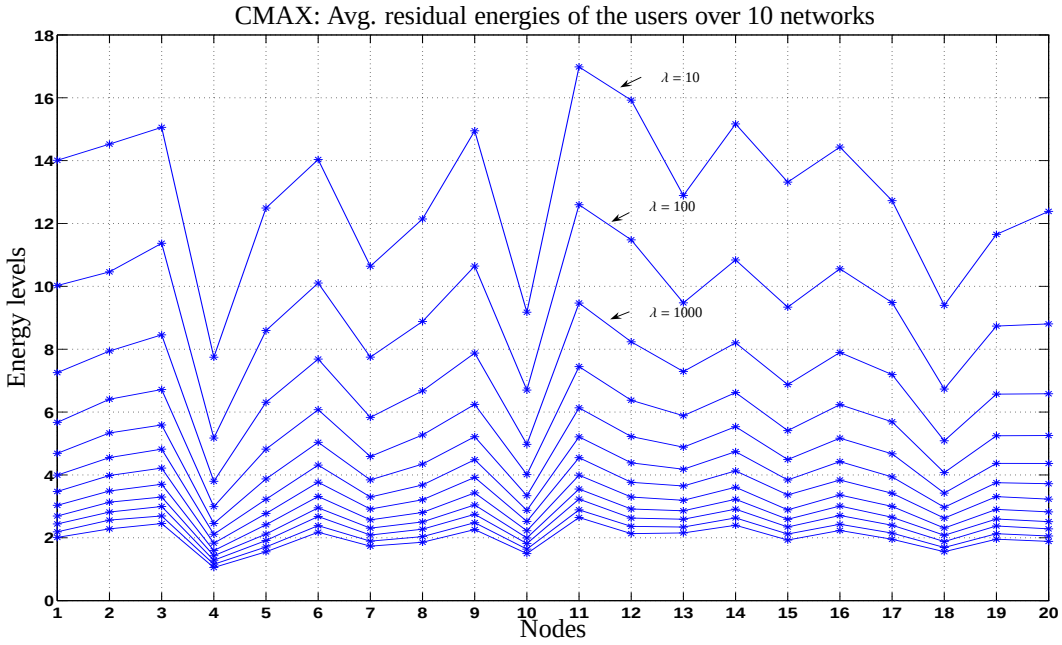

Figure 3.2: Average residual energy levels of 20 nodes over $10 \times 10 = 100$ simulations for each λ of CMAX. Each dot on the y-axis for node i represents the average remaining energy level of node i just after the first node failure, when CMAX performs routing with the corresponding λ value. For the sake of clarity, the dots corresponding to the same λ value are connected with lines.

energy-efficient operation is important for general longevity of the network, balancing this with an effort to certain delay requirements is sometimes required. Moreover, in addition to low latency, there may be other quality of service objectives as well. The rest of this section is devoted to studies that address delay and other QoS related issues within the design of energy efficient routing algorithms.

3.3.3.1 Energy-Aware Routing Algorithm for Cluster-Based Sensor Networks

The authors in [147] proposed a novel energy-aware routing approach for sensor networks. The approach (called EAR-cluster in this thesis) requires network clustering and assigns a less-energy-constrained gateway node that acts as a centralized network manager. The gateway node takes charge of sensor organization and network management based on the mission (sensing or relaying) and available energy in each sensor. Knowing the energy usage of each sensor node and which nodes need to be active in signal processing, the gateway node sets routes for sensor data, monitors latency throughout the cluster, and arbitrates medium access

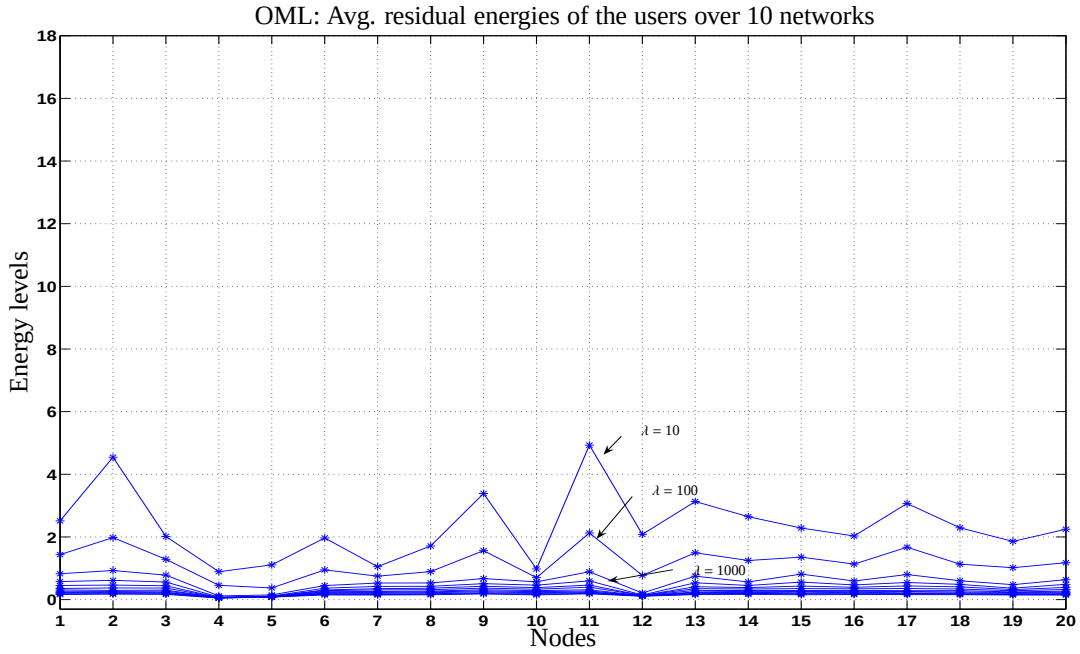

Figure 3.3: Average residual energy levels of 20 nodes over $10 \times 10 = 100$ simulations for each λ of OML. Each dot on the y-axis for node i represents the average remaining energy level of node i just after the first node failure, when OML performs routing with the corresponding λ value. For the sake of clarity, the dots corresponding to the same λ value are connected with lines.

among sensor nodes. Hence, the gateway node configures the sensors and the network to operate efficiently in order to extend the network lifetime.

Younis et al. [147] observe that the gateway node is not as energy-constrained as the sensor nodes and therefore it is better for the gateway to send commands to the sensors directly without involving relays. Hence, their problem becomes limited to routing sensor data to the gateway and thus they use a least-cost (or shortest-path) unicast routing algorithm in order to solve the routing problem. The model used for testing this algorithm assumes that nodes, sensors and gateway, are connected by bidirectional wireless links with a cost associated with each direction. The cost of a path between two nodes (source and destination) is defined as the sum of the costs of the links traversed. For each sensing-enabled node, the routing algorithm finds a least-cost path from this node (source) to the gateway node (destination). The proposed algorithm uses the following link-cost metric for the link *(i,j)*:

$$C_{ij} = \sum CF_k \quad for\ 0 \leq k \leq 7 \tag{3.27}$$

where

$$CF_0 = c_0.d_{ij}^{l},\ CF_1 = c_1.f(e_j),\ CF_2 = \frac{c_2}{T_j},\ CF_3 = c_3 \tag{3.28}$$

and

$$CF_4 = c_4,\ CF_5 = c_5,\ CF_6 = c_6.d_{ij},\ CF_7 = c_7.total\ load \tag{3.29}$$

CF_0 is the Communication cost and reflects the cost of the wireless transmission power. CF_1 represents the Energy stock and favors nodes with more energy. CF_2 is the Energy consumption rate and makes the heavily used nodes less attractive, even if they have a lot of residual energy. CF_3 is the Relay enabling cost and favors the relay-enabled nodes for routing rather than inactive nodes. CF_4 is the Sensing-state cost and favors selecting non-sensing-enabled nodes to serve as relays, since they have not committed any energy for data processing. CF_5 depends on the maximum connections per relay and extends the life of overloaded relay nodes by making them less favorable. CF_6 represents the Propagation delay and favors closer nodes. Finally, CF_7 is the Queuing Cost and does not favor relay nodes with long queues to avoid dropping or delaying data packets.

d_{ij} represents the distance between nodes i and j, l depends on the environment and $f(e_j)$ is a function of node j's residual energy. T_j is the expected time under the current consumption rate until node j's energy level hits the minimum acceptable threshold. Finally, the weighting constants c_i's are system-defined based on the current mission of the network. We refer the interested reader to [147] for more information about these weighting constants, how to use them in various cases and how to compute the total load of a network. It should be noted that some of the CF_i's factors are conflicting. For example, in order to minimize the transmission power, multiple short distances are required. However, using multiple short distances leads to more number of hops and thus the delay increases. The proposed routing algorithm provides a balance among these factors and uses Dijkstra's algorithm with the link cost C_{ij} to find the appropriate route.

In [147], the performance of the proposed algorithm was evaluated with respect to various criteria such as the time for last node to die, time till network partitions, average and standard deviation of node lifetime, average delay per packet, network throughput, and average energy consumed per packet. Based on this, it was claimed that a good balance between these performance objectives has been struck, as consistent good performance was seen with respect to both energy-based metrics, e.g. network lifetime, as well as throughput and end-to-end delay.

It was shown that although other algorithms (such as min hop routing, min. distance, Min Distance Sq.) may slightly outperform the proposed algorithm for some metrics, the same algorithms were shown to perform poorly with respect to others. However, the algorithm in [147] pays a computational price as well as an overhead, because on each link multiple cost metrics need to be calculated, stored and shared.

3.3.3.2 Energy-Aware QoS Routing Protocol

Akkaya and Younis [148] proposed an energy-aware QoS routing protocol for sensor networks. The proposed protocol extends the routing approach in [147] and finds QoS paths for real-time data with certain end-to-end delay requirements. The protocol pursues a cluster-based approach and only focuses on the QoS routing of data within one particular cluster. Hence, the main aim of the protocol is to find an optimal path to the gateway node in terms of energy consumption and error rate while meeting the end-to-end QoS requirements. The protocol operates as follows:

First, the candidate paths are calculated without considering the end-to-end delay. In order to do this, the protocol uses an extended version of Dijkstra's algorithm and finds an ascending set of least cost, delay-constrained paths for real-time data in terms of link cost. This link cost captures nodes' energy reserve, transmission energy, error rate and some other communication parameters, and is defined as follows (for a link between nodes *i* and *j*):

$$C_{ij} = \sum CF_k \quad for\ 0 \leq k \leq 3 \tag{3.30}$$

where

$$CF_0 = c_0.d_{ij}^l,\ CF_1 = c_1.f(E_j),\ CF_2 = \frac{c_2}{T_j},\ CF_3 = c_3.f(e_{ij}) \tag{3.31}$$

Here, d_{ij} represents the distance between the nodes *i* and *j* and $f(E_j)$ is the function for finding current residual energy of node *j*. T_j is the expected time under the current consumption rate for the node *j* energy level reaches the minimum acceptable threshold and $f(e_{ij})$ is the function for finding the error rate on the link between node *i* and node *j*.

Finding paths that meet the requirements for real-time traffic is not the only aim. The protocol is designed in such a way that the throughput for non-real time traffic is maximized as well.

In order to do this, a specific queueing model is used. The selected queuing model for the protocol allows the throughput for normal data not to reduce by employing a network wide r-value (related to the service rate), which guarantees certain service rate for real-time and non-real-time data on each link. Hence, once the candidate paths are obtained, these paths are checked to see which one meets the end-to-end QoS requirements by trying to find an r-value that also maximizes the throughput for non-real-time traffic. If there is no feasible r-value, the connection of that node to the gateway is simply rejected.

Simulation results show that the proposed protocol consistently performs well with respect to both energy-based metrics, e.g. average lifetime of a node, as well as contemporary metrics, e.g. throughput and average delay. Although the throughput for real-time data may decrease depending on increased number of packets, the throughput for non-real-time data does not change much since r-value ensures that there is a constant dedicated bandwidth for such data.

3.3.3.3 DAPR Protocol

DAPR (Distributed Activation based on Predetermined Routes) [111], a distributed and integrated protocol for sensor management and routing in large-scale wireless sensor networks, allows sensor nodes to become active as network coverage quality demands and to sleep whenever possible during the remainder of the time. In DAPR, time is divided into rounds. The beginning of a round consists of a Route Discovery Phase, followed by a Role Discovery Phase that is divided into an Opt In Phase (activation phase) and an Opt Out Phase (deactivation phase) [111]. Upon completion of the Role Discovery Phase (deciding whether to become active or not) sensors resume normal activity and perform routing until the beginning of the next round. The sensor nodes consider the cost to the entire network in their decision to become active, and the routes between a source node and a destination node are calculated so that minimum cumulative cost paths (shortest cost paths) are used. The proposal in [111] built on the work of [134] and [130] and developed an application based routing cost. This new metric, the application cost, aims to avoid the use of sensors in areas of critically sparse sensor deployment as routers. Unlike previously mentioned cost metrics considering only the residual energy or the energy transmitted, the application cost metric considers both the residual energy of the sensor node to whom the cost is being assigned and the residual energy

of its redundant neighboring sensor nodes. The application cost of a node is defined as:

$$C_{app}(n_j) = max \frac{1}{E_{total}(x,y)} \quad (x,y) \in C(n_j) \tag{3.32}$$

where $C(n_j)$ is the area that sensor n_j is capable of monitoring and $E_{total}(x,y)$ represents the total residual energy of the nodes which are capable of monitoring a particular location (x,y). We refer the interested reader to [111] for more detailed information about defining the regions covered by sensor nodes, calculating the application cost and obtaining $E_{total}(x,y)$ value. Application costs are assigned to sensor nodes and the cost of activating a sensor node for a given route is a weighted sum of the work that each sensor must perform:

$$C_{act}(n_s) = \sum_{n_s}^{n_d} C_{link}(n_i, n_j) \tag{3.33}$$

where n_s and n_d represents the source and destination nodes respectively and the cost of a link is calculated as follows:

$$C_{link}(n_i, n_j) = C_{app}(n_i).E_t + C_{app}(n_j).E_r \tag{3.34}$$

Here, E_t and E_r represent the energy required to transmit a packet and the energy required to receive a packet respectively. According to the simulation results reported, when compared to other cost metrics such as number of hops and the reciprocal of a node's residual energy, using the application cost metric provides the best network lifetime for DAPR. Although the performances of the second metric and the application cost is close when all the nodes have the same initial energy, the benefit from using application cost increases when the energy is not equally distributed. Furthermore, without sensor management, the effect of the increase in the number of sensors available to route the data is canceled by the effect of an increase of data generated on the network. Hence, the network lifetime is not affected much. However, with DAPR which allows sensor management, the fraction of deployed sensor nodes that are used as routers increases as the sensor nodes that are used as sensors (sensing only) decreases.

3.3.3.4 MDML Algorithm

Minimum Delay Maximum Lifetime (MDML) algorithm [149] is an energy efficient routing algorithm, specifically designed for wireless sensor networks deployed inside underground mines. MDML, is mainly a shortest path based algorithm that uses two different link costs

based on the network traffic that needs to be routed. The authors in [149] report that the WSN in mines carry two types of network traffic: Emergency traffic and Non-emergency (regular) traffic. Emergency traffic occurs occasionally and can be the result of a sudden change in the mine environmental condition, or a safety alarm sent by one of the miners. Hence, the main objective for the delivery of emergency traffic is to find a highly reliable path that incurs minimum delay. Regular traffic, on the other hand, is the result of constant monitoring and measurements and, is not delay or reliability constrained. The primary goal of routing for regular traffic is to select energy-efficient paths that can maximize the network lifetime. Hence, MDML was designed to ensure reliable and timely delivery of emergency data while maximize the lifetime of the sensor network to avoid costly redeployment of sensor nodes.

MDML algorithm makes use of a priority queue which has a classifier to check the type of the incoming packet (emergency or regular) and send it to the appropriate queue. After defining the type of the packet (or traffic), the appropriate link cost is chosen. In the case of emergency traffic, since minimizing losses and delay is more important than energy-efficiency, the following link cost, which combines the hop-count (delay), link quality, and residual energy, is used:

$$C_{ij} = ETX_{ij} * f(E_r) \tag{3.35}$$

where ETX_{ij} is the ETX (Expected Transmission Count) [150] value of the link between nodes *i*, and, *j*, and $f(E_r)$ is a function of the residual energy of node *i*. $f(E_r)$ takes on two values depending on the expected lifetime of a node; 1 if the expected lifetime is greater than the route update time (τ) and ∞ otherwise. The expected lifetime ($E(T)$) is given by [149]:

$$E(T) = \frac{E_r}{(\lambda_i - \lambda_s)e_i^r + \lambda_i e_i^t} \tag{3.36}$$

where e_i is energy consumed per bit and λ_i and λ_s are the traffic rates for self generating and outgoing traffic repectively. As a consequence, for emergency traffic, the algorithm selects the least ETX cost path among all survivable paths. As it was mentioned before, the aim is different in the regular traffic case. Therefore, the link cost metric for regular traffic mainly considers the energy-efficiency and is simply a modified version of the link cost used in FA

algorithm (eq. 3.10):

$$C_{ij} = (e_{ij}^{t})\underline{E}_{i}^{-x} + (e_{ij}^{r})\underline{E}_{j}^{-x} \tag{3.37}$$

where e_{ij}^{t} and e_{ij}^{r} are the energy consumption for unit data transmission over the link *(i,j)*, and $\underline{E}_{i}$ and $\underline{E}_{j}$ represent the residual energy levels of nodes *i* and *j*.

After defining the appropriate link cost metric, the next step is to apply the Bellman-Ford Algorithm to find the least cost path. In order to do this, each node maintains a routing table to keep the next hops for each type of the traffic (emergency and regular); and the cost to reach the destination. Then, a sink periodically (every τ second) broadcasts Sink Announcement Packets (SAP). When a node receives the SAP, it calculates the new path costs for both traffics based on the new information, it compares the new cost with the one in the routing table, and decides whether to update its routing table or discard the SAP. After updating its routing table, the node broadcasts the SAP to advertise the newly computed path to its neighboring nodes. Otherwise, it drops the packet.

The simulation results show that zero loss and low delay can be achieved for emergency traffic. MDML guarantees that delay of emergency traffic never exceeds the delay of regular traffic. The simulation results also indicate that in most cases MDML shows slightly lower network lifetime compared to the FA algorithm (called Non-MDML in [149]) which was discussed in section 3.3.1.7. Hence, MDML compromises network lifetime to achieve high reliability and low-delay.

3.3.3.5 Energy Efficient Routing with Delay Guarantee

Another study that uses Linear programming in designing an energy efficient routing algorithm was done by Coleri Ergen and Varaiya [151]. Ergen and Varaiya proposed algorithms that maximize the lifetime of a sensor network and approximate the results obtained by solving the linear programming problem. The use of linear programming and, the relation of maximizing the minimum lifetime of the nodes to minimizing the cost per packet was investigated in [138]. However, the proposed algorithms, LR-ENR and HR-ENR, take this relation one step further to provide a delay guarantee on the time the packets reach their destination, while maximizing the lifetime of a sensor network.

The proposed algorithms are developed in three steps. Firstly, an energy efficient routing protocol which aims to maximize the network lifetime is proposed for the centralized implementation of the Linear Programming (LP) solution. The protocol generates a single path from any sensor node to the Access Point (AP) at each time and attains the optimal flow rates at each link at the end. However, solving the LP problem by using the centralized protocol, requires knowing the whole network topology and packet generation rate at each node. Therefore, a distributed routing protocol [151] (which forms the basis of LR-ENR and HR-ENR protocols) is proposed to learn the optimal flow rates iteratively by using a sequence of least cost path algorithms; least sum-cost path algorithm and least max-cost path algorithm. The cost of a path for these algorithms is defined as the sum or maximum of the costs of the links on that path. Two different types of link costs are used in these algorithms:

$$C_{ij} = \frac{1}{(1 - C_j)^n} \quad or \quad C_{ij} = C_j^n \tag{3.38}$$

where C_j is the cost of node j at p^{th} iteration and is given as follows [151]:

$$C_j = p * \frac{\sum_i p_{tx} f_{ji} + \sum_i p_{rx} f_{ij} + p_s g_j + (1 - \sum_i f_{ji} - \sum_i f_{ij}) p_l}{e_j} \tag{3.39}$$

where f_{ji}, p_{tx}, p_{rx}, $p_s g_j$ and p_l represent the packet flow rate, the energy spent in the transmission of a packet in unit time, the energy spent in the reception of a packet in unit time, the energy spent in sensing and the energy spent per unit time by the radio in sleep mode respectively. In distributed routing protocol, the Bellman-Ford algorithm is used to calculate the least cost path for both of the algorithms [151].

LR-ENR and HR-ENR protocols are modified versions of the distributed protocol and, they provide a delay guarantee by limiting the length of the routing paths from each node to the AP (or sink). Level Restricted Energy Efficient Routing (LR-ENR) protocol executes a modified version of Bellman-Ford algorithm to find minimum cost paths of length at most d_{max} from each node to the sink where d_{max} is upper bounded by the worst case delay. LR-ENR protocol operates as follows: By considering the fact that the time is divided into frames, at the beginning of each frame, the sink floods the network with a tree construction packet (TCP). TCP keeps a counter c, the routing path, the node cost and the cost of the transmitting node. When a node receives a TCP, it checks whether the counter c (initially set to 0) is less than d_{max}. If $c > d_{max}$, the packet is ignored. Otherwise, the node checks whether the transmitting

node is the next hop on a path of smaller cost than previously learned paths. This is done by checking the condition $C_j > C_i + C_{ij}$ ($C_j > max(C_i, C_{ij})$) in least sum-cost (max-cost) path algorithm, in which i is the transmitting node and j is the receiving node. If the condition holds, j updates this cost by $C_j = C_i + C_{ij}$ ($C_j = max(C_i, C_{ij})$), increases the counter c by 1, adds its ID to the routing path, and rebroadcasts the packet. At the end of the flooding process, each node chooses the minimum cost routing path, and uses this path until the end of the time frame.

Hop restricted energy efficient routing (HR-ENR) protocol can achieve higher network lifetime and connectivity for a given delay constraint by using a centralized controller. The first part of the HR-ENR protocol is similar to LR-ENR's. The only difference is that each node i keeps the minimum cost path of length at most l for all $1 \leq l \leq |V|$ where $|V|$ denotes the number of nodes in the network. Then, upon reception of a TCP, similar to LR-ENR, the node checks whether the counter c is less than d_{max} where d_{max} denotes the exact worst case delay (instead of an upper bound). If $c < d_{max}$, the node checks its path cost and updates its minimum cost path for that length as described before, to rebroadcast the packet. When the flooding ends, each node knows about the minimum cost path of each length. They then send only the cost of the paths corresponding to each length l to the sink. The sink finds the optimal path length for each node based on Integer Programming model described in [151] and sends it back to the nodes in the network. Then, the nodes use the routing path of the optimal length until the end of the frame.

Simulation results indicate that the network lifetime increases significantly by optimal routing (LP formulation), and including delay constraint in energy efficient routing (LR-ENR and HR-ENR) improves the network performance since the delay of the network keeps increasing even after the optimal lifetime is achieved. The simulation results also show that for the maximum allowed delay where LR-ENR cannot provide connectivity, HR-ENR provides connectivity of all the nodes and achieves optimal network lifetime.

3.3.3.6 FMOLD algorithm

The fuzzy multiobjective routing for maximum lifetime and minimum delay (FMOLD) algorithm [152] is an extension to the FML algorithm which was previously discussed in section 3.3.1.14. For a routing request, the FMOLD algorithm finds a path that offers a good

balance between the two routing objectives, namely maximizing the network lifetime and minimizing the source-to-sink delay. Like the FML algorithm, the path search process in the FMOLD algorithm is based on the use of an edge-weight function designed by using a fuzzy membership function. FMOLD algorithm operates as follows: When a routing request is initiated, the fuzzy multiobjective membership value is computed by using the following equation:

$$\mu^{ij} = \beta.min(\mu_{lt}^{ij}, \mu_{md}^{ij}) + (1-\beta).\frac{\mu_{lt}^{ij} + \mu_{md}^{ij}}{2} \tag{3.40}$$

where μ^{ij} is the fuzzy multiobjective membership of the edge $e(v_i, v_j)$ and β is a constant (according to [152] $\beta = 0.2$ has the best effect on the maximum lifetime as well as the minimum delay objective.). μ_{lt}^{ij} represents the fuzzy lifetime and computed as described in section 3.3.1.14. μ_{md}^{ij}, on the other hand, denotes the minimum delay membership value for each edge, and, is computed as follows:

$$\mu_{md}^{ij} = 1 + \frac{(\theta - 1).pD(v_i)}{max(pD)} \tag{3.41}$$

where $pD(v_i)$ is defined as the length of the partially constructed path from the source node to the node v_i, and, $max(pD) = max_f(pD(v_f))$ for every f such that v_f is in the set of all leaf nodes of the Dijkstra's path search tree. Similar to the FML algorithm, the last step of the algorithm is to assign a weight,$w_{ij} = 1 - \mu^{ij}$, to each edge and then use the Dijkstra algorithm to find the multiobjective path between the source and destination. Simulation results presented in [152] indicate that FMOLD algorithm is able to achieve a desirable tradeoff between the maximum lifetime and the short end-to-end delay objectives. As expected FMOLD outperforms FML in terms of delay, however, it should be noted that in order to understand the delay minimization performance of FMOLD better, its performance should be compared to the other algorithms' discussed in this section.

3.3.4 Energy Efficient Routing and Scheduling

Although the main focus of this chapter, routing is not the only significant problem for energy efficient design. Due to the shared nature of the wireless channel, scheduling plays a

big role in managing interference and thus overall performance. Some studies have looked into combining routing with scheduling for an overall optimized performance. In this section, we overview several studies that combine scheduling with a shortest path based routing mechanism.

3.3.4.1 Energy Efficient Routing Using Directional Antennas

Almost all the protocols reviewed in this chapter have assumed that nodes are equipped with omni-directional antennas. That is, all nodes have a 360^o degree coverage angle and do not need to point at each other in order to communicate. The main advantage of this approach is its simplicity. However, in the case of using omnidirectional antennas, the power is broadcasted towards all directions and, therefore it attenuates rapidly with distance. Spyropoulos and Raghavendra [113] claim that a lot of energy is wasted by using omnidirectional antennas. The use of directional antennas allows nodes to communicate using less power (energy) than omnidirectional ones. Therefore, the potential energy savings that come from the use of directional antennas is significant. Hence, in [113] Spyropoulos and Raghavendra proposed an energy-efficient routing and scheduling algorithm (called EER-Directional in this thesis) that coordinates transmissions in ad hoc networks where each node has a single directional antenna. The basic idea of the proposed algorithm is to do energy efficient routing first, in order to find minimum energy paths and then schedule nodes' transmissions, accordingly.

The proposed algorithm consists of 4 major steps: In the first step of the algorithm, Shortest Cost Routing, a topology consisting all the possible links in the network is obtained by pointing the directional antenna into different directions. Then, Dijkstra's algorithm is used to find the energy efficient shortest cost paths. Two different metrics which were proposed by Singh et al. [130] are used in order to relate link/node cost with energy consumption. The first metric minimizes the energy consumed per packet by assigning the link cost as the energy consumed in transmitting (and receiving) a packet over a link. The goal of the second metric, however, is to avoid routing traffic through nodes with depleted energy. In order to do this, each node is assigned a cost (or weight) which is a function of the remaining energy of the node. Then, the total cost of sending a packet through the chosen path is minimized. This way the network lifetime is maximized. In the second step of the algorithm, Link flow matrix calculation, the amount of traffic that has to go over each link is calculated. This is done by

defining a link flow matrix whose entries are the traffic flows on the corresponding links. In the next step, Topology update, by using the end-to-end traffic information and considering the fact that only one link can be up for each node at a time, the maximum amount of time each link can be up is found. The final step is to schedule individual links in a way to minimize the total time it takes to serve all links. The scheduling problem is solved by using a series of Maximum Weighted Matchings [153].

Simulation results show that using directional antennas instead of omnidirectional ones decrease the total energy consumption and thus increase network lifetime by a factor, which is proportional to the antenna gain. Simulation results also demonstrate that additional energy savings ranging from 10 percent to 45 percent are obtained by using energy-aware routing instead of conventional routing schemes (e.g. minimum hop routing). As it can be seen from above, since this chapter is about shortest path based algorithms, we focused on the shortest cost routing part of the algorithm and gave some brief information about the other steps. We refer the interested reader to [113] for more detailed information about the other steps.

3.3.4.2 Joint Routing and Scheduling Algorithms

Girici and Ephremides studied the problem of energy-efficient distributed routing and scheduling for ad hoc wireless networks supporting connectionless traffic [116]. The authors observed the trade-offs between energy, delay and network lifetime and proposed link metric-based routing and scheduling (link activation) algorithms. The proposed link-metric based routing algorithm provides a solution for the following routing problem: Each node should select the the next hop for a packet destined to any given destination so that good communication performance is achieved in terms of energy consumed per packet, the throughput (volume of transmitted traffic throughout the network lifetime) and total delay per packet. In order to find the paths that satisfy the requirement mentioned above, each link *(i,j)* is assigned a value that indicates the cost of using that link according to a link cost metric. The combined link cost metric reflects transmission power requirements (energy), residual energy of the relaying node (volume of delivered traffic) and, congestion on that link (delay and stability), and is defined as follows:

$$C_{ij} = (\frac{P_{ij}}{P_{max}})^{W_p}(\frac{E_o}{E_i^R})^{W_e}(Q_{ij})^{W_d} \quad if E_i^R, E_j^R \neq 0 \tag{3.42}$$

where W_p, W_e and,W_d are the parameters of the algorithm that are adjusted to favor any of the three terms. Here, P_{ij} represents the RF power required for error-free transmission from node *i* to *j* and P_{max} represents the maximum RF power transmission from node *i* to *j*. The residual energy of node *i* is E_i^R. E_o is the initial energy of node *i* and finally, Q_{ij} is the congestion on the link *(i,j)*. After a link cost has been assigned for each link, based on the assigned link metrics, distributed Bellman-Ford algorithm [18] is applied for shortest path computation. The next step is to apply the scheduling algorithm. The scheduling algorithm aims to find a maximal utility link activation set (the set of links that can be activated in a conflict-free manner), *S*, such that if any node tries to activate an outgoing link with higher utility than the present activated outgoing link, the resulting activation set *S'* has lower utility than *S*. In order to do this, the communication performance is considered as a utility and each link *(i,j)* is assigned a dynamically changing utility value, W_{ij}:

$$W_{ij} = (\frac{(Q_{ij})^{\gamma}(E_i^R)^{\beta}}{(F_{ij})^{\theta}(P_{ij})^{\alpha}} \tag{3.43}$$

where α,β,γ,θ are algorithm parameters, F_{ij} indicates the number of non-empty links adjacent to link *(i,j)*, and, Q_{ij}, E_i^R, P_{ij} are as defined in eq. (3.42). The goal is to maximize the the total utility of the link activation set $U(S) = \sum W_{ij}$ [116]. The simulation results show that the routing algorithm increases the total volume of transmitted packets, and, network lifetime significantly. Also, the scheduling algorithm has a slight positive effect on the network lifetime, throughput. Based on the simulation study, it can be said that the proposed link metric-based policy jointly considering routing and scheduling, more precisely; transmission power requirements, residual energy information, link queue sizes and transceiver utilization, provides a better performance in terms of energy consumption, average delay and throughput.

3.3.4.3 Energy-Efficient Interference-Based Routing Algorithm (OptSINR)

Optimal SINR Routing (OptSINR) [154], [155] is an energy-efficient interference-based routing algorithm that, given a certain class of link scheduling schemes, aims to find the optimal routes in terms of energy consumption over the entire network. OptSINR is a cross layer routing algorithm in that it takes into account the interference created by existing flows in the

network and exploits both the SINR (physical layer information) and power control (MAC layer). The algorithm mainly deals with the problem of finding a route that minimizes the total energy increment needed to serve the new arrival over the entire network for given SINR constraints. In order to solve this problem, Kwon and Shroff [154] proposed two algorithms (in which the second one is OptSINR algorithm) for two different cases. The first case considers the impact of admitting a new flow on the network as it traverses an entire route. One or more links in the route may simultaneously transmit the flow. Therefore, a transmitting link is interfered by the other links that transport the flow over the route. This causes additional energy consumption on the network. Hence, the first routing algorithm uses SINR metrics in order to satisfy the minimum constraints of all the links and to minimize the energy consumption over the network. Since this algorithm in procedure is similar to the OptSINR algorithm, it will not be discussed in detail. We refer the interested reader to [154] for more information about the first algorithm.

As mentioned above, in the first case, the links are activated at the same time and hence, the link scheduling over the network is not considered. In the second case, on the other hand, the links are not activated at the same time but scheduled according to some link scheduling mechanism. In order to solve this new problem with scheduling and minimize the average energy consumption, OptSINR algorithm was used. The algorithm is based on some matrix operations and, for a given link scheduling matrix *S*, it operates as follows: Firstly, a directed graph consisting of *N* nodes and *L* links, *G = (N,L)*, is constructed. Then, for an incoming flow, the availability of the resources are checked. If the resources are not available, the incoming flow is rejected and the source is notified of the rejection. Otherwise, the average interference strength at all nodes is measured and, the time average of the link scheduling matrix (used to schedule the links), Π, is calculated. Then, a matrix, $(I - F)^{-1}$ is calculated based on path loss and correlation between links. In this matrix, I is the identity matrix and F is *LxL* matrix with (l,m) entry:

$$F(l,m) = \frac{G_{i(m),j(l)}c(l)}{G_{i(l),j(l)}} \quad , l \neq m \quad and\ 0\ otherwise \tag{3.44}$$

where $G_{i(m),j(l)}$ represents the path gain between transmitter *i(l)* and receiver *j(l)* and *c(l)* is the SINR constraint for link *l*. The next and may be the most important step is to calculate the link cost metric that will be used in shortest cost computation. OptSINR algorithm uses the

following cost metric for a link *l=(i,j)*:

$$C_{i(l)j(l)} = \frac{\eta_{j(l)}}{G_{i(l),j(l)}}(\Pi(I-F)^{-1})_{(\sum l)} \quad \forall; l \in L \tag{3.45}$$

where $\eta_{j(l)}$ is the average of the interference and noise measured at the receiving node, *j*, of link *l* when link *l* is active and, the term $\Pi(I-F)^{-1}$ corresponds to global information in the network. After assigning a link cost to each one of the corresponding links, a shortest path algorithm, e.g. Dijkstra's algorithm or Bellman-Ford algorithm, is applied to find the minimum cost path. Finally, the minimum cost path is used to serve the incoming flow. A nice feature of OptSINR algorithm is that it automatically routes around congested areas (the areas in which the average power of the ongoing links increases), and thus results in mitigating the overall congestion in the network. Kwon and Shroff also developed the distributed version of this algorithm which uses local information and requires a substantial reduction in computational overhead [154], [155]. The simulation results show that both centralized and distributed versions of the OptSINR algorithm are more energy efficient than other routing algorithms using the minimum transmission energy or the minimum interference as a link cost (Minimum Energy (ME), Least Interference Routing (LIR) [156]. Finally, it should be noted that, for a given class of link scheduling schemes, OptSINR algorithm is asymptotically optimal in the sense of average energy consumption.

3.3.4.4 EURo Algorithm

As mentioned in the previous sections, the three key elements of transmission power, interference, and residual energy play an important role in choosing energy-efficient routes. Ignoring one or more of these metrics, may result in algorithms which are not energy efficient in a real wireless environment, where all of these elements should be considered. Hence, Kwon and Shroff [157] developed a unified routing algorithm called the Energy-efficient Unified Routing (EURo) algorithm that parameterizes all the three key metrics: transmission power, interference between links (or routes), and residual battery energy. EURo is algorithmically similar to OptSINR algorithm [154]. However, unlike OptSINR, EURo takes into account the residual energy of nodes. Hence, in terms of energy efficiency, EURo algorithm is an improved version of OptSINR algorithm. Mainly, for a given scheduling policy S, the set of nodes, *N*, the set of links, *L*, and interference constraints, EURo algorithm tries to solve the

following optimization problem:

$$arg\ min_{R \in R(i,j)} \quad \sum_{l \in R} (C_{i(l)j(l)}) \tag{3.46}$$

$$s.t.\ \ P^{max}(l) \geq P(l) \geq 0 \ \ \forall l \in L,$$

$$\theta(l) \geq c(l) \ \ \forall l \in L,$$

$$\epsilon_n \geq 0 \ \ \forall n \in N,$$

where $P^{max}(l), \theta(l), c(l)$ and, ϵ_n represent the maximum transmission power for link l, SINR at link l, the minimum requirement of link in terms of SINR, and the remaining energy at node n respectively. In this problem, $C_{i(l)j(l)}$ is the cost metric for link *l=(i,j)* and is defined as follows:

$$C_{i(l)j(l)} = (W(I-F)^{-1})_l(\frac{\eta_{j(l)}}{G_{i(l),j(l)}}) \ \ \forall l \ \in \ L \tag{3.47}$$

where W is a weight vector that is a function of the residual energy of nodes when a new flow arrives to the network, I is the identity matrix and F is LxL matrix with *(l,m)* entry given in eq (3.44). $\eta_{j(l)}$ is the average of the interference and noise measured at the receiving node, j, of link l when link l is active and, $G_{i(l),j(l)}$ represents the path gain between transmitter *i(l)* and receiver *j(l)* and *c(l)* is the SINR constraint for link l.

Without considering scheduling, EURo algorithm using the link cost metric defined above operates as follows: Firstly, a directed graph *G = (N,L)*, is constructed. Then, for an incoming flow, each node checks the availability of two resources: battery energy and transmission power. If the battery energy is drained or the transmission power is saturated at a node, the node rejects the incoming flow and notifies the rejection to the source. Otherwise, the interference strength at all nodes are measured and, $(I-F)^{-1}$ matrix is calculated based on path loss and constraints. The next step is to calculate the present weight vector W and link cost $C_{i(l)j(l)}$. After assigning $C_{i(l)j(l)}$ as a link cost to each one of the corresponding links, a shortest path algorithm, e.g. Dijkstra's algorithm or Bellman-Ford algorithm, is applied to

find the minimum cost path. Finally, the minimum cost path is used to serve the incoming flow. It should be noted that when the links are randomly scheduled, the algorithm is similar to the one mentioned above with a small modification in the link cost [157]. The distributed version of EURo algorithm (dEURo), was also proposed in [157]. dEURo algorithm uses only local information and to update local information, each node periodically announces its status information to its neighbors via a control channel. Moreover, dEURo employs a distributed shortest path routing algorithm such as the Distributed Bellman-Ford algorithm [18].

Simulation results show that EURo algorithm outperforms other competitive energy-efficient routing algorithms such as ME(Minimum energy), CMAX [123], E-WME [124], and, LIR [156]. This is because, when there is no interference between routes, the metrics of E-WME and EURo are identical so that EURo works the same as E-WME. In addition, in the case when the arrival rate is high, due to interference between the links, the algorithms that do not consider the impact of interference (ME, CMAX, E-WME) are more affected than EURo. The simulation results also indicate that even distributed version of EURo that uses local information from adjacent neighborhoods outperforms other routing algorithms. However, it should be noted that since dEURo uses only truncated information, its performance is slightly poorer than EURo.

3.3.5 Retransmission Aware Energy Efficient Routing Algorithms

Most of the algorithms explored in this chapter are designed to use link cost metrics that are based on the energy spent in a single transmission. Banerjee and Misra [158] argues that such a formulation of the link cost fails to capture the actual energy spent in reliable packet delivery. They claim that the proper metric should include the the total effective energy (including that expended for any retransmissions necessary) spent in reliably delivering the packet to its destination and, they propose a series of retransmission aware algorithms. In this section, we describe these algorithms by focusing on their link costs and the shortest path method that they use.

3.3.5.1 Retransmission-Energy Aware Routing Algorithm

According to Banerjee and Misra, to account for the potential cost of retransmissions needed for reliable packet delivery, an accurate formulation needs to consider the link error rates as well. Hence, they proposed a shortest path based Retransmission-Energy Aware routing algorithm [158] that uses a link cost which is a function of both the link distance (related to energy) and the link error rate.

Retransmission-Energy Aware (RA) algorithm uses two different metrics designed for two distinct operating models: HHR and EER. These operating models differ in the type of retransmission mechanism that they use. HHR model allows Hop-by-Hop Retransmissions where each individual link provides reliable forwarding to the next hop using localized packet retransmissions. Hence, in this case, a transmission error on a specific link implies the need for retransmissions on that link alone. Since the number of transmissions on each link is independent of the other links, the total cost of a path is found as a linear sum of individual link costs. RA algorithm uses the following link cost for the HHR case:

$$C_{i,j} = \frac{E_{i,j}}{1 - p_{i,j}} \tag{3.48}$$

where

$$p_{i,j} = S\,p_b = 0.5S \times erfc(\sqrt{\frac{P_r}{N * f}}) \tag{3.49}$$

where $E_{i,j}$ is the energy associated with the transmission of a packet over link $l_{i,j}$, $p_{i,j}$ is the link packet error probability associated with that link, S is the packet size and, the p_b is the bit error rate for link $l_{i,j}$.

On the other hand, in End-to-End Retransmissions (EER) model, the individual links do not provide link-layer retransmissions and hence, a transmission error on any link leads to an end-to-end retransmission over the path. Therefore, the total cost of the path cannot be expressed as a linear sum of individual link costs which in turn leads to an inappropriate formulation for a minimum-cost path computation. In order to avoid this inappropriate formulation and be able to use a minimum-cost (shortest cost) routing algorithm, an approximate link cost is used. The link cost indicates the minimum approximate energy cost and is defined as:

$$C_{i,j}^{approx} = \frac{E_{i,j}}{(1 - p_{i,j})^L} \tag{3.50}$$

where L represents the average path length of the network and is same for all links. After assigning the link costs to the corresponding links, in both cases (HHR and EER), RA algorithm uses a shortest path algorithm (Dijkstra's or Bellman Ford) to determine the minimum cost path that will be used to send the incoming packet.

The EER framework results in at least an order of magnitude higher energy consumption than the HHR case. However, the energy savings achieved by the RA algorithm is more pronounced in EER case. Simulation results indicate that the RA algorithm can lead up to 30-70 percent energy savings over MH routing (minimum hop routing in which the link costs are unity), and, EA routing (Energy aware routing in which a link cost is the energy required to transmit a single packet) algorithms. The advantages of using RA algorithm is significant irrespective of whether fixed or variable transmission power is used by the nodes to transmit across links.

3.3.5.2 MRPC and CMRPC Algorithms

Misra and Banerjee proposed two power-aware algorithms for energy-efficient routing in ad-hoc wireless networks [159]. These algorithms differ from previously mentioned power-aware algorithms. The reason for this is that instead of basing their routing decisions on a function of the battery power alone, these algorithms also consider the fact that different links require different transmission powers, and also have different impacts on reliable packet transfers. The first algorithm, Maximum Residual Packet Capacity (MRPC), is conceptually similar to the MMBCR algorithm [120] which uses a max-min formulation to select the path that has the largest packet capacity at the critical node (the one with the smallest residual packet transmission capacity). However, in [158], the authors showed that a routing algorithm for reliable packet transfer should include the link's packet error probability in formulating the transmission energy cost. Hence, unlike MMBCR, MRPC uses links that have varying transmission energy costs and link error probabilities. MRPC algorithm uses the following cost metric for routing:

$$C_{i,j} = \frac{B_i}{E_{i,j}} \tag{3.51}$$

where

$$E_{i,j} = \frac{D_{i,j}^{k}}{1 - p_{i,j}} \tag{3.52}$$

Here B_i represents the residual battery power at a certain instance of time at node i and $E_{i,j}$ is the transmission energy required by node i to transmit a packet over link *(i,j)*. $D_{i,j}$ represents the distance between nodes i and j, and $p_{i,j}$ is the link's packet error probability. The maximum lifetime of a route was defined in [159] as follows:

$$Life_P = min[C_{i,j}] for\ (i, j) \in P \tag{3.53}$$

Given the cost and lifetime formulations above (eqs. 3.51 and 3.53), MRPC uses a modified version of Dijkstra's minimum cost algorithm for decentralized route computation. While $Life_P$ is not an additive function of the individual node-link costs, it is computed over a path by applying the *MIN* operator in an iterative fashion. Detailed information about the modified version of Dijkstra's minimum cost algorithm and the iterative process can be found in [159]. Simulation results show that MRPC leads to superior performance (longer network lifetimes) than alternative suggested algorithms (Min-Hop Routing, Min-Energy Routing, MMBCR and, CMMBCR) since it considers the importance of the link error rates. Moreover, in contrast to MMBCR, MRPC is not only able to transmit a much larger number of packets but also at a lower per-packet energy consumption.

Misra and Banerjee also presented CMRPC, a conditional variant of MRPC. The CMRPC algorithm is the MRPC equivalent of the CMMBCR algorithm presented in [120]. CMRPC switches from minimum energy routing to MRPC only when the remaining battery power at the constituent nodes falls below a certain threshold, γ. Simulation results show that CMRPC outperforms CMMBCR in both the total packet throughput as well as the energy efficiency. However, unlike the MMBCR case, CMRPC does not always outperform MRPC. The relative performance of MRPC and CMRPC depend on the choice of the threshold γ. While smaller values lead to higher variability in the expiration times, larger values fail to exploit minimum energy paths even if the residual battery capacities are sufficiently large. The reason for this is that when γ increases CMRPC performs minimum-energy routing for a smaller duration and

the average energy per packet increases. The total network throughput is maximized by using intermediate values for γ.

3.3.5.3 BAMER, GAMER and DAMER Algorithms

Dong et. al. [160] proposed a number of algorithms (BAMER, GAMER and DAMER) to compute minimum energy paths for reliable communication over lossy links in multi-hop wireless networks. This problem was previously studied by Banerjee and Misra [158] and, RA algorithm (which is called BMA in [160]) was proposed in order to solve the problem. However, in [158] the problem of computing optimal energy efficient paths was solved for the hop-by-hop retransmission model only. The reason for this is that for the hop-by-hop retransmission model, it is straightforward to use a traditional shortest path algorithm (e.g. Dijkstra's algorithm) to compute minimum energy paths. However, the same is not true in the end-to-end retransmission model. Therefore, Banerjee and Misra only proposed an approximate heuristic that defines the link cost $C_{i,j}$ in eq. (3.48) and used Dijkstra's algorithm to compute low-energy paths. Hence, the optimal approaches for the end-to-end case was left as an open problem.

Basic Algorithm for Minimum Energy Routing (BAMER) optimally solves the problem of computing minimum energy paths for reliable communication in the pure end-to-end retransmission model where none of the links in a wireless path guarantees any reliability. Basically, BAMER is a generalized extension of Dijkstra's shortest path algorithm which uses the following link cost (link weight):

$$W(i, j) = c\beta_o N_o d_{ij}^{\alpha} \tag{3.54}$$

where c and, α are both constants, and, β_o, N_o and, d_{ij} are the required signal-to-noise ratio (SNR), the strength of ambient noise, and, the distance between nodes i and j respectively. Although similar to the Dijkstra's algorithm, BAMER is different in that it not only takes into account the link weights, but also the link error rates. Moreover, the way of computing the total cost of a path is different in the case of BAMER. In order to clarify the difference between two algorithms (BAMER and Dijkstra), assume that for any path $P(i, j), C(P(i, j))$ denote the energy consumption of successfully delivering a packet along that path from i to j,

then, for BAMER, the total cost of a path is calculated as in eq. (3.55) whereas in Dijkstra's, the total cost of the same path is found by using the same equation and setting $N(i, j) = 1$ for all links *(i,j)*.

$$C(P(s, v)) = N(u, v)[C(P(s, u)) + W(u, v)] \tag{3.55}$$

Here, u is an intermediate node between the source node s, and the destination node v and, $N(i, j)$ is the expected number of transmissions (including retransmissions) of a successful delivery over link *(i,j)* [160]:

$$N(i, j) = \frac{1}{1 - E_r(i, j)} \tag{3.56}$$

where $E_r(i, j)$ is the link error rate of link *(i,j)*. Although BAMER is originally designed for the pure end-to-end retransmission model, an appropriate preprocessing stage which is explained in [160] enables BAMER to solve the same problem in the mixed retransmission model. The mixed retransmision model is a more general and realistic model in which some links may provide partial reliable delivery while the others may not. The General Algorithm for Minimum Energy Routing (GAMER) is proposed for that case. GAMER can be defined as a further generalization of BAMER, where each individual link may or may not provide per hop reliability. GAMER uses the same link cost (eq. 3.54) and the algorithm is the same with BAMER except that depending on the availability of per hop reliability for a link, the way of computing the total cost of a path changes. In GAMER, if link *(u,v)* provides per hop reliability, the total cost of a path is:

$$C(P(s, v)) = C(P(s, u)) + N(u, v)W(u, v) \tag{3.57}$$

Otherwise, the total cost of a path is computed as in eq.(3.55). BAMER and GAMER are both optimal and centralized algorithms. Dong et. al. also proposed a distributed algorithm, the Distributed Algorithm for Minimum Energy Routing (DAMER), that approximates the performance of the centralized algorithms. Unlike BAMER and GAMER which only compute the one-to-all shortest paths from a single source to all other nodes, DAMER computes an energy efficient path from each node to every other node. DAMER operates in a periodic round

by round fashion and for each destination *v*, DAMER chooses for (an intermediate node) *u* the next hop node that minimizes the expected energy consumption of delivering a packet from *u* to *v*.

Simulation results indicate that BAMER, GAMER and DAMER algorithms effectively improve energy efficiency over the best known existing techniques (e.g. RA [158] (or BMA as called in [160])) in the general mixed retransmission model. Furthermore, it is shown through simulations that high link error rates generally emphasize the effectiveness of the proposed algorithms. Even with the optimal setting of the algorithm parameters of BMA, BAMER and GAMER still consume less energy than BMA by up to 43 percent, and DAMER consumes less energy than BMA up to 22 percent. Moreover, DAMER is able to find minimum energy paths in the hop-by-hop model and leads to significant performance improvement over existing single-path or multi-path based techniques. Although not previously mentioned in this thesis, multi-path routing have been utilized to improve throughput or reliability, possibly at the cost of increased energy consumption. However, in some cases ([160], [161]), multi-path routing may reduce the expected energy consumption. Hence, the problem of finding the minimum energy multi-path routing is also formally analyzed in [160]. However, since this chapter is about shortest (single) path routing algorithms, energy efficiency in multi-path routing is beyond the scope of this study. We refer the interested reader to [160], (STPS, OCND) [161] and (SAR) [162] for good examples of energy efficient multi-path routing.

Up to now, we have explained many shortest path based energy efficient routing algorithms under a classification that we have done. For the sake of clarity, we illustrate this classification in Table 3.2. The table not only shows how different routing algorithms fit under different (or same) categories, but also, compares different routing techniques according to many criteria. In Table 3.2, N.S. abbreviation in SPR method column indicates that the SPR method used is not specified in algorithm (can be either Dijkstra or Bellman-Ford).

3.3.6 Concluding Remarks About the Algorithms Mentioned in This Chapter

Each of the aforementioned routing algorithms have their own functional advantages and performance limitations. Before choosing a routing algorithm to be used in an M2M application, one should first consider the requirements of that specific application. For example, in medical M2M applications, algorithms minimizing delay or maximizing QoS such as EAR-Cluster, Energy-Aware QoS, DAPR, MDML, LR-ENR, HR-ENR and FMOLD should be preferred since in case of emergency situations, the late delivery of the patient's vital information may be intolerable. Certain building management applications (e.g., Structural health monitoring), on the other hand, require the network to be available and efficient for a long time and, do not have delay as a constraint. In such a case, energy-aware routing algorithms (described in Section 3.3.1) that consider residual energy levels and seek to maximize lifetime would be more appropriate. For other monitoring applications such as Volcano monitoring [163], in addition to aiming to maximize the network lifetime, it is vital to consider the amount of retransmissions caused by the poor channel quality in the environment. Packet retransmissions may severely decrease network lifetime. So, in such an application, retransmission-aware energy efficient routing algorithms such as RA, MRPC, CMPRC, BAMER, GAMER and DAMER could be more useful. We refer the reader to Table 3.2 in which we have classified all of the algorithms considered in this chapter with respect to criteria and characteristics that we determined as significant and relevant. We intend this as a reference chart for algorithms and/or link cost metrics for different applications. In Section 3.4, we provide a brief account of link cost metrics that can be used in accordance with the discussed algorithms.

Among the many algorithms mentioned in this chapter, CMAX, OML, FML, EURo and Keep-connect algorithms are the most promising in terms of the sole objective of prolonging the lifetime of a network. CMAX and OML algorithms are among the most robust and well-known among these and often serve as benchmarks. Many recently proposed energy aware routing algorithms (such as [125], [126], [126], [157] and [164]) have been compared to CMAX and OML. FML outperforms OML and seems to be the best alternative in terms of lifetime maximization (when link error rates or SINR is not taken into account). Moreover, its two extensions; namely, multi-hop extension [143], delay extension [152] make it attractive for the research community. The EURo algorithm, on the other hand, is unique in the sense that it considers both residual energy and SINR in its link cost definition. Its optimization

Table 3.2: Comparison of energy efficient unicast routing algorithms considered in this thesis (Part I)

Algorithms		Developed for (Ad-hoc/Sensor) networks	SPR method	Centralized/ Distributed (C)/(D)	Considering Clustering	Considering Delay (or QoS)	Considering Scheduling	Retransmission Aware
Energy Aware Routing Algorithms	MMBCR	Ad-hoc	N.S.	C	No	No	No	No
	CMMBCR	Ad-hoc	N.S.	C	No	No	No	No
	Max-min *zPmin*	Ad-hoc	Dijkstra	C	No	No	Yes	No
	Zone-Based Routing	Sensor	Bellman Ford	C	Yes	No	No	No
	EAR-Low	Sensor	N.S.	C	No	No	No	No
	EAR, DEAR	Ad-hoc	Bellman Ford	D	No	No	No	No
	EERP	Sensor	Bellman Ford	C	No	No	No	No
	FA	Sensor	Bellman Ford	C	No	No	No	No
	CMAX	Ad-hoc	N.S.	C	No	No	No	No
	OML	Sensor	Dijkstra	C	No	No	No	No
	E-WME	Ad-hoc	N.S.	D	No	No	Yes	No
	SWP	Ad-hoc	Dijkstra	C	No	No	No	No
	SWCRP, SFWP	Ad-hoc	Dijkstra	C	No	No	No	No
	OLSRE	Ad-hoc	Dijkstra	C	No	No	No	No
	FML	Sensor	Dijkstra	C	No	No	No	No
	FMO	Sensor	Dijkstra	C	No	No	No	No
	MHKC	Sensor	Dijkstra	C	No	No	No	No
	MMREKC	Sensor	Dijkstra	C	No	No	No	No
	MMKC	Sensor	Dijkstra	C	No	No	No	No
	MTEKC	Sensor	Dijkstra	C	No	No	No	No
	Distributed MTEKC	Sensor	Bellman Ford	D	No	No	No	No
Delay Optimization and QoS Related Energy Efficient Routing Algorithms	EAR-Cluster	Sensor	Dijkstra	C	Yes	Yes	No	No
	Energy-Aware QoS	Sensor	Dijkstra	D	Yes	Yes	Yes	No
	DAPR	Sensor	N.S.	D	Yes	Yes	Yes	No
	MDML	Sensor	Bellman Ford	D	Yes	Yes	No	Yes
	LR-ENR, HR-ENR	Sensor	Bellman Ford	D	No	Yes	Yes	No
	FMOLD	Sensor	Dijkstra	C	No	Yes	No	No
Energy Efficient Routing and Scheduling Algorithms	EER-Directional	Ad-hoc	Dijkstra	C	No	No	Yes	No
	Joint Rout. & Sch.	Ad-hoc	Bellman Ford	D	No	Yes	Yes	No
	OptSINR	Ad-hoc	N.S.	C	No	No	Yes	No
	EURo	Ad-hoc	N.S.	C	No	No	Yes	No
	d-EURo	Ad-hoc	Bellman Ford	D	No	No	Yes	No
Retransmission Aware Energy Efficient Routing Algorithms	RA (BMA)	Ad-hoc	Dijkstra	D	No	No	No	Yes
	MRPC, CMRPC	Ad-hoc	Dijkstra	D	No	No	No	Yes
	BAMER, GAMER	Ad-hoc	Dijkstra	C	No	No	No	Yes
	DAMER	Ad-hoc	Dijkstra	D	No	No	No	Yes

Table 3.2: Comparison of energy efficient unicast routing algorithms considered in this thesis (Part II)

Algorithms		Power Control	Maximizing Lifetime as an Objective	Considering Residual Energy	Considering Residual Energy in Link Cost	Considering Transmission , Receiving Energy in Link Cost : (T),(R)	Considering link-error rates in Link Cost	Considering SINR (or SNR) in Link Cost
Energy Aware Routing Algorithms	MMBCR	No	Yes	Yes	Yes	- , -	No	No
	CMMBCR	No	Yes	Yes	Yes	T , -	No	No
	Max-min *zPmin*	No	Yes	Yes	No	T , -	No	No
	Zone-Based Routing	No	Yes	Yes	No	T , -	No	No
	EAR-Low	No	Yes	Yes	Yes	T , R	No	No
	EAR, DEAR	No	Yes	Yes	Yes	T , -	No	No
	EERP	No	Yes	Yes	Yes	T , R	No	No
	FA	No	Yes	Yes	Yes	T , R	No	No
	CMAX	No	Yes	Yes	Yes	T , -	No	No
	OML	No	Yes	Yes	Yes	T , -	No	No
	E-WME	No	Yes	Yes	Yes	T , R	No	No
	SWP	No	Yes	Yes	Yes	T , -	No	No
	SWCRP, SFWP	No	Yes	Yes	Yes	T , -	No	No
	OLSRE	No	Yes	Yes	No	T , R	No	No
	FML	No	Yes	Yes	Yes	T , -	No	No
	FMO	No	Yes	Yes	Yes	T , -	No	No
	MHKC	No	Yes	No	No	- , -	No	No
	MMREKC	No	Yes	Yes	Yes	- , -	No	No
	MMKC	No	Yes	Yes	Yes	- , -	No	No
	MTEKC	No	Yes	No	No	T , R	No	No
	Distributed MTEKC	No	Yes	No	No	T , R	No	No
Delay Optimization and QoS Related Energy Efficient Routing Algorithms	EAR-Cluster	No	Yes	Yes	Yes	T , -	No	No
	Energy-Aware QoS	No	Yes	Yes	Yes	T , -	Yes	No
	DAPR	No	Yes	Yes	Yes	T , R	No	No
	MDML	No	Yes	Yes	Yes	T , R	Yes	No
	LR-ENR, HR-ENR	No	Yes	No	No	T , R	No	No
	FMOLD	No	Yes	Yes	Yes	T , -	No	No
Energy Efficient Routing and Scheduling Algorithms	EER-Directional	No	Yes	Yes	Yes	T , R	No	No
	Joint Rout. & Sch.	No	Yes	Yes	Yes	T , -	No	No
	OptSINR	Yes	No	No	No	T , -	No	Yes
	EURo	Yes	No	Yes	Yes	T , -	No	Yes
	d-EURo	Yes	No	Yes	Yes	T , -	No	Yes
Retransmission Aware Energy Efficient Routing Algorithms	RA (BMA)	No	No	No	No	T , -	Yes	No
	MRPC, CMRPC	No	Yes	Yes	Yes	T , -	Yes	No
	BAMER, GAMER	No	No	No	No	T , -	Yes	Yes
	DAMER	No	No	No	No	T , -	Yes	Yes

based approach comes at significant complexity cost. Finally, the Keep-connect algorithm proposed in [1], called MTEKC, is interesting since, as stated by the authors, it explicitly considers network connectivity in performing the routing task. Although there is novelty in the routing, the link cost metric used in accordance with the connectivity weights is quite conventional. We believe that it could be interesting and worthwhile to use more complex link costs in accordance with the connectivity weights. This could improve performance in terms of connectivity as well as network lifetime. Finally, there are other recently reported shortest path based energy efficient solutions that have not discussed within the scope of this thesis [165], [166], [167], and, [168].

3.4 A Brief Account of Link Cost Metrics

As stated before, one of the main objectives of this study is to provide a useful reference for those who are interested in developing their own shortest path based energy efficient routing algorithms with new and more efficient metrics. Therefore, up to now, we have described many shortest path based algorithms by focusing on the link cost metric that they use. However, in literature, there are several efficient metrics which are not used in accordance with an algorithm. Hence, in this section, we present a brief discussion on some of these efficient routing (link cost) metrics; ETX [150], [169], ETT [170], MTM [171], MIC [172], mETX [173], ENT [173], Link Inefficiency [174], and, RLQ [175]. The reason why we have chosen especially these routing (or link cost) metrics is that all of them can be used in accordance with a shortest path based (or a least cost based) algorithm. Moreover, these metrics are among the well-known metrics proposed in the literature.

One of the earliest proposed metrics is the ETX (Expected Transmission count) metric which was proposed by De Couto et al. [150], [169]. ETX is defined as the expected total number of packet transmissions (including possible retransmissions) that is needed for successfully delivering a packet to the destination through a wireless link. The ETX metric incorporates the effects of link loss ratios, the asymmetry of the loss ratios in the two directions of each link, and, the interference among the successive links of a path. The ETX of a link is calculated using the forward and reverse delivery ratios of the link. The forward delivery ratio, d_f, is the measured probability that a data packet successfully arrives at the recipient (packet success probability (PSP) or packet success rate (PSR)). The reverse delivery ratio, d_r, is the

probability that the acknowledgement (ACK) packet is successfully received. These delivery ratios are measured as described in [150], [169]. The expected probability that a transmission is successfully received and acknowledged is $d_f * d_r$. A sender will retransmit a packet that is not successfully acknowledged. Hence, the expected number of transmissions, ETX of a link l, is:

$$ETX = \frac{1}{d_f * d_r} \tag{3.58}$$

The weight of a path is defined as the summation of the ETX's of all links along the path. Hence, ETX is an isotonoic metric. The isotonic property of a metric means that the metric ensures that the order of weights of two paths is preserved if they are prefixed or appended by a common third path. Isotonicity is the necessary and sufficient condition of a routing metric for the existence of efficient algorithms (Dijkstra or Bellman-Ford) to find minimum weight (or least cost) paths. Hence, ETX metric is appropriate for any minimum weight (or shortest cost path based) algorithm. Measurements on a 29-node wireless test-bed [150] show that ETX finds routes with significantly higher throughputs than a minimum hop-count metric and, it becomes more useful to use ETX as network grows larger and paths become longer. However, despite its benefits, ETX has some drawbacks. It does not consider interference and the fact that different links may have different transmission rates.

An evolution to ETX is the Expected Transmission Time (ETT) metric [170], [172]. Instead of computing the number of tries, ETT metric computes the expected MAC layer duration for a successful transmission of a packet on a particular link. This way, the metric accounts for different link transmission rates. The relationship between ETT and ETX of a link l can be expressed as:

$$ETT_l = ETX_l \frac{S}{B_l} \tag{3.59}$$

where S is the packet size and, B_l is the transmission rate of link l. Similar to ETX, the weight of a path is simply the summation of the ETT's of the links on the path. Hence, ETT is also an isotonic metric. Therefore, it can be used in accordance with any shortest path (or least cost) based algorithm. By measuring the link capacities, ETT has the main advantage of increasing the throughput of a path, and, the overall performance of the network. However, ETT has

some drawbacks. Although it overcomes the different transmission rates problem of ETX, it still does not fully capture the intra-flow (two nodes transmitting packets from the same flow) and inter-flow (among concurrent flows) interference in the network. To reduce intra-flow interference, Weighted Cumulative ETT (WCETT) was proposed by Draves et al. [170]. However, this metric is not an isotonic metric and therefore, it will not be explained in this thesis. We refer the interested reader to [170] and [172] for more information about this metric.

A similar metric to ETT, called Medium Time Metric (MTM), was independently proposed by Awerbuch et al. [171]. MTM is an additive (and isotonic) metric that allows any shortest path routing protocol to select a high throughput path. The MTM assigns a weight to each link that is proportional to the amount of medium time used by sending a packet on that link. The medium time for link l and packet p is defined as:

$$\tau(l, p) = \frac{overhead(l) + \frac{size(p)}{rate(l)}}{reliability(l)} \tag{3.60}$$

where $overhead(l)$ is the amortized average per-packet overhead of link l and, $reliability(l)$ reperesents the fraction of packets which are successfully received over link l. As it can be observed from equations (3.59) and (3.60), there is an one-to-one correspondance between the terms used in both equations. $size(p)$, $rate(l)$, $1/reliability(l)$ terms in eq. (3.60) correspond to S, B and, ETX terms in eq. (3.59) respectively. The only difference between ETT and MTM metric is that unlike ETT, MTM accounts for the MAC related overheads. Since the weight of any given path is computed by adding all MTM metrics of consecutive links on that path, shortest path protocols that use MTM find paths that minimize the total transmission time. MTM has several advantages; such as minimizing medium time consumption and thus, maximizing path capacity and, residual capacity available to other flows. The experimental results [171] show that the MTM achieves significantly higher throughput then alternative metrics (such as Min. hop and ETX).

The Metric of Interference and Channel switching (MIC) [172], is an interference-aware routing metric which improves WCETT by solving its problem of non-isotonicity and the inability to capture inter-flow interference. The inter-flow interference problem, is solved by the following definiton of the MIC. The MIC of a path with N links is defined as:

$$MIC = \frac{1}{N_n * min(ETT)} \sum_{i=1}^{N} IRU_i \; + \; \sum^{N_i=1} CSC_i \tag{3.61}$$

where N_n represents the total number of nodes in the network and, $min(ETT)$ is the minimum ETT in the network. For a given link, IRU is the aggregated channel time of neighboring nodes that transmissions on the current link consumes and, CSC represents the intra-flow interference. The non-isotonicity problem is solved by decomposing the MIC into virtual nodes [172] while applying minimum weight path finding algorithms such as Dijkstra's algorithm. Despite its advantages such as isotonicity and taking both inter-flow and intra-flow interference into account, MIC has some drawbacks. One of these drawbacks is the overhead required to maintain update information of the ETT for each link. Because, depending on traffic load, this can significantly affect the performance of the network. Another drawback is the fact that CSC captures intra-flow interference only in two consecutive links.

Besides interference, one of the most important problems of the wireless multi-hop networks is the fast link quality variation. Metrics based on average values computed on a time interval, such as ETX, may not follow the link-quality variations. To overcome this problem, Koksal and Balakrishnan [173] proposed two quality-aware routing metrics: modified ETX (mETX) and Effective Number of Transmissions (ENT). mETX proposes a more accurate model to estimate the expected transmission count (ETX) of a single packet for time-varying links. The model assumes that the bit error probability (BER) is a non-iid (independent and identically distributed) stochastic process. Under the assumption of this model, mETX works at the bit level and, computes the mean number of transmissions by summing the first two order statistics (mean and variance) of the BER over a packet duration. Specifically, mETX is expressed as:

$$mETX = exp(\mu_\Sigma + \frac{1}{2}\sigma_\Sigma^2) \tag{3.62}$$

where μ_Σ and σ_Σ^2 are the average channel BER and variability of the channel BER respectively. Similar to the ETX, mETX is an additive metric. Therefore, it can be used in accordance with a shortest path based algorithm. By combining the impact of average and variability of the loss rate, mETX provides a significant reduction in network loss rate and, improves the network performance. However, when maximizing the total throughput problem is combined

with the packet loss rate constraint, mETX may not be sufficient since the links selected by mETX may achieve the maximum link-layer throughput but incur high loss rates at the same time. ENT metric is proposed to deal with the problem of optimizing total throughput, while bounding the packet loss rate visible to higher-layer protocols. If a link causes a number of expected transmissions higher than the maximum tolerated by an upper-layer protocol (such as TCP), ENT excludes this link from the routing computation and assigns to it an ∞ metric. Otherwise, it uses the following routing metric:

$$ENT = exp(\mu_\Sigma + 2\delta\sigma_\Sigma^2) \tag{3.63}$$

As shown in both experimental and simulation results provided in[173], mETX and ENT can achieve a 50 percent reduction in the average packet loss rate as compared to ETX.

Up to now, we have mentioned many efficient metrics; ETX, ETT, MTM, MIC, mETX, and, ENT. Although these metrics can be combined with energy-efficient metrics in order to reduce the energy consumption in the network (e.g. MDML algorithm described in section 3.3.3.4), their main objectives are not energy-efficiency related. Now, we will describe two metrics that are especially designed for energy-efficiency purposes; Link Inefficiency [174], and, RLQ [175].

The Link Inefficiency metric is a link quality based cost metric which was proposed by Lal et al. [174] for energy constrained wireless sensor networks. By assuming that an ARQ protocol is employed on every link and therefore, a node will need to repeat the transmission of a packet until it is correctly received, Lal et. al. defines the following cost metric for link *i* as Link Inefficiency at time *t*:

$$I_i(t) = \frac{1}{\mathrm{PSP}_i(t)} \tag{3.64}$$

where $\mathrm{PSP}_i(t)$ is the probability that a packet will be successfully transmitted over link *i*. When a link gets worse, the packet success probability decreases and therefore, the inefficiency increases, corresponding to a larger amount of energy spent on that link due to retransmissions. Therefore, the expected energy consumption on link *i* is proportional to I_i. Hence, minimizing the sum of the Link Inefficiency metrics of all links along a path, means

minimizing the expected energy consumption along that path. It is also shown that Link Inefficiency is essentially determined by one or two simple statistics of the channel, such as deep fade probability, and probability of SNR being over a certain threshold, especially in indoor environments where channel memory is significant.

Keeping the energy expense at a minimum is directly related to fast channel measurements. However, estimating the probabilities (that will be used for computing the Link inefficiency metric) in a short time is hard due to link quality variations. Hence, Lal et. al. [174] devise a method for estimating link inefficiency metric in a short time and an energy efficient way. In order to do this, the authors first model packet transmission as a probabilistic process and then, study wireless link quality variation over a sufficiently long period of time for various link configurations. The study shows that only a few measurements of the channel are sufficient to obtain a good estimate of the Link Inefficiency metric and hence, design an efficient topology.

Another metric which is especially designed for energy efficiency purposes is the Resource-aware and Link Quality based (RLQ) metric [175]. RLQ is a combined link cost metric which is based on both the energy efficiency and the link-quality statistics. The main objective of the RLQ metric is to adapt to varying channel conditions and at the same time exploit the heterogeneous capabilities in WSANs (Wireless Sensor and Actor Networks). In WSANs, there exist two kinds of nodes; one type is the battery-powered sensor node whereas the other type is the line-powered actor node. The RLQ metric for a link between any two nodes is defined as follows [175]:

$$C_{link} = \eta_{tx}\alpha_{tx} + \eta_{rx}\alpha_{rx} \tag{3.65}$$

where α_{tx} and α_{rx} can take one of the following values depending on the type of a node: 0 for line-powered actor nodes and, 1 for battery-powered sensor nodes and, η_{tx}, η_{rx} are the normalized energy costs for the transmitter and the receiver:

$$\eta_{tx} = [(C_{tx-data} + C_{rx-ack})E_{link}]^x[1 + (1 - \frac{E_{tx-res}}{E_{tx-init}})]^y \tag{3.66}$$

$$\eta_{rx} = [(C_{rx-data} + C_{tx-ack})E_{link}]^x[1 + (1 - \frac{E_{rx-res}}{E_{rx-init}})]^y \tag{3.67}$$

where C_{tx} and C_{rx} represent the energy consumption during transmission and reception, respectively. Here, $E_{tx-init}$, $E_{rx-init}$, E_{tx-res} and, E_{rx-res} represent the initial and remaining energy levels of the transmitter and receiver, respectively. E_{link} is the expected number of transmissions and, it is computed as:

$$E_{link} = \sum_{i=0}^{K} i(1 - PRR)^{i} PRR \tag{3.68}$$

where PRR stands for the Packet Reception Rate and, K is the maximum number of retransmissions performed before ignoring the packet. As it can be seen from the formulation above, the RLQ metric not only considers both the transmission energy and the receiving energy, but also, it considers the residual (remaining) energy of a node and possible number of retransmissions. Hence, RLQ is a very efficient metric in terms of both energy efficiency and reliability. Performance evaluations, via test-bed experiments [175], show that the RLQ metric can achieve high performance in terms of PRR, network throughput and, the network lifetime.

CHAPTER 4

RESOURCE MANAGEMENT AND SCHEDULING IN WSNS POWERED BY AMBIENT ENERGY HARVESTING

This chapter investigates the state-of-the-art resource allocation and scheduling schemes / algorithms that could be used for sustainable operation of industrial WSNs.

4.1 Algorithms

4.1.1 SSEA and ASEA Schemes

In [176], the authors present both basic and advanced expectation models for solar energy harvesting. Based on these expectation models, they suggest energy allocation algorithms, SSEA (Simple Solar Energy Allocation), ASEA (Accurate Solar Energy Allocation), to achieve optimal use of harvested energy. Both algorithms operate based on time-slots. Assuming that the cycle of energy harvests has a period of T, and that T is divided into sub-periods (called slots) of equal length, the base energy harvest expectation increases during morning slots, decreases during afternoon slots, and stays nearly zero during the night. The basis of the expected harvest for each slot reflects relatively long-term tendencies such as seasonal or monthly trends. Nevertheless, short-term conditions, i.e., temporary environmental conditions are also important, especially in locations or seasons with frequent weather changes. Therefore, an "advanced" energy expectation that factors in faster weather dynamics is also computed.

Both algorithms focus on allocating energy fairly over time, leading to a more stable application performance, while at the same time maximizing utilization of the energy harvest. The

SSEA algorithm is designed for a resource-constrained sensor. It uses a basic expectation model. Thus, it is simple, and has low overhead, but it sacrifices some degree of effectiveness in energy allocation. SSEA operates as follows: (1) Determine the amount of residual energy stored in the battery, and the expected amount of energy to be harvested during each slot. (2) Using this information, find an appropriate energy budget for every slot in a harvesting cycle. (3) Then, go to sleep until the start of the next harvesting cycle. ASEA algorithm, on the other hand, is based on an advanced expectation model, and is suitable for a node which needs a more precise energy allocation and has adequate resources to support additional computation, as it comes with a higher overhead than the SSEA scheme. Based on the expectation of harvested energy, ASEA solves a linear programming problem at every start of each slot [176], and calculates the energy to be allocated for the next slot.

Both algorithms are reported to dramatically reduce the number of occasions on which a node stays in sleep mode during an entire slot [176]. When compared to the ideal scheme (which assumes that the amount of energy that will be harvested during the harvesting period is known a priori), ASEA is shown to achieve results closest to those of the ideal scheme in all respects, and SSEA comes next.

4.1.2 A Practical Flow Control Scheme

Noh and Kang [31] develop a practical flow control (called PFC in this thesis) algorithm that aims to maximize the amount of data collected by both flow-centric[1] and storage centric[2] WSNs. The algorithm is distributed and operates in a time slotted system. It cooperates with an energy allocation algorithm (called Simple Solar Energy Allocation (SSEA) in [176]) so as to use the harvested energy optimally. Under the constraints of the energy allocated to each time slot, at the start of every time-slot, the algorithm determines an appropriate flow-rate of the outgoing links, while aiming to maximize its utilization of the energy budget for this slot. The algorithm tries to maintain an ABP (Adaptive Back Pressure) super-flow as long as possible, and thus, it can be seen as a modified version of the adaptive flow control algorithm proposed in [177].

[1] When there is a sink node in the network, WSN works as a flow-centric network and aims to maximize throughput to the sink node.

[2] When the network operates without a sink, it operates as a storage-centric network with the aim of minimizing the amount of data loss due to storage constraints.

Mainly, the algorithm operates as follows: In a time-slotted system where the unit block for energy allocation is slot, it is assumed that, each slot is divided into several sub-slots; the unit blocks used for determining the transfer rate. Under this setting, each node determines the transfer rate of each of its outgoing links at the beginning of each sub-slot, and maintains this rate during the duration of a sub-slot. The transfer rate is computed by the transfer rate determination algorithm proposed in [31] which is designed to maintain the ABP superflow during a sub-slot. After the node determines the transfer rate for all outgoing links during a sub-slot, it checks whether it has enough energy to operate during that sub-slot. If it finds that there is not enough energy to sustain the node, the algorithm is terminated while making the node go into sleep mode and setting wake-up time to the start of the next slot. The flow-control scheme is shown to produce [31] the lowest amount of data loss (for the case of storage-centric WSN) as well as the highest throughput, proving that it can maximize the amount of collected data by the sink while balancing the data efficiently when the network operates in the flow-centric mode.

4.1.3 Fixed Power (FP), Minimum-Interference (MI) and Multi-Sink (MS) Power Allocation Schemes

FP, MI, and MS are simple, location-based power allocation algorithms [178] developed for structural monitoring applications with multiple sinks. Note that, all these schemes assume that energy harvesting nodes can only communicate with the sinks, not with each other. Moreover, as they do not consider energy harvesting statistics, all algorithms operate such that a data packet is sent to the sink(s) whenever sufficient energy is accumulated. FP is the simplest power allocation scheme since it assigns the same (fixed) transmit power (P) to all nodes. Tan et. al. report in [178] that, for FP, a large P permits direct communication with more sink(s) (causing multi-sink redundancy), but, not only it results in a longer harvesting period, but also introduces the near-far effect[3]. Assigning low P, on the other hand, shortens the harvesting period and reduces the level of interference, at the expense of reducing the scope for exploiting multi-sink redundancy. MS and MI schemes are designed so that powers are assigned according to nodes' proximity from the sinks, i.e., P for each node depends on its communication range and distance from the sink. MS is a multi-sink scheme, where each node is

[3] Near-far problem: At higher node densities, contention becomes more severe resulting in nodes closer to the sink becoming more favored.

assigned a power level just sufficient to communicate with its nearest j sinks. MI is the special case of MS, where j = 1, i.e., each node is able to communicate only with its nearest sink. This scheme minimizes the interference and near-far effect, while ensuring connectedness. According to the simulation results obtained for various node densities [178], FP poses a trade-off between throughput and fairness: throughput is maximized at lower powers at the expense of fairness and vice versa. The MS scheme does not perform as expected (close to MI) as the interference outweighs the potential benefits of multi-sink redundancy. Finally, by assigning the minimum P required for each node to communicate with its nearest sink, the MI scheme enables more nodes (from different locations) to have successful simultaneous transmissions, causing its superior performance in terms of throughput, data reliability and fairness.

4.1.4 QuickFix/SnapIt Algorithms

QuickFix and SnapIt [179] were proposed as two different algorithms that work in tandem, to maximize the network utility, i.e., the sum of the utility functions of the nodes, with the aim of achieving proportional fairness in a slotted-time system. The system is designed in such a way that the time during a day is broken into multiple time intervals called epochs, where each epoch consists of τ slots. Quickfix is an efficient dual decomposition and subgradient method based algorithm that operates within each epoch, to reveal the feasible region and the optimum solution differing in each epoch. It exploits the special structure of a DAG (Directed Acyclic Graph) to form an efficient control message exchange scheme, which is motivated by the general solution structure of a dynamic program. QuickFix offers a distributed solution that does not require any knowledge of the future recharging rates. Moreover, it can efficiently track instantaneous optimal sampling rates (for every slot) and routes in the presence of time-varying recharging rates. However, QuickFix's solution to the proposed utility maximization problem depends on the average (long term) energy replenishment rate of a node and not the state of the battery. Hence, if fluctuations in recharging happen at a faster time-scale than the convergence time of QuickFix, undesired battery outage and overflow scenarios may arise, causing missed samples and lost energy harvesting opportunities respectively. Therefore, Liu et. al. introduce a localized scheme called SnapIt that uses the current battery level to adapt the rate computed by QuickFix with the goal of maintaining the battery at a target level, i.e., chosen as the half of the local battery state in [179]. SnapIt chooses the rate, independently at

each node i based on the current state of the battery as follows: the rate found by QuickFix is reduced by δ_i (different for each node) if the battery is less than half full, and, is increased by the same amount when it is more than half full. We refer the interested reader to [179], for the effect of δ on the performance of QuickFix/SnapIt. In [179], QuickFix/SnapIt is compared to a modified version of IFRC (Interference-aware Fair Rate Control) [180], a backpressure-based protocol, which aims to achieve max-min fairness in WSNs. The results show that the two algorithms, working in tandem, can increase the total data rate at the sink by 42% on average when compared to IFRC, while significantly improving the network utility.

4.1.5 DRABP and NRABP Schemes

Gatzianas et. al. [181] model energy harvesting as a time-varying process and consider jointly managing the data and battery buffers (queues). The authors consider infinite data buffer and finite battery buffer sizes. They assume that the energy harvesting process is memoryless (it is claimed that, for a more general process, a slot analysis can be applied whose complexity will grow with the network size). Two policies (DRABP and NRABP) with decoupled admission control and power allocation are proposed with the goal of maximizing the total system utility (the long-term rate achieved per link) while satisfying energy and power constraints. They are carefully crafted modifications of the ABP-based policy of [182], which is known to achieve the optimal utility in the infinite battery scenario (non-rechargeable batteries). DRABP (Downlink Rechargeable Adaptive Backpressure Policy) is developed for downlink scenarios, whereas NRABP (Network Rechargeable Adaptive Backpressure Policy) is developed for multi-hop networks (ad hoc networks, sensor networks, etc.). DRABP is proven to be asymptotically optimal [181] when all nodes have sufficiently large battery capacities. Both schemes operate on virtual queues, which are constructed in such a way that any policy that stabilizes them also satisfies the appropriate long-term constraints. Since our main concern is WSNs, we focus on NRABP and refer the interested reader to [181] for details of DRABP.

NRABP operates as follows: (1) At the beginning of slot t, observe the virtual data queues and select appropriate packets for admission into the network layer, for every link, as the solution to the related problem described in [181]. (2) Observe the channel state, and, choose a power vector $P(t) = (P_1(t), ..., P_L(t))$ for each node (where $P_l(t)$ is the selected transmission

power in link l during slot t), based on the constraints on the power-related virtual queues of each link, and the result of (1). (3) Update the states of all queues according to the number of bits that have arrived and departed in this stage. It is shown in [181] that, under NRABP, all queues are bounded and thus, NRABP stabilizes any multihop network. Performance bounds on NRABP can be found in [183].

4.1.6 Duty Cycling and Power Management Algorithm

Reddy et. al. [184] develop a suboptimal duty cycling and power management algorithm (which we call DC-PM) for a single hop WSN, where K EHS (Energy Harvesting Sensor) nodes communicate with a powered destination over a wireless fading channel. The algorithm manages the power harvested at the individual nodes and duty cycle across them to avoid collisions in order to maximize the average sum data rate, subject to energy causality constraint, ECC (called energy neutrality constraint - ENC in [184]), at each node. The algorithm is build on two basic assumptions: (i) Time is slotted, with each constant channel (CC) slot of duration equal to the coherence time T_c of the channel. (ii) The harvested power at each node is assumed to remain constant for a constant power (CP) slot which contains a large number of CC slots. DC-PM consists of an inner stage (IS) of optimal duty cycling over the CC slots within each CP slot and an “outer stage" of power allocation across the constant-power slots while satisfying ECC at each of the nodes. Although suboptimal, the solutions to both stages are very simple in form and thus convenient for implementation.

The outer stage sets the short-term power constraints with the goal of maximizing the long-term expected sum data rate, subject to long-term energy causality at each node. It essentially solves the power management problem for a virtual sensor whose harvested power equals the sum harvested power across the nodes. The resulting power allocation scheme is to assign a clipped version of the sum harvested power across all the nodes, where the clipping thresholds are set to maximize the average sum throughput, subject to a sum power ECC. Hence, the average sum throughput depends only on the sum harvested power and its statistics. IS determines the duty cycles of the nodes that maximizes the average data rate (expected sum throughput) within a CP slot. It requires that the duty cycle allotted to each node be proportional to the power consumed by it in the CP slot, i.e., the duty cycle allocated to each node is the fractional allocated power of that node relative to the total allocated power. The

transmission depends on the channel gain threshold at each node, which is noted to be the same [184] at all the nodes within a CP slot. DC-PM is shown to outperform other naive schemes mentioned in [184], such as equal duty cycling with scheduling, and optimal duty cycling without scheduling.

4.1.7 MAX-UTILITY and MAX-UTILITY-D Algorithms

MAX-UTILITY [185] is an epoch-based (harvested energy is modeled as an epoch-varying function), polynomial-time, and centralized rate allocation algorithm, designed to maximize total network utility, i.e., the aggregate utility of all nodes. It is applicable to arbitrary utility functions that are concave and non-decreasing. MAX-UTILITY exploits the concavity of the chosen utility function, and a special property of tree-based networks to allocate rates to nodes as evenly as possible for achieving the main goal of utility maximization, while maintaining the minimum sensing rate required by the application and energy neutral operation for every node. The algorithm is shown to be optimal [185] in terms of assigning rates to individual nodes to maximize overall utility, while ensuring energy-neutral operation. MAX-UTILITY runs in multiple iterations, assigning rates to a subset of nodes in each iteration. The algorithm uses one global variable and three per-node variables that are updated from iteration to iteration. The global variable is a set containing all the nodes in the network that have been assigned rates so far. The per-node variables are; the remaining capacity of node *i*, the set of unassigned nodes in node *i*'s subtree, the maximum common rate for unassigned nodes in node *i*'s subtree. In each iteration, MAX-UTILITY picks a critical node (the node with the least common rate among many unassigned nodes) of the current tree, and assigns its rate to the unassigned nodes in the cirital node's subtree, then, it produces a pruned tree by removing any newly assigned nodes. MAXUTILITY stops when rates are assigned to all N nodes. The distributed version of MAX-UTILITY is also available, MAX-UTILITY-D [185]. MAX-UTILITY-D is an entirely feasible alternative to MAX-UTILITY, and allows sensor nodes to collaboratively produce optimal rate assignments. It only requires a single coordinator node such as a routing tree root, which can be any node in the network. Each iteration of MAX-UTILITY-D consists of two stages: (1) determining the minimum common rate and the critical node in the tree, by requiring all the nodes to forward their maximum common rate for the unassigned nodes in their subtree, to the root. (2) Requiring the root to disseminate the minimum common rate discovered in (1), across the network, so that all unas-

signed nodes in the subtree of the critical node can receive and use this rate as their packet rate. MAX-UTILITY is a fast and efficient algorithm that can operate with various utility functions, and has an run time of $O(N^3)$, where N is the number of nodes. When compared to an alternative heuristic called Random Rate Augmentation (RRA), proposed by Zhang et. al. [185], MAX-UTILITY is claimed to deliver superior utility improvement while ensuring energy neutral operation for all nodes.

4.1.8 NetOnline Algorithm

NetOnline is a distributed low-complexity algorithm heuristically developed for maximizing the throughput over a finite time horizon, in a sensor network with energy replenishment. The main motivation for this development [186] is the fact that, while the finite-horizon throughput optimization problem can be formulated as a convex optimization problem, its solution suffers from high complexity brought about by strong dependence of current decisions on future performance, *time coupling property*[4].

The NetOnline algorithm is comprised of two stages: (1) finding a throughput maximizing energy allocation through T slots, (2) routing. In part I, it is assumed that the energy replenishment (energy harvesting) profile can be estimated (predicted) for that period, ahead of time. Every node performs the following operations: Calculate the lower bound on the energy allocation from the lower-bound of the estimated replenishment profile, via the shortest-path solution (SPS)[5], i.e. SPS is shown to be optimal for a single node case [186], when the replenishment rate profile for the entire finite-horizon period is known in advance. Then, based on these estimations and current amount of recharging (harvesting), determine the energy to be allocated for each slot. In Part (2), the main concern is to determine the amount of data in the outgoing links of each node for the corresponding destination node in time slot t. The routing in each slot is determined by solving a simple linear programming (LP) problem. Since the defined problem is also a convex optimization problem, the authors use duality and the Lagrange multiplier method to get the optimal solution. The NetOnline algorithm is shown to be optimal under homogeneous replenishment profiles with perfect estimation for all nodes. Chen et. al. reports in [186] that, in more general settings, the algorithm significantly out-

[4] In a time-slotted system, if energy is overused in a previous period, the total throughput attainable over the time horizon will decrease as a result. On the other hand, if energy is underused in a previous period, the total throughput will also decrease, even though there is no wasted energy.

[5] The shortest path is calculated using the linear time algorithm in [187], whose complexity is O(T).

performs a state-of-the-art infinite-horizon based scheme (NRABP proposed by Gatzianas et.al. [181]), and it achieves empirical performance close to optimal.

4.1.9 The Joint Rate Control, Power Allocation and Routing Algorithm

The joint rate control, power allocation and routing algorithm [39] (called RC-PA-R in this thesis) is a resource allocation algorithm developed for multihop networks operating in a time-slotted setting, under node-exclusive interference model. The algorithm jointly controls the data queue and battery (energy) buffer to maximize the long-term average sensing rate of an energy harvesting wireless sensor network under certain QoS constraints for the data and battery queues. The resource allocation part of the algorithm consists of two components: a rate control (RC) component and a power allocation (PA) component. Both components are index policies, i.e., the solutions depend on the instantaneous values of the system variables and thus, they are memoryless. The algorithm can either be implemented in a centralized or distributed manner depending on the algorithm used for the (RC) component. For the centralized version, the classical Maximal Weighted Matching (MWM) algorithm [188] is used whereas the distributed version employs the Maximal Matching (MM) based algorithm as in [189]. RC decides on the amount of data that will be sensed, by comparing all available data with a finite tunable approximation parameter that controls the efficiency of the algorithm. Thus, the rate controller makes sure that the data queue remains within a certain bound, making a positive effect on the battery (energy level) as well, since a certain portion of the data packets are not allowed into the transmitting node. PA solves a simple convex optimization problem in each time slot to determine the powers to be allocated so that no node transfers data of a flow to a relay node that is not the destination of that flow, unless the differential backlog for that flow is greater than a fixed value, which is chosen such that the resulting backlog of the receiving node is not larger than that of the transmitting node after the transmission. Thus, the data flow is pushed from the source node to the destination with a positive back pressure. It is shown through both analysis and simulation [39] that the performance of the proposed algorithm is close to that of the optimal solution. Specifically, as V increases, the average total sensing rates of the MWM and MM based algorithm are reported to keep increasing and get closer to the optimum and a value that is much larger than half optimum, respectively.

4.2 Comparison of the Algorithms

After having described several promising candidate algorithms above for possible application in energy harvesting industrial WSNs, we shall now comparatively address the drawbacks, advantages, and possible application areas of these algorithms. As a quick referral guide, a detailed comparison of all algorithms, considered in this chapter, is provided in Table 4.1.

Table 4.1: Comparison of algorithms considered in this chapter (N.S. denotes "Not Specified")

Algorithms	Distributed / Centralized / Node level	Prediction	Type of Allocation	Battery Buffer (Finite, Infinite)	Harvesting Method
SSEA, ASEA	Node level	Yes	Energy	Finite	Solar
PFC	Distributed	Yes	Energy & Rate	Finite	Solar
FP, MI, MS	Centralized	No	Power	Infinite	N. S.
QuickFix/SnapIt	Distributed	No	Rate	Finite	Solar
DRABP, NRABP	Centralized	No	Power	Finite	N. S.
DC-PM	Centralized	No	Power	Infinite	N. S.
MAX-UTILITY	Centralized	Assumes	Rate	Finite	Solar
MAX-UTILITY-D	Distributed	Assumes	Rate	Finite	Solar
NetOnline	Distributed	Assumes	Energy	Finite	Solar
RC-PA-R with MWM	Centralized	No	Power	Finite	N. S.
RC-PA-R with MM	Distributed	No	Power	Finite	N. S.

Despite their simplistic design, FP, MI, and MS algorithms [178] operate only in a single-hop architecture, where a node can only be configured either as a source or a sink. Moreover, the algorithms require the location of each sensor node to be obtained from GPS or some other method, during deployment. The main drawback of these algorithms is that, when employed, nodes can transmit sensed data only when sufficient energy is harvested. This may cause long delays in terms of data delivery, i.e., when more energy is harvested, the packets will be sent, but, when low energy is harvested, packets will be kept waiting until required energy is accumulated. The best one of these three algorithms is known to be MI. Although not applicable for event-driven applications (e.g., detection of threats and oil spills) where data dissemination is only triggered upon the detection of abnormal phenomena, MI can be a good choice for predictive monitoring based WSN applications such as monitoring of road infrastructure, where sensed data is continuously being disseminated (e.g., periodically). The algorithm proposed by Reddy et. al. [184], DC-PM, also operates on a single hop network where a bunch of energy harvesting sensor nodes communicate with a powered destination (sink), with the goal of maximizing the sum data rate. DC-PM is the only algorithm (among the ones mentioned

in this chapter) that considers duty cycling as a part of the optimization process. Although centralized and suboptimal, DC-PM turns out to have a surprisingly simple form of power allocation and duty cycling. The algorithm is suitable for applications that require simple duty cycling and power management techniques. However, in order to implement solution, it requires the knowledge of the sum normalized power (sum harvested energy) for every slot (energy inter-arrival times form the slots) and its statistics.

In contrast to MI and DC-PM, QuickFix/SnapIt algorithms [179] can be used in well-structured networks with an underlying directed acyclic network graph (DAG). The algorithms working in tandem provide a distributed solution that does not require any knowledge of the future recharging rates. The combination (QuickFix/SnapIt) is suitable for WSN applications that demand proportional fairness and perpetual operation. Another advantage of QuickFix/SnapIt over MI is that, when solar energy harvesting is used, based on the application's minimum rate requirement, one can determine the minimum battery level that can support the minimum rate at night (when no, or, too little energy harvesting is available) and, trust on SnapIt algorithm to maintain the battery at that level to ensure the network remains active during the night time. However, although [179] target general multihop networks and offer an innovative solution, the proposed solution (QuickFix/SnapIt scheme) is not optimal, and can incur high control overhead and unpredictable running time, thus potentially limiting the practical implementation within resource-constrained WSNs. MAX-UTILITY algorithm, on the other hand, offers a time complexity of $O(N^3)$ for a system with N nodes. MAX-UTILITY-D, fully distributed version of MAX-UTILITY, allows resource-constrained sensor nodes to collaboratively produce optimal rate assignments. A common limitation of the algorithms is that they apply only to tree-based WSNs. As they require energy prediction, MAX-UTILITY and MAX-UTILITY-D algorithms are not suitable for WSNs powered by unpredictable energy sources (such as vibration). [176] and [31] also depend on energy prediction. The proposed algorithms, SSEA, ASEA [176], and PFC [31], are developed for WSNs that use solar energy harvesting. SSEA and ASEA are suitable for WSNs that require minimizing variations in the energy allocation. Note that, there is no energy to be harvested when the sun is down. However, in some industrial applications, data needs to be collected at the same rate at all times. SSEA and ASEA allow sensor nodes to reserve an adequate amount of energy to operate at a constant level at all times. Hence, the target application of SSEA and ASEA is time-driven WSNs, not the event-driven WSNs.

PFC algorithm cooperates with the SSEA energy allocation scheme to maximize long-term performance, especially the amount of data collected at the system level. It is suitable both for flow-centric and storage centric industrial WSNs. In storage centric networks, the acquired data has to be stored in the network temporarily (may be couple of days) until the sink node is connected to the network in order to gather it. Note that, this algorithm is the only one (among the ones mentioned in this chapter) considering storage centric networks. Hence, if the IWSN has the ability of solar energy harvesting, and the sink node is not usually, but only periodically, connected to the network, PFC algorithm seems to be the best choice in terms of minimizing the amount of data loss due to storage constraints. The algorithm is a good alternative for flow-centric networks (the aim is maximizing the throughput) as well, since it can operate in a distributed manner. However, the algorithm can only operate in solar-based networks and when a reasonable amount of solar data is available for prediction process (SSEA).

When prediction is not possible (or available), approaches that dynamically adapt to instantaneous energy and data buffer states are recommendable. For example, Gatzianas et al. [181] model energy harvesting as a time-varying process and consider jointly managing the data and battery buffers. The authors consider infinite data buffer and finite battery buffer sizes. They assume that the harvesting process is i.i.d, and, show that under the proposed policy, DRABP, the probability of battery state being less than the peak power or close to the full battery state vanishes as the battery size grows. NRABP, the multi-hop version of DRABP, is also shown to stabilize [181] any multi-hop network. Note that, if the data buffer size is infinite, the concern is the stability of the data queue, while for finite data buffer, excessive data losses should be avoided. A common drawback of the proposed schemes is that, although Gatzianas et. al. claim that for non i.i.d processes, a slot analysis can be applied, this operation has high complexity and depends on the network size.

In [39], Mao et. al. consider all combinations of finite and infinite data and battery buffer sizes by defining minimum number of virtual queues in a general format. In addition to the constraint on the stability of the data queue (constraint on the data loss ratio when the data buffer size is finite), they also impose a constraint on the frequency of battery discharge. Rather than assuming an i.i.d energy harvesting process as in [181], they allow for a general harvesting process without assuming ergodicity, and, consider jointly managing the data and battery buffers to deal with the coupling between them. The developed algorithm is more

advantageous over previously mentioned algorithms, as it has a built-in routing algorithm, and can be used in industrial WSNs requiring high long-term average sensing rate. However, Chen et al. argue in [186] that the infinite-horizon based solutions, such as those proposed in [181] and [39], may be highly inefficient, especially in the context of networks with energy harvesting. The stated reason is that the harvesting profiles are time varying and may not even be stationary and ergodic. Note that, the finite-horizon problem is important and challenging as well because it necessitates optimizing performance metrics that are exhibited in the short term rather than metrics that are averaged over a long period of time. One difference between the finite horizon problem in [186] and the infinite-horizon problem in [181] is, in the finite horizon problem inefficiencies cannot be made to vanish to infinitely small values. This implies that new techniques, such as NetOnline [186], need to be developed to mitigate these inefficiencies. Although no comparison of NetOnline to RC-PA-R [39] exists, NetOnline is shown to outperform the NRABP algorithm proposed in [181].

It should be noted that, depending on the requirements of the chosen WSN application (whether long-term or short term metrics are more appropriate), battery capacity, and the type of application, the relative performances of the various proposals surveyed in this chapter will be perceived differently. Depending on the type of the industrial setting, the network size and the performance criteria, we believe that one of these solutions, when appropriately tuned, will provide an efficient resource management and scheduling solution.

CHAPTER 5

PROPORTIONAL FAIR RESOURCE ALLOCATION ON AN ENERGY HARVESTING DOWNLINK - PART I: STRUCTURE

In this chapter, we pose the following problem whose objective is proportional fairness among users: How to allocate among users the transmission power and the proportion of the time between energy harvests, to achieve a good balance between throughput and fairness in an energy harvesting broadcast system. Specifically, we investigate the proportional fairness based utility maximization problem in a time-sharing multi-user additive white Gaussian noise (AWGN) broadcast channel, where the transmitter's battery gets recharged periodically (at known intervals). Energy is assumed to be harvested at the transmitter during the course of transmission (or reception). The data, on the other hand, is assumed to be ready at the transmitter before the transmission starts. We focus on finding the optimum *offline* schedule, by assuming that the energy arrival profile at the transmitter is deterministic and known ahead of time in an *offline* manner for a time window, i.e. a *frame*. The times at which harvested energy becomes available and the amounts that become are known in an *offline* fashion, at the beginning of each frame. The challenge of the optimization problem is the set of *causality* constraints introduced by the energy arrival times, *i.e.*, energy may not be used before it is harvested.

Analysis of structural characteristics of the problem reveals that it can be formulated as a biconvex optimization problem, and that it has multiple optima. Due to the biconvex nature of the problem, a Block Coordinate Descent (BCD) based optimization algorithm that converges to an optimal solution is presented in this Chapter. It should be noted that a part of this study appears in [62].

5.1 System Model

There is a single transmitter that transmits to N users by time sharing on a bandwith W. The power spectral density of the background noise is N_o. Channel conditions will be supposed to remain constant during a duration F that will be referred to as a "frame": i.e., g_n, the gain of user n, is chosen to be constant throughout the frame. The transmitter is equipped with a rechargeable battery such that harvested energy becomes available at distinct instances. With some abuse of terminology, the durations between two harvest instants will be called as "slot". The amount of energy harvested from the environment at the beginning of time slot t is E_t, and the length of the t^{th} slot is T_t as illustrated in Figure 5.1.

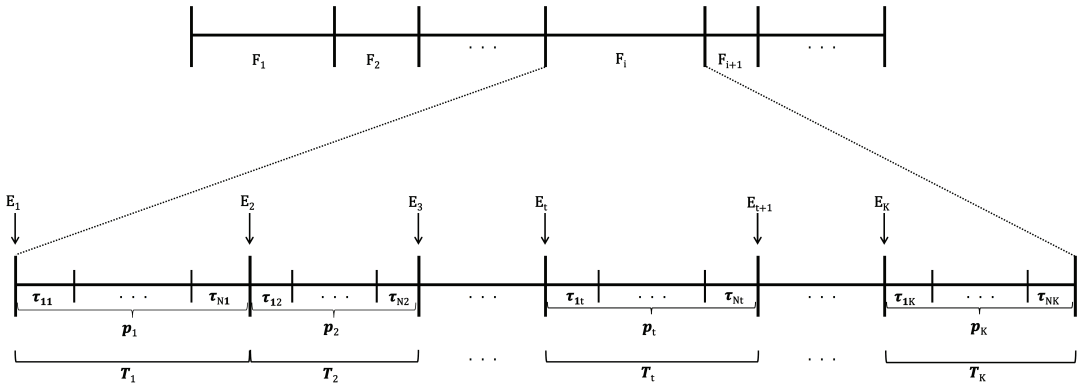

Figure 5.1: Multiple frames in a timeline. The highlighted frame, frame i, includes K energy arivals. The time between consecutive arrivals is allocated to N users.

The figure shows the details of a specific frame within a timeline. Note that, the slot lengths do not necessarily need to be equal as the energy arrivals may occur in different moments in time. We do not restrict our problem formulation to the case of periodic energy arrivals ($T_t = T$ for all $t \in \{1, \ldots, K\}$). In Chapter 6 however, we use periodic energy arrivals assumption to derive the characteristics of the optimal solution of the problem described in this chapter, Problem 1. In this *offline* problem, energy arrival times and amounts that will occur within the frame are known at the beginning of the frame. For a given frame, the transmitter chooses a power level p_t and a time allocation vector $\tau_t = (\tau_{1t}, ..., \tau_{Nt})$, for each time slot t of the frame, where $p_{nt} = p_t$ is the selected transmission power for user n during slot t and, τ_{nt} is the time allocated for transmission to user n during slot t.

5.2 Problem Statement and Structure

We define the total achievable rate for user n (the total number of bits transmitted to user n within the i_{th} frame), $R_n = \sum_{t=1}^{K} \tau_{nt} W \log_2\left(1 + \frac{p_t g_n}{N_o W}\right)$. It should be noted that, with very long transmission blocks, in terms of the fading dynamics, the ergodic nature of the fading process is revealed. Thus, we assume that the ergodic capacity is almost achieved. Our goal is to maximize a total utility, selected as the log-sum of the user rates $\sum_{n=1}^{N} log_2(R_n)$, which is known to result in proportional fairness [179]. The concept of proportional fairness and the reasoning behind the chosen utility function is explained in detail, in Appendix A. Due to the nature of the time and power allocation problem, and, energy harvesting procedure, some constraints need to be satisfied when maximizing the utility function. Accordingly, we define the following constrained optimization problem, Problem 1, where (5.1) represent the nonnegativity constraints. The set of equations in (5.2), called time constraints, ensure that the total time allocated to users does not exceed the slot length. The set of equations in (5.3), on the other hand, are technical constraints included to ensure that every user gets a non-zero time allocation during the frame. Finally, the set of equations in (5.4), called energy causality constraints, ensure no energy is consumed before becoming available.

Problem 1

$$\textit{Maximize: } U(\overline{\tau}, \overline{p}) = \sum_{n=1}^{N} \log_2\left(\sum_{t=1}^{K} \tau_{nt} W \log_2\left(1 + \frac{g_n p_t}{N_o W}\right)\right)$$

$$\textit{subject to: } \tau_{nt} \geq 0\ ,\ p_t \geq 0 \tag{5.1}$$

$$\sum_{n=1}^{N} \tau_{nt} = T_t \tag{5.2}$$

$$\sum_{t=1}^{K} \tau_{nt} \geq \epsilon \tag{5.3}$$

$$\sum_{i=1}^{t} p_i T_i \leq \sum_{i=1}^{t} E_i \tag{5.4}$$

where $t = 1, ..., K$ and $n = 1, ..., N$. Please note that, Problem 1 can be written as a minimization problem in which the function to be minimized is $-U(\overline{\tau}, \overline{p})$. Unfortunately, (1) is a nonlinear non-convex problem with potentially multiple local minima, some of which are

also globally optimum. Thus, we can only expect that by proper choice of the initial value, our algorithm converges to a stationary point that is nearby the true optimum. In order to develop such an algorithm, we first decompose the problem into two parts (power allocation, time allocation) and determine some characteristics that will be useful in understanding the problem structure better. Fortunately, these characteristics lead us to Corollary 1, which we exploit to determine the most appropriate algorithm for Problem (1).

5.2.1 Structure of the Optimal Power Allocation Problem

In this section, we assume that the time allocation, $\overline{\tau}$, is determined, and try to characterize the structure of the optimal power allocation problem for this $\overline{\tau}$. When the only variables are power variables, Problem 1 reduces to the following constrained optimization problem:

Problem 2

$$\begin{aligned} &\textit{Maximize: } U(\overline{p}) = \sum_{n=1}^{N} f_n(\overline{p}) \\ &\textit{subject to: } p_t \geq 0\ , \ \sum_{i=1}^{t} p_i T_i \leq \sum_{i=1}^{t} E_i \end{aligned} \tag{5.5}$$

where $t = 1, ..., K$ and, f_n is a function of the total number of bits sent to user n:

$$f_n(\overline{p}) = \log_2 \left(\sum_{t=1}^{K} \tau_{nt} R_{nt} \right) \tag{5.6}$$

and R_{nt} represents the rate of link n in the t^{th} slot:

$$R_{nt} = W log_2 (1 + L_n p_t) \ \text{ where } \ L_n = \frac{g_n}{N_o W} \tag{5.7}$$

Lemma 5.2.1 will be useful to get a handle on the characteristics of the problem. Although we claim no originality for the results of the lemma, we provide a proof for completeness.

Lemma 5.2.1 *i) Let $h_1, \ldots, h_K$ be strictly concave functions of $p_1, \ldots, p_K$ respectively, and, $c_1, \ldots, c_K \geq 0$. Then, $l = \sum_{i=1}^{K} c_i h_i$ is concave. If one of the c_i's is positive (> 0), then l is strictly concave.*

ii) Increasing concave functions of strictly concave functions are strictly concave.

Proof. The proof is provided in Appendix B.1. ■

Theorem 5.2.2 *Problem 2 can be formulated as a strictly convex optimization problem. Thus, there exists only one global optimum for a given time allocation.*

Proof. The proof is provided in Appendix B.2. ■

5.2.2 Structure of the Optimal Time Allocation Problem

In this section, we assume that the power allocation across all slots has been determined. Then, given that the power variables are known constants, we determine the characteristics of the time allocation. So Problem 1 reduces to Problem 3, where the only variables are the time variables:

Problem 3

$$\text{Maximize: } U(\overline{\tau}) = \sum_{n=1}^{N} s_n(\overline{\tau})$$

$$\text{subject to: } \tau_{nt} \geq 0\ ,\ \sum_{n=1}^{N} \tau_{nt} = T_t\ ,\ \sum_{t=1}^{K} \tau_{nt} \geq \epsilon \tag{5.8}$$

where $t = 1, ..., K$, $n = 1, ..., N$ and, s_n is a function of the time variables:

$$s_n(\overline{\tau}) = \log_2\left(\sum_{t=1}^{K} \tau_{nt} R_{nt}\right) \tag{5.9}$$

and R_{nt}'s (defined in Eq. 5.7) are known constants that represent the rate of link n in the t^{th} slot.

Theorem 5.2.4 below records the convexity of Problem 3, and leads to one of the main results of this thesis, Corollary 1. The proof of Theorem 5.2.4 rests on the observation in Lemma 5.2.3. Although we claim no originality for the results of the lemma, we provide the details for completeness.

Lemma 5.2.3 *i) Let $q_1, \ldots, q_K$ be affine functions of $\tau_{n1}, \ldots, \tau_{nK}$ respectively, and $d_1, \ldots, d_K \geq 0$. Then, $m = \sum_{i=1}^{K} d_i q_i$ is affine.*

ii) Increasing concave functions of affine functions are concave.

Proof. The proof is provided in Appendix B.3. ■

Theorem 5.2.4 *Problem 3 can be formulated as a convex optimization problem. Thus, all local optima are global optima.*

Proof. The proof is provided in Appendix B.4. ■

Note that, Problem 3 is convex, but not necessarily *strictly* convex. Therefore, in general, rather than a unique global optimum, there may be multiple local optima which are all also globally optimum.

Corollary 1 *Problem 1 can be formulated as a biconvex optimization problem.*

Proof. The proof is provided in Appendix B.5. ■

5.3 Solution Method

In the previous section, we have shown that Problem 1 can be formulated as a biconvex optimization problem since $-U(\overline{\tau}, \overline{p})$ is a biconvex function. Such functions are well-studied in the optimization literature [190], [58]. While not convex, they admit efficient coordinate descent algorithms that solve a convex program at each step. In this section, we present a block coordinate descent based algorithm, shortly BCD, for solving Problem 1. In the BCD solution method, sequentially one block of variables is minimized under corresponding constraints while the remaining blocks are fixed. We have the simplest case of only two block variables $\overline{\tau}$ and $\overline{p}$. Hence, the algorithm alternates between minimization with respect to $\overline{\tau}$ and minimization with respect to $\overline{p}$. Our BCD algorithm operates explicitly as follows:

1. Start from any valid time allocation, for example assign each time slot to different user in the form of TDMA. Assuming that all of the energy E_t is used up until the end of period t, the power is determined. This power setting satisfies Eq. (5.4).

2. Keep τ_{nt} fixed for all n and t. Optimize $U(\overline{\tau}, \overline{p})$ with respect to $p_t, t = 1, \ldots, K$ and the constraints given by (5.4).

3. Repeat the following for all $t = 1, \ldots, K$.: Keep τ_{ni} fixed for all $n = 1, \ldots, N$ and $i \neq t$. Also keep p_t fixed for all t. Maximize $U(\overline{\tau}, \overline{p})$ with respect to $\tau_{nt}, n = 1, \ldots, N$ and constraint in Eq. (5.2).

4. If the variables have converged, stop. Otherwise, go to Step 2.

For optimization of the time variables, the Lagrange multiplier method is used. The optimization of the power variables, however, is accomplished by using the Sequential Unconstrained Minimization Technique (SUMT) [191]. SUMT is an optimization method that converts a constrained optimization problem into an unconstrained one by adding the constraints to the objective function as a "penalty". It then uses a standard unconstrained optimization algorithm (*e.g.*, Newton, Steepest Descent, etc.) [192], [193] to solve the problem with the new objective function.

Regarding the issue of convergence, Problem 1 is a biconvex optimization problem and as such potentially, there exist many local optima. Therefore, convergence to the global optimum is not guaranteed. However, provided that some conditions are satisfied, convergence to a partial optimum (see Definition 5.3.1) is guaranteed. As also discussed by Lin in [194], convergence to a stationary (or critical) point, for block coordinate descent methods requires sub-problems to have unique solutions ([195], [144]), but this property does not hold here: Although sub-problem 2 is strictly convex, 3 is not strictly convex (only convex). Fortunately, for the case of two blocks, Grippo and Sciandrone [196] have shown that this uniqueness condition is not needed. Hence, BCD converges to a stationary point of Problem 1. As a stationary point can be minimum, maximum, or a saddle point, this convergence result may not be sufficient. However, we can still use the following definition and theorem (Definition 4.1 and Theorem 4.2 of [190], respectively) to build a stronger result. For this, let $X \subseteq \mathfrak{R}^n$ and $Y \subseteq \mathfrak{R}^m$ be two nonempty sets, let $B \subseteq X \times Y$, and, let B_x and B_y denote the x-sections and y-sections of B, respectively.

Definition 5.3.1 *Let $f : B \to \mathfrak{R}$ be a given function and let $(x^*, y^*) \in B$. Then, (x^*, y^*) is called a partial optimum of f on B, if*

$$f(x^*, y^*) \leq f(x, y^*) \ \forall x \in B_{y^*} \quad \textit{and} \quad f(x^*, y^*) \leq f(x^*, y) \ \forall y \in B_{x^*} \tag{5.10}$$

Theorem 5.3.2 *Let B be a biconvex set and let $f : B \rightarrow \Re$, be a differentiable, biconvex function. Then, each stationary point of f is a partial optimum.*

Hence, we conclude that the BCD algorithm surely converges to a partial optimum of Problem 1. Furthermore, Theorem 4.9 of [190] shows that, when subproblems are solvable, for BCD-like algorithms[1] (There are only two block of variables, and, sequentially one block of variables is minimized under corresponding constraints and the other block is fixed), if the sequence generated by the algorithm is contained in a compact set, then the sequence has at least one accumulation point. The theorem further states that; when one of the subproblems is strictly convex, all accumulation points are partial optima, and have the same function value (Note that while a global optimum is a partial optimum by definition, it may not be an accumulation point. In that case, all the partial optima that are in the set of accumulation points have strictly lower values than optimum function value.) Hence, we conclude that the BCD algorithm surely converges to an accumulation point, which is also partial optimum, of Problem 1, and all accumulation points (a set of partial optima) yield the same utility value. Note that although the final allocation, $(\overline{\tau^*}, \overline{p^*})$ generated by the BCD algorithm, might be a partial optimum, it neither has to be a global nor a local optimum to the given biconvex optimization problem. Because, although the set of accumulation points BCD converges to are partial optima and have the same function value, there may be other partial optima that may have different function values. Depending on the starting point of the algorithm, BCD may converge to a set that includes the global optimum, or a different set that includes local optima, or just partial optima. According to [190], there exists a theorem, originally developed by Wendell and Hurter [197], that describes the connection between partial and local optima for the following biconvex minimization problem,

$$min\{f(x,y) : \ x \in X \subseteq \Re^n, \ y \in Y \subseteq \Re^m\} \tag{5.11}$$

However, as also noted in [190], the given local optimality condition is in general not sufficient. Indeed, Wiesemann claims in [198] (p. 92) that, even the verification whether a particular solution to a biconvex problem is locally optimal is *NP*-complete. Gorski et.al. [190], on the other hand, claims that to find the global optimum of a biconvex minimization problem by a BCD-like algorithm (ACS [190]), a multistart version of BCD can be used. But, still, there is no guarantee to find the global optimum within a reasonable amount of time or to be

[1] ACS (Alternate Convex Search) algorithm proposed in [190].

sure that the actual best minimum is the global one. Hence, it seems justified to settle for the modest goal to find a partial optimum in our case.

5.4 Numerical and Simulation Results

In this section, we present the numerical and simulation results related to BCD algorithm. Throughout our simulations we use the following setup: $W = 1kHz$, $N_o = 10^{-6}W/Hz$. Unless otherwise stated, all powers are in Watts and all energies are in Joules. For the sake of an example, we suppose that there are five users in the system and 10 energy arrivals in 100 secs (frame length). The arrivals are $\overline{E} = [20, 100, 1, 1, 1, 70, 100, 1, 10, 40]$ joules in the $[1^{st}, 2^{nd}, \ldots, 10^{th}]$ slots respectively. The first user is the strongest one, and, other users are ordered in a such way that the preceding user is twice as strong as the previous one, i.e., path losses of the users are; 25, 28, 31, 34, 37 dB respectively. The starting point of the algorithm is the "Spend What You Get" policy (proposed by Gorlatova et. al. [199]) combined with TDMA time allocation. This policy corresponds to using all energy in the epoch it was harvested in, and will be referred to in the rest as SG+TDMA. We performed simulations both for unequal and equal slot lengths. In our simulations, we use the following sequence of slot sizes for 10 slots; $S_1 = [10, 12, 5, 7, 4, 15, 20, 2, 10, 15]$ and $S_2 = [25, 44, 14, 7, 3, 32, 47, 19, 26, 38]$, for the case of unequal slot lengths, and, $\widetilde{S_1} = [10, 10, \ldots, 10]$ and $\widetilde{S_2} = [25.5, 25.5, \ldots, 25.5]$ for the case of equal slot lengths. Note that, S_1 and $\widetilde{S_1}$ have the frame length of 100 secs, whereas S_2 and $\widetilde{S_2}$ have the frame length of 255 secs.

First, we assume that the frame length is 100 secs, and, we illustrate the power iterations of the BCD algorithm, for S_1, in Figure 5.2. The power convergence of the algorithm for periodic energy arrivals ($\widetilde{S_1}$), however, is illustrated in Figure 5.3. As observed from the figures, rather than transmitting with full power, saving some energy for the future use is preferred. Another observation is about the fast convergence of the algorithm, i.e., the powers seem to rarely change after just a few iterations. In order to observe the effect of frame length and different slot lengths, in Figure 5.4, we show how utility improves through the iterations for all of the aforementioned slot length sequences, i.e., S_1, $\widetilde{S_1}$, S_2, $\widetilde{S_2}$. The fast convergence of the BCD algorithm is more evident in this figure. The optimal schedules (power and time), optimal utility and thus, the utility improvement (when compared to SG+TDMA) obtained by BCD, for all four sequences are presented in Table 5.1. The reason for comparing the proposed

algorithm with SG+TDMA can be explained as follows: In most of the technical papers, proposed algorithms are compared to some other previously proposed algorithms. Since what we have done in this thesis is new, earlier work related to our subject, "proportional fair power and time allocation on an energy harvesting broadcast system", does not exist. Hence, there were no algorithms to compare to our proposed algorithms. Therefore, for the power allocation, we have chosen the SG (Spend what you Get) algorithm proposed by Gorlatova et. al. [199], since our algorithm considers spreading energy through time, but SG does not. And, for the time allocation, we have chosen basic time division, the most simple and common approach for dividing the channel.

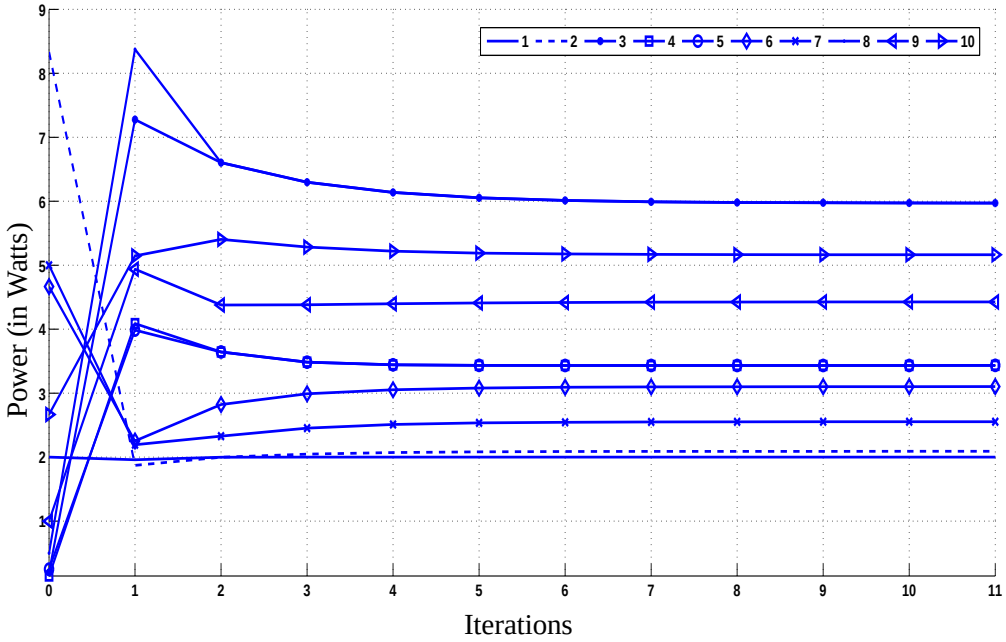

Figure 5.2: Powers vs. iterations (N=5, K=10, Unequal slot lengths): The numbers in the legend represent the corresponding slots. Starting from SG policy, BCD converges to the optimal powers in 11 iterations.

In some energy harvesting systems, transmitters have supercapacitors that can store the harvested energy and supply in every predetermined time window, allowing the case of periodic energy arrivals. In such a case, if no energy is harvested within a slot, we set the amount of harvested energy to 0 for that slot. As observed from Figure 5.4 and Table 5.1, periodic energy arrivals assumption does not degrade the system performance. Moreover, as we have shown in Chapter 6, by using the periodic energy arrivals assumption we can analytically derive the characteristics of the optimal solution of Problem 1 and, develop two heuristics that closely

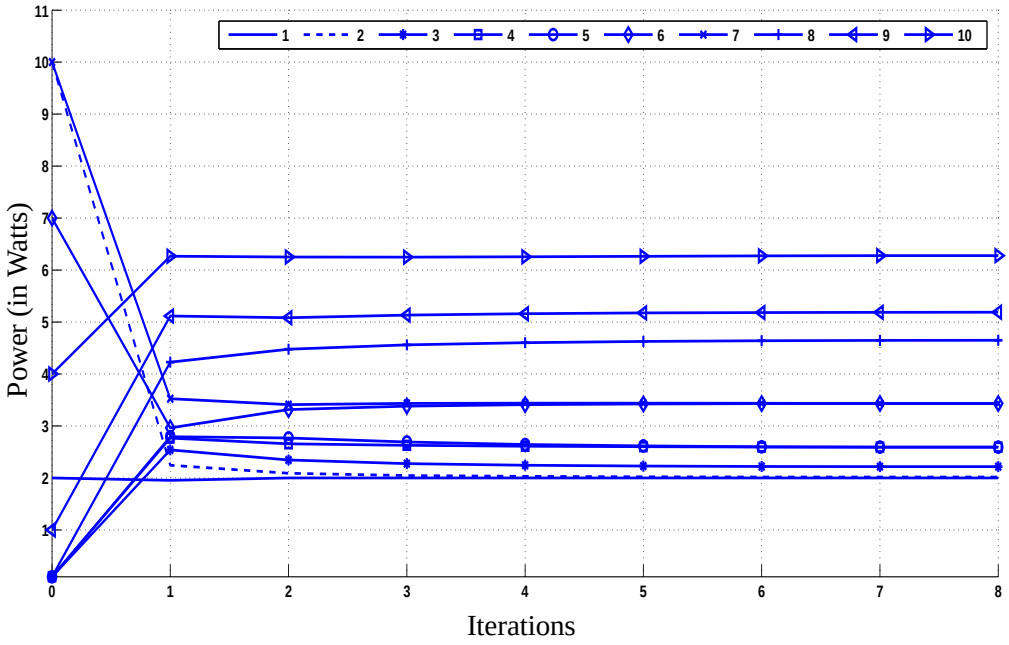

Figure 5.3: Powers vs. iterations (N=5, K=10, Equal slot lengths): The numbers in the legend represent the corresponding slots. Starting from SG policy, BCD converges to the optimal powers in 8 iterations.

track the performance of BCD algorithm. Hence, from now on, we present results only for the case of periodic energy arrivals in 100 secs, $\widetilde{S_1}$.

Throughput improvement is another important criteria in our problem setup. Hence, we next investigate the throughput improvement of the users for increasing path losses. The results are illustrated in Figure 5.5. In the figure, the Mean Path Loss, is computed as $\widetilde{L} = \frac{1}{N}\sum_{i=1}^{N} L_i$ where L_i represents the path loss of user i. As seen from the figure, with minor decrease in the throughput improvement of the stronger users, the weak users receive much more bits than that they used to receive with SG policy and TDMA. As $\widetilde{L}$ increases, the overall throughput improvement also increases. For instance, when $\widetilde{L} = 31$, User 1, User 3, User 4, and, User 5 enjoy approximately 3 %, 1621 %, 361 %, 80 % throughput improvement respectively, while User 1 suffers only 32 % of loss. Clearly, BCD is a proportionally fair algorithm which tries to maximize the utility by meeting certain demands of every user.

We next analyze the effect of number and amount of energy arrivals. Assume that there are six users in the system with the following path losses: 19, 22, 25, 38, 31 dB. The results obtained for six different energy arrival sequences (two for K=10 and four for K=12) are

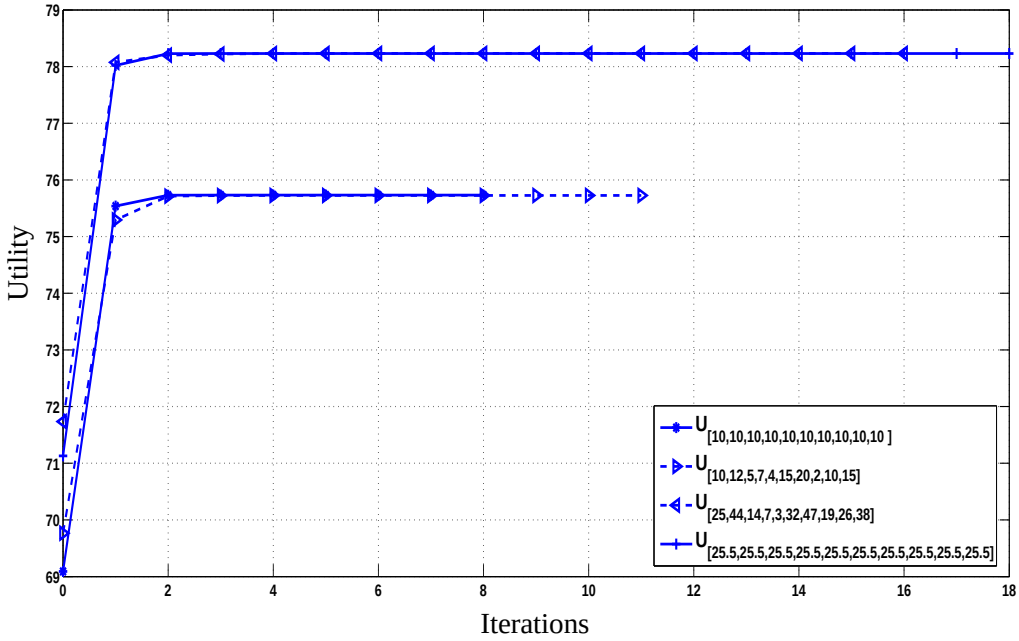

Figure 5.4: Utility vs. iterations (N=5, K=10): Starting from SG+TDMA, BCD converges to the optimal utility in 11,8,16,18 iterations for the following slot length sequences respectively: $S_1, \widetilde{S_1}, S_2, \widetilde{S_2}$.

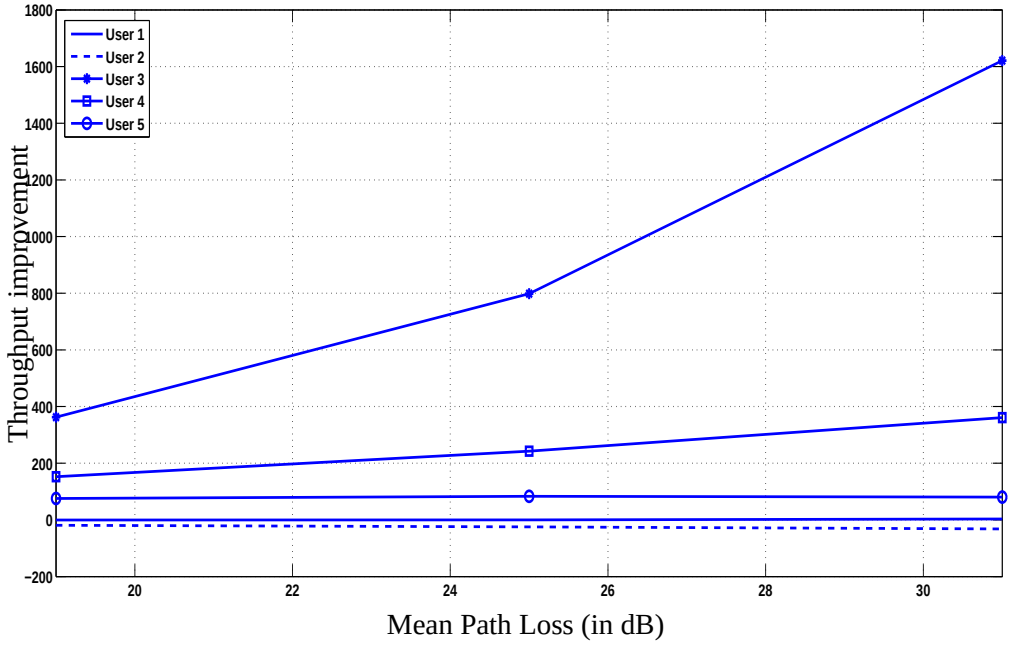

Figure 5.5: Throughput improvement vs. mean path loss (N=5, K=10): Mean path loss is computed as the mean of the path losses of all users in the system. Results represent the throughput improvement of five users for three different path loss patterns. With minor decrease in the throughput of the stronger users, the weak users receive much more bits than that they used to receive with SG+TDMA.

Table 5.1: The results of BCD algorithm for four different slot length sequences

		Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7	Slot 8	Slot 9	Slot 10	Utility	Utility Imp. (%)
Time Allocation	Users vs. Slot Lengths	10	12	5	7	4	15	20	2	10	15		
	1	10	12	0	0	0	0	3.128	0	0	0	75.727	8.544
	2	0	0	0	0	0	5.763	16.871	0	0	0		
	3	0	0	0	7	4	9.236	0	0	0.198	0		
	4	0	0	0	0	0	0	0	0	9.801	6.920		
	5	0	0	5	0	0	0	0	2	0	8.079		
Power Allocation		2	2.091	5.972	3.433	3.433	3.102	2.553	5.972	4.426	5.163		
Time Allocation	Users vs. Slot Lengths	10	10	10	10	10	10	10	10	10	10		
	1	10	10	6.233	0	0	0	0	0	0	0	75.732	9.613
	2	0	0	3.766	10	10	0	0	0	0	0		
	3	0	0	0	0	0	10	10	0	0	0		
	4	0	0	0	0	0	0	0	10	6.309	0		
	5	0	0	0	0	0	0	0	0	3.960	10		
Power Allocation		2	2.018	2.218	2.592	2.592	3.432	3.432	4.648	5.187	6.277		
Time Allocation	Users vs. Slot Lengths	25	44	14	7	3	32	47	19	26	38		
	1	25	44	0	0	0	0	0.961	0	0	0	78.233	9.056
	2	0	0	0	0	0	13.416	46.038	0	0	0		
	3	0	0	14	0	0	18.583	0	19	0	0		
	4	0	0	0	0	3	0	0	0	0	38		
	5	0	0	0	7	0	0	0	0	26	0		
Power Allocation		0.795	0.795	1.386	2.449	1.839	1.212	1.027	1.386	2.449	1.839		
Time Allocation	Users vs. Slot Lengths	25.5	25.5	25.5	25.5	25.5	25.5	25.5	25.5	25.5	25.5		
	1	25.5	20.899	0	0	25.5	0	0	0	0	0	78.231	9.983
	2	0	4.600	25.5	25.5	0	4.230	0	0	0	0		
	3	0	0	0	0	0	21.269	25.5	3.441	0	0		
	4	0	0	0	0	0	0	0	22.058	18.595	0		
	5	0	0	0	0	0	0	0	0	6.904	25.5		
Power Allocation		0.775	0.815	1.047	1.047	0.775	1.314	1.384	1.759	1.985	2.586		

shown in Table 5.2. The arrival sequences are intentionally chosen similar to each other, so that it would be easier to determine the effect of small changes on the utility improvement. As observed, it is not the number of slots (number of energy arrivals) but the amount of every individual energy harvest that determines the utility improvement. The events like; sudden decrease in energy level or harvesting very small amount of energy for a long time increases the utility improvement obtained by BCD, as SG+TDMA policy may cause the base station to stay idle for a long time because it does not save energy for future use.

Table 5.2: The effect of number and amount of energy harvests

Number of Harvests	Harvests												Utility Improvement (%)
10	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7	Slot 8	Slot 9	Slot 10			
	20	100	10	60	10	70	100	10	10	40			2.711
	20	100	1	1	1	70	100	1	10	40			7.846
12	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7	Slot 8	Slot 9	Slot 10	Slot 11	Slot 12	
	20	60	100	1	1	1	70	85	100	1	10	40	7.256
	20	60	100	0.5	1	0.5	70	85	100	0.5	10	40	8.430
	20	60	100	0.5	50	0.5	70	85	100	0.5	10	40	6.713
	20	60	100	1	0.5	0.5	1	1	100	0.5	10	40	8.515

Knowing that it is not the number but the nature of harvests that affect the utility improvement, from now on, we set the number of energy arrivals to be 10 (K=10) and the harvests to arrive as in the 2^{nd} arrival sequence, i.e. [20, 100, 1, 1, 1, 70, 100, 1, 10, 40] so that we can analyze the effect of number of users to the performance of the BCD algorithm. Keeping the number of harvests and harvest values the same, we perform a series of simulations with different number of users. First, the effects of the optimal power-time allocation pairs on utility, utility improvement, and fairness are investigated. In order to be able to analyze all scenarios, in the next three figures, we use the following setups: a) The strongest user in the system has 13 dB path loss, and, every new user that joins the system has 3 dB more path loss than the previous one. b) The strongest user has 19 dB path loss, and, every new user deviates 3 dB. c) The strongest user has 25 dB path loss, and, every new user deviates 3 dB. Hence, 13 dB, 19 dB, and, 25 dB seen in the figures represent the path loss of the strongest user in the system. Figures 5.6 and 5.7 show how utility and percentage improvement in utility, respectively, change with the increasing number of users. As seen from the figures, the solution found by BCD exhibits significant improvement over a SG+TDMA schedule. Between two methods, SG+TDMA is the worst one since even with a few users, utility can be improved. The results

show that when case c) is valid, a utility improvement of approximately 20% is possible with BCD.

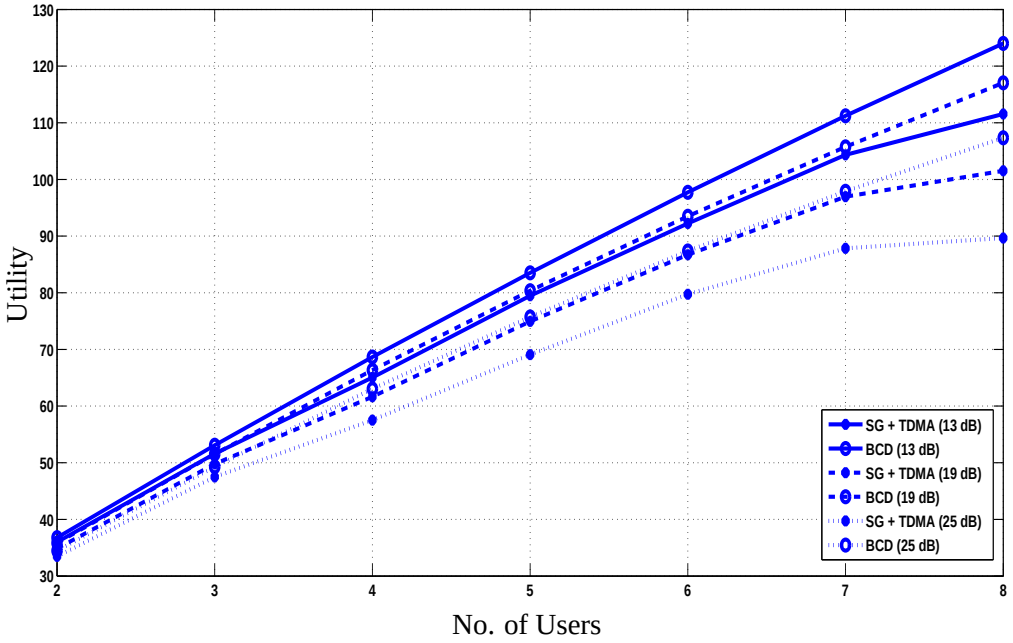

Figure 5.6: Utility (SG+TDMA, BCD) vs. no. of users: The utilities obtained by SG+TDMA and the proposed algorithm, for increasing number of users, are compared. The effect of path loss (the strongest user's path loss is shown between parentheses) on utility is shown. As path losses of the users increase the utility decreases.

Although we aim at proportional fairness in this thesis, it may be interesting to analyse max-min fairness of the BCD algorithm. Jain's index [200], [201] is a well-known measure of fairness. The index FI takes the value of 1 when there is a complete fair allocation.

$$FI = \frac{(\sum_{i=1}^{N} x_i)^2}{N \cdot \sum_{i=1}^{N} x_i^2} \tag{5.12}$$

For computing FI, we use the no. of bits transmitted to the users, $x_i = \sum_{t=1}^{K} \tau_{it} R_{it}$ for $i = 1, \ldots, N$. From Figure 5.8, it is clear that SG+TDMA is worse than BCD in terms of fairness. Especially for eight users, $FI_{SG+TDMA} = 0.41$ whereas $FI_{BCD} = 0.80$. Although low path losses embrace lower utility improvement, they mainly allow BCD algorithm to be very efficient in terms of fairness, *e.g.*, above 0.8. However as the path loss difference between users increase, completely fair allocations may not be the optimal ones. For instance, when there are eight users in the system the path losses of the users in case c) are; 25, 28,

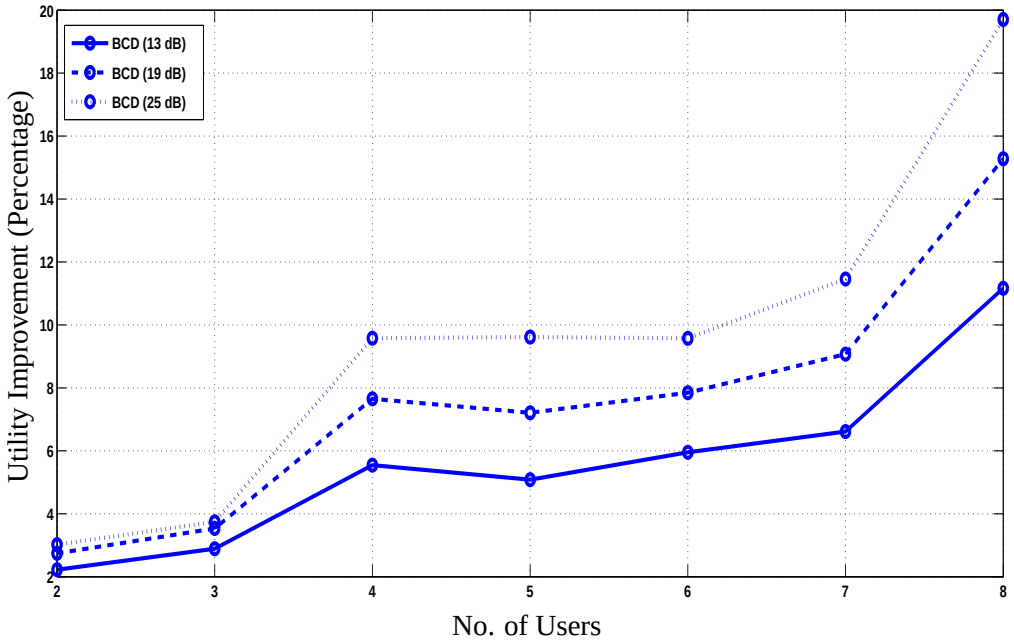

Figure 5.7: Utility improvement vs. no. of users: The utility improvement of the proposed algorithm over SG+TDMA, for increasing number of users, are compared. The effect of path loss (the strongest user's path loss is shown between parentheses) on utility improvement is shown. As path losses of the users increase the utility improvement increases, i.e, BCD's performance improves as the channel quality becomes degraded.

31, 34, 37, 40, 43, 46 dB, yielding an excessive difference of 23 dB between the weakest and the strongest user. In this case, the algorithm should favor user 1 more than it favors user 8, in order to maximize the utility function, causing a proportionally (instead of purely) fair allocation.

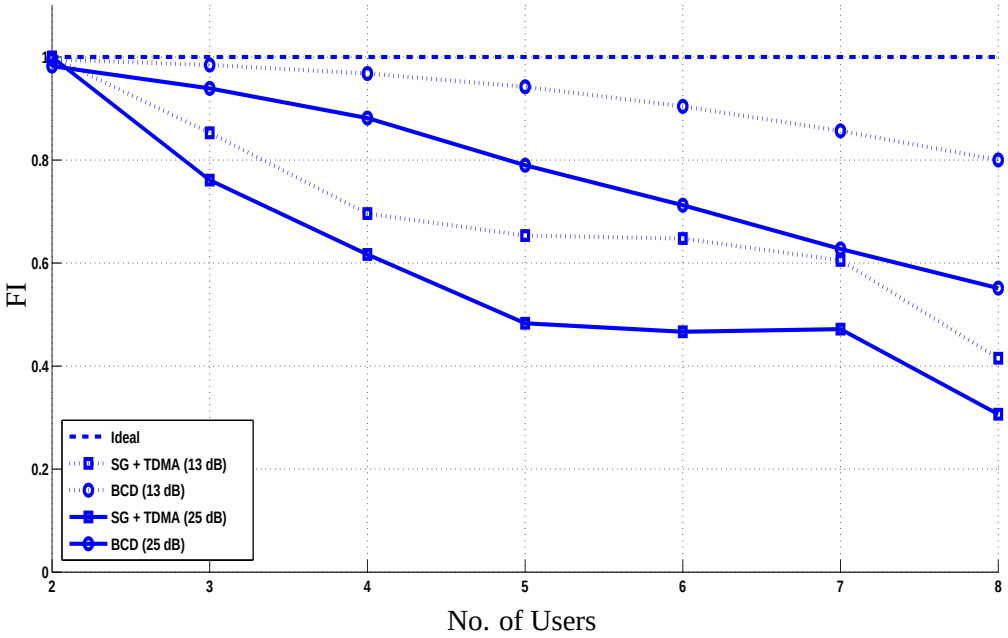

Figure 5.8: Fairness index (SG+TDMA, BCD) vs. no. of users: The fairness of SG+TDMA and the proposed algorithm, for increasing number of users, are compared through *FI*, which takes the value of 1 when there is a complete fair allocation. The effect of path loss (the strongest user's path loss is shown between parentheses) on fairness is shown. As difference among the path losses of the users increase the fairness indexes of the schemes decrease (SG+TDMA being the worst), causing comparatively unfair allocations among users.

CHAPTER 6

PROPORTIONAL FAIR RESOURCE ALLOCATION ON AN ENERGY HARVESTING DOWNLINK - PART II: ALGORITHMS

In this chapter, we show that by using the periodic energy arrivals assumption, it is possible to analytically derive the characteristics of the optimal solution of the Problem proposed in Chapter 5. In Chapter 5, we proved that the problem in hand is a biconvex problem and has multiple optima. This allowed us to decompose the problem into two parts (power allocation, time allocation) and present a Block Coordinate Descent based optimization algorithm, BCD [62], that converges to a partial optimal solution. Although BCD is guaranteed to converge to a partial optimal solution and thus the partial optimal utility, it is computationally expensive and when there are tens of users and energy arrivals, forming invertible hessian matrices (needed for the optimization of the power variables) may be computationally excessive. Hence in this chapter, we first derive the characteristics of the optimal solution of Problem 1 and then, build on those to develop simple and computationally scalable heuristics, PTF and ProNTO [63] that closely track the performance of the BCD solution.

6.1 System Model

Consider a time-slotted system where each frame, of length F_i, is divided into K slots. There is a single energy harvesting transmitter that transmits to N users by time sharing. Note that, we use the same system model as in Chapter 5. However, unlike Chapter 5, in this chapter we assume periodic energy arrivals and hence equal slot lengths ($T_t = T$ for all $t = 1, \ldots, K$), as shown in Figure 6.1, to reveal the characteristics of the optimal solution of Problem 1.

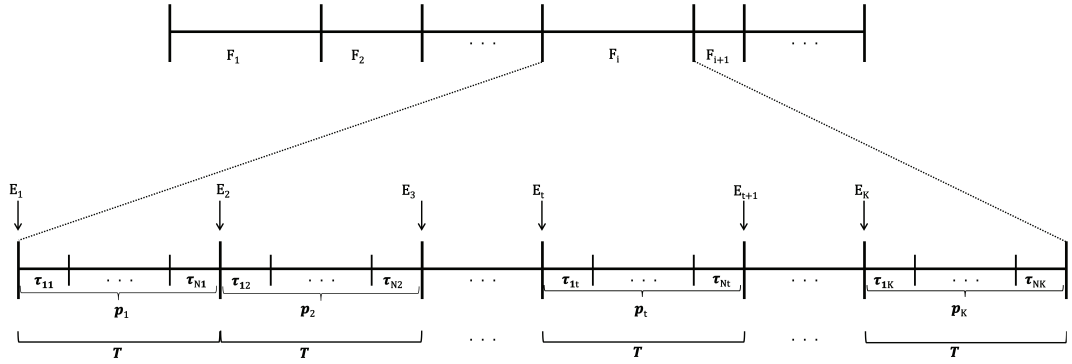

Figure 6.1: Problem illustration: There are K energy arivals in a frame, and, the time between consecutive arrivals are allocated to N users.

Similar to the setting in Chapter 5, for a given frame, the transmitter chooses a power level p_t and a time allocation vector $\tau_t = (\tau_{1t}, ..., \tau_{Nt})$, for each time slot t of the frame, where $p_{nt} = p_t$ is the selected transmission power for user n during slot t and, τ_{nt} is the time allocated for transmission to user n during slot t.

6.2 Structure and Properties of the Optimal Solution

In this section, we analyze the structure and properties of the hybrid power-time allocation policy. Remember that the utility function of Problem 1 is

$$U = \sum_{n=1}^{N} log_2(\sum_{t=1}^{K} \tau_{nt} R_{nt}) \tag{6.1}$$

where R_{nt} represents the rate of link n in t^{th} slot, as in Eq. (5.7).

Let $\overline{R_n} = [R_{n1}\ R_{n2}\ \ldots\ R_{nK}]^T$ and $\overline{\tau_n} = [\tau_{n1}\ \tau_{n2}\ \ldots\ \tau_{nK}]^T$. Then, utility can be rewritten as

$$U = \sum_{n=1}^{N} log_2(\overline{\tau_n}^T \overline{R_n}) \tag{6.2}$$

$$= U_1 + U_2 + \ldots + U_N \tag{6.3}$$

where U_n, the utility of user n, is

$$U_n = log_2(\overline{\tau_n}^T \overline{R_n}) \tag{6.4}$$

In order to reveal characteristics related to the optimal solution that will help us develop computationally efficient and close-to-optimal heuristics, we decompose the problem into two parts (similarly as in Chapter 5): power allocation and time allocation.

6.2.1 Structure of an Optimal Power Allocation Policy

In this section, we analyze the structure and properties of the optimal power allocation policy. In order to do this, we assume that the time allocation is determined, and try to characterize the structure of the optimal solution of the power allocation problem for this time allocation. Clearly, when the only variables are power variables, Problem 1 reduces to the following constrained optimization problem, which is equivalent to Problem 2 of Chapter 5:

Problem 4

$$\textit{Maximize: } U(\overline{p}) = \sum_{n=1}^{N} U_n(\overline{p})$$

$$\textit{subject to: } p_t \geq 0 \tag{6.5}$$

$$\sum_{i=1}^{t} p_i T_i \leq \sum_{i=1}^{t} E_i \tag{6.6}$$

where $t = 1, ..., K$ and, U_n is a function of the power variables (as defined in Eq. (6.4)). In previous chapter, Chapter 5, we proved the strict convexity[1] of Problem 4. Similarly, the general problem, Problem 1, is shown to be a biconvex optimization problem that has many local minima [62]. As Problem 4 has a unique optimum, the optimal power allocation changes for every given time allocation. In Theorem 6.2.1, we claim that one of the optimum schedules of Problem 1 has a nondecreasing power schedule. Lemma 6.2.3 not only helps us to prove our claim but also reveals that Problem 1 has multiple optima. From the proof Lemma 6.2.3, the attentive reader can observe that any feasible permutation[2] of the optimal

[1] Maximizing $U(\overline{p})$ is equivalent to minimizing $-U(\overline{p})$ which is a convex objective function.

[2] A feasible permutation is any permutation of a given schedule that does not violate the constraints described

schedule $(\overline{\tau}^*, \overline{p}^*)$, described in Theorem 6.2.1, is also optimal.

Theorem 6.2.1 *When all slots have equal length ($T_j = T,\ for\ \forall j \in \{1, ..., K\}$), there exists an optimal schedule $(\overline{\tau}^*, \overline{p}^*)$ such that $\overline{p}^*$ is nondecreasing, (e.g., $\overline{p}^* = (p_1, ..., p_K)$ where $p_1 \leq p_2 \leq ... \leq p_K$).*

Proof. The proof is provided in Appendix C.1, and rests on Lemma 6.2.3 below. ■

We shall need the following definition of a permutation of a vector sorted in nondecreasing order of elements, for stating Lemma 6.2.3.

Definition 6.2.2 *Given a vector $\overline{R_n} = [R_{n1}\, R_{n2}\, \ldots\, R_{nK}]^T$, we define $\overline{R_n}^{\uparrow} = [R_{n\pi(1)}\, R_{n\pi(2)}\, \ldots\, R_{n\pi(K)}]^T$ where $\overline{R_n}^{\uparrow}$ is a permutation (sorted in increasing order) of $\overline{R_n}$, such that*

$$R_{n\pi(1)} \leq \ldots \leq R_{n\pi(2)} \leq \ldots \leq R_{n\pi(K)} \tag{6.7}$$

Lemma 6.2.3 *When all slots have equal length ($T_j = T,\ for\ \forall j \in \{1, ..., K\}$), for any given schedule $(\overline{\tau}, \mathcal{P}_C)$, we can find such $\overline{\tau'_n}$,$\overline{R'_n}$ (where $\overline{R'_n} = \overline{R_n}^{\uparrow}$) that $(\overline{\tau'_n})^T \overline{R'_n} = \overline{\tau_n}^T \overline{R_n}$ for all $n = 1, \ldots, N$; i.e., the utility, U, does not change. Hence, if $(\overline{\tau_n}^*, \overline{R_n}^*)$ is optimal, then $(\overline{\tau'_n}^*, \overline{R'_n}^*)$ is also optimal.*

Proof. The proof is provided in Appendix C.2. ■

6.2.2 Structure of an Optimal Time Allocation Policy

In this section, we assume that the power allocation through the slots is determined. Then, given that the power variables are known constants, we determine the structure and properties of the optimal time allocation policy. Since, the only variables are time variables, Problem 1 reduces to Problem 5, which is equivalent to Problem 3 of Chapter 5:

in Eqns. (1)-(5.4).

Problem 5

$$\textit{Maximize: } U(\overline{\tau}) = \sum_{n=1}^{N} U_n(\overline{\tau})$$

$$\textit{subject to: } \tau_{nt} \geq 0 \tag{6.8}$$

$$\sum_{n=1}^{N} \tau_{nt} = T \tag{6.9}$$

$$\sum_{t=1}^{K} \tau_{nt} \geq \epsilon \tag{6.10}$$

where $t = 1, ..., K$, $n = 1, ..., N$ and, U_n is a function of the time variables (as defined in Eq. (6.4)). In Chapter 5, Problem 5 is shown to be convex. Thus, the analysis can rely on KKT (Karush-Kuhn-Tucker) optimality conditions, which must be satisfied by the global optimum. We start by forming the Lagrangian function as follows:

$$L(\overline{\tau}, \overline{\lambda}, \overline{\mu}) = -U(\overline{\tau}) + \sum_{j=1}^{K}\sum_{i=1}^{N} \mu_{(N(j-1)+i)}\tau_{ij} + \sum_{j=NK+1}^{NK+N} \mu_j(\epsilon - \sum_{t=1}^{K} \tau_{(j-NK)t}) + \sum_{i=1}^{K} \lambda_i(\sum_{n=1}^{N} \tau_{ni} - T_i) \tag{6.11}$$

where μ's are the Lagrange multipliers, and, the total number of constraints[3] is $N(K+1)+K$. After defining the Lagrangian as in Eq. (6.11), one can construct the KKT conditions for the optimal solution, which are presented in Appendix C.3. Please note that the optimal time allocation should jointly satisfy the set of equations that arise from KKT conditions. Clearly, as the number of users, N, and, the number of slots, K, increase, the number of equations increases dramatically making it cumbersome to write analytical solutions. Therefore, for the sake of conciseness, we continue the analysis with the special case of two users and two slots which allows us to construct the characteristics of the optimal time allocation policy.

Consider two consequtive slots with different power levels. Let us call the one with the least power *the weak slot*, and the one with the highest power *the strong slot*. When the slots have equal length ($T_1 = T_2 = T$), the optimal policy has the properties described in Lemma 6.2.4.

[3] There are K equality constraints and $NK + N$ inequality constraints.

Lemma 6.2.4 *In an optimal schedule, time allocation over the two slots (of equal length) has the following properties:*

1. *The weak slot is assigned to only one of the users. The strong slot, however, is shared between users. When both power levels are equal; if one slot is assigned to user 1 (user 2), the other slot is assigned to user 2 (user 1).*

2. *To whom the the weak slot will be assigned depends on two criteria: first,* $\Gamma_n = \frac{R_{n2}}{R_{n1}}$, *which is the ratio of user n's rate in the second slot to that in the first, and second, whether the strong slot is before or after the weak slot. When the weak slot preceeds the strong slot, it is assigned to the user with the smaller* Γ. *Otherwise (implying the decrease in power level), it is assigned to the user with the higher* Γ.

3. *In a strong slot, the user that did not (or will not) receive any data in the weak slot is favored, i.e., more than half of the slot is assigned to that user. In order to preserve fairness, this favoring operation is done by considering* Γ_1 *and* Γ_2.

Proof. The proof is provided in Appendix C.3. ■

6.3 PTF Heuristic

In this section, we develop a heuristic algorithm, Power-Time-Fair (PTF), based on the characteristics (discovered in the previous section) of an optimal power/time allocation schedule. As also described in [63], the PTF algorithm operates as follows:

1. **For Power Allocation:** Assign nondecreasing powers through the slots by using the energy harvest statistics, as follows:

   (a) From a slot, say i, to the next one $i+1$: If harvested energy decreases, defer a Δ amount of energy from slot i to slot $i+1$ to equalize the power levels. Do this until all powers are nondecreasing, and, form a virtual nondecreasing harvest order.

   (b) By using the virtual harvest order, assign nondecreasing powers through the slots, i.e., in each slot, spend what you virtually harvested at the beginning of that slot.

2. **For Time Allocation:** For the power allocation found in 1), let, $B_{nt} = R_{nt}T$ be the number of bits that would be sent by user n if the whole slot (of length T) was allocated to that user. Assign the first slot to the user who has the maximum rate, R_{nt}, in that slot. For the other slots, apply the following: At the beginning of each slot, $t \in \{2, \ldots, K\}$, determine the user with the maximum β where,

$$\beta_n = \frac{B_{nt}}{\sum_{i=1}^{t} B_{ni}}$$

and, assign the whole slot to that user. If multiple users share the same β, then, allocate the slot to the user with the best channel.

Simulation results show that the performance of the PTF algorithm is close to the performance of the BCD algorithm.

6.4 ProNTO Heuristic

In this section, we develop a fast and simple heuristic, ProNTO (Powers Nondecreasing - Time Ordered), based on the optimal power allocation related characteristics discovered in Section 6.2.1 and the simulation results obtained by running BCD algorithm for periodic energy arrivals. The ProNTO algorithm operates as follows:

1. **For Power Allocation:** Assign nondecreasing powers through the slots by using the energy harvest statistics, as done in part (1) of PTF algorithm.

2. **For Time Allocation:** Order the users, $u_1, \ldots, u_N$, according to their channel quality and form a user priority vector, $\overline{u^{\downarrow}} = [u_1^{\downarrow}, \ldots, u_N^{\downarrow}]$ where $u_1^{\downarrow}$ represents the user with the best channel. As $K > N$, Allocate every user $\frac{K-mod(K,N)}{N}$ slots as follows: The first $\frac{K-mod(K,N)}{N}$ slots are allocated to $u_1^{\downarrow}$, the next $\frac{K-mod(K,N)}{N}$ slots are allocated to $u_2^{\downarrow}$, etc. Add the remaining $mod(K,N)$ slots to the most powerful $mod(K,N)$ users' slots. For example; Let $K = 12$ and $N = 5$, and the path losses of the users to be 13 dB, 17 dB, 10 dB, 12 dB, 20 dB respectively. Then, the first 3 slots are allocated to user 3, the next 3 slots are allocated to user 4, the following 2 slots are allocated to user 1, 9^{th} and 10^{th} slots are allocated to user 2, and the last 2 slots are allocated to user 5.

Thus PTF and ProNTO differ only in time allocation part. The time allocation method used in ProNTO is proposed according to the following observation: when a partial optimal solution

obtained by BCD algorithm is modified as described in Lemma 6.2.3 and its proof, to form the nondecreasing optimal schedule, the time allocation becomes ordered, e.g., as shown in Table 6.2. As time allocation method used in ProNTO is simpler than the one used in PTF, ProNTO can operate faster. Simulation results show that the performance of ProNTO is close to the performance of the BCD algorithm.

6.5 Numerical and Simulation Results

In this section, we present the numerical and simulation results related to PTF and ProNTO heuristics. Throughout our simulations, we use the following setup: $W = 1kHz$, $N_o = 10^{-6}W/Hz$. We assume that some amount of energy ($\epsilon < E < \infty$ where ϵ is an infinitely small value) is harvested every 10 seconds ($T = 10$), within a frame (period of known harvests). Note that, throughout this section, the units used for frame length, energy, and power are; seconds, Joules, and Watts respectively. Throughout our simulations, we use four different frame lengths; 20, 80, 100, 120. For the frame of 20 secs, we use three different energy harvest models; $[0.5, 50]$, $[50, 0.5]$, $[60, 20]$. We define different cases for the remaining three frame lengths; *Regular*, *Bursty*, and, *Very Bursty*. In *Regular*, the harvest amounts are close to each other and form a regular pattern; $E_R = [73, 65, 9, 19, 40, 37, 22, 84, 39, 67, 81, 100]$. In *Bursty*, there are short term sudden decreases and increases in harvest amounts, causing a bursty pattern; $E_B = [20, 100, 1, 1, 1, 70, 100, 1, 10, 40]$. Finally, *Very Bursty* represents an extreme case where the transmitter stays energy-hungry for a long time; $E_V = [90, 2, 0.5, 0.1, 0.3, 0.7, 40, 60]$.

We start by the simplest case of two users and two slots ($N = 2$, $K = 2$, frame of 20 secs) to compare the results obtained by BCD algorithm [62], with the optimal ones presented in Table C.2. Our objective in doing such a comparison is to prove the accuracy of both theoretical and simulation results. We refer the interested reader to Appendix C.3 for the details of the optimality table, and provide the comparison in Table 6.1. Note that as in Chapter 5, the starting point of the algorithm is the Spend What You Get (SG) policy (proposed by Gorlatova et. al. [199]) combined with TDMA time allocation (SG+TDMA). The first column of Table 6.1 shows the amount of the harvests (E_1, E_2). The second column represents the mean path loss (in dB) of the two users. As observed from the table, for a given power allocation, the results found by BCD algorithm and the optimal ones (obtained by KKT optimality conditions) are

almost the same, verifying the consistency and optimality of the algorithm.

Table 6.1: BCD vs. optimal results for the special case of two users and two slots

Harvests	Mean Path loss	Power Allocation by BCD	Time Allocation by BCD	Optimal Time Allocation	Utility by BCD	Optimal Utility
[0.5 50]	20.5	$[0.050 \quad 5.000]$	$\begin{bmatrix} 10 & 4.412 \\ 0 & 5.587 \end{bmatrix}$	$\begin{bmatrix} 10 & 4.412 \\ 0 & 5.587 \end{bmatrix}$	29.809	29.809
	26.5	$[0.050 \quad 5.000]$	$\begin{bmatrix} 10 & 4.739 \\ 0 & 5.260 \end{bmatrix}$	$\begin{bmatrix} 10 & 4.739 \\ 0 & 5.260 \end{bmatrix}$	28.406	28.406
[50 0.5]	20.5	$[2.299 \quad 2.750]$	$\begin{bmatrix} 10 & 0.242 \\ 0 & 9.757 \end{bmatrix}$	$\begin{bmatrix} 10 & 0.243 \\ 0 & 9.756 \end{bmatrix}$	30.940	30.940
	26.5	$[2.246 \quad 2.803]$	$\begin{bmatrix} 10 & 0.429 \\ 0 & 9.570 \end{bmatrix}$	$\begin{bmatrix} 10 & 0.429 \\ 0 & 9.570 \end{bmatrix}$	29.461	29.461
[60 20]	2.5	$[3.823 \quad 4.176]$	$\begin{bmatrix} 10 & 0.053 \\ 0 & 9.946 \end{bmatrix}$	$\begin{bmatrix} 10 & 0.054 \\ 0 & 9.945 \end{bmatrix}$	33.527	33.527
	8.5	$[3.787 \quad 4.212]$	$\begin{bmatrix} 10 & 0.078 \\ 0 & 9.921 \end{bmatrix}$	$\begin{bmatrix} 10 & 0.078 \\ 0 & 9.921 \end{bmatrix}$	32.957	32.957

The attentive reader can observe from Table 6.1 that, when harvests decrease from one slot to another, the optimal powers tend to be nondecreasing. Hence in that case, the algorithm seems to be converged to the nondecreasing optimal discussed in Theorem 6.2.1. Note that, this nondecreasing optimal could also be obtained by using the modification method explained in Lemma 6.2.3. By using that method, we modify the results obtained by BCD algorithm to reveal the optimal (nondecreasing) power and time allocation policies for increasing number of users. For our analysis, we use three different path loss patterns, called, *Low*, *Moderate*, *High* respectively. In *Low*, the strongest user in the system has 13 dB path loss, and, every new user that joins the system deviates by 3 dB from the previous one (has 3 dB more path loss than the preceding user). In *Moderate*, the strongest user has 19 dB path loss, and, every new user deviates by 3 dB. Finally, in *High*, the strongest user has 25 dB path loss, and, every new user deviates 3 dB. Due to space limitations, we present only the *Bursty-Moderate* case's results in Table 6.2. As illustrated, when the number of users increase, BCD algorithm tends to assign increasing powers rather than nondecreasing. One can also see from the table that, no matter how many users exist in the system, ordering powers in nondecreasing order, causes the time allocation to be ordered too. By ordered, we mean that the first slot(s) are allocated to the user with the best channel, the next slot(s) are allocated to the user with the second best channel, etc. , and the last slot(s) are allocated to the user with the worst channel. This observation constitutes the main motivation for the ProNTO heuristic.

We next use the above-mentioned energy harvesting cases (*Regular*, *Bursty*, *Very Bursty*) to

Table 6.2: Optimal time and power allocation policies vs. number of users: Found by BCD algorithm and modified according to Lemma 6.2.3

No. of Users	T.A. / P.A.	Users/ Slots	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7	Slot 8	Slot 9	Slot 10
2	T.A.	1	10	10	10	10	10	3.666	0	0	0	0
		2	0	0	0	0	0	6.333	10	10	10	10
	P.A.		2	2.575	2.575	2.575	2.575	4.211	4.472	4.472	4.472	4.472
3	T.A.	1	10	10	10	7.188	0	0	0	0.012	0	0
		2	0	0	0	2.811	10	10	10	9.987	0	0
		3	0	0	0	0	0	0	0	0	10	10
	P.A.		2	2.415	2.415	2.553	2.916	3.916	4	4.727	4.728	4.728
4	T.A.	1	10	10	8.577	0	0	0	0	0	0	0
		2	0	0	1.422	10	10	4.759	0	0	0	0
		3	0	0	0	0	0	5.240	10	8.378	0	0
		4	0	0	0	0	0	0	0	1.621	10	10
	P.A.		2	2.313	2.381	2.802	2.802	3.634	4.050	4.207	5.074	5.074
5	T.A.	1	10	10	3.332	0	0	0	0	0	0	0
		2	0	0	6.667	10	5.107	0	0	0	0	0
		3	0	0	0	0	4.892	10	5.286	0	0	0
		4	0	0	0	0	0	0	4.713	10	3.425	0
		5	0	0	0	0	0	0	0	0	6.574	10
	P.A.		2	2.135	2.428	2.579	3.152	3.157	3.890	4.372	5.124	5.559
6	T.A.	1	10	10	0.533	0	0	0	0	0	0	0
		2	0	0	9.466	10	0	0	0	0	0	0
		3	0	0	0	0	10	7.386	0	0	0	0
		4	0	0	0	0	0	2.613	10	3.376	0	0
		5	0	0	0	0	0	0	0	6.623	7.772	0
		6	0	0	0	0	0	0	0	0	2.227	10
	P.A.		1.863	1.863	2.233	2.255	3.059	3.258	3.870	4.082	5.313	6.597
7	T.A.	1	10	8.988	0	0	0	0	0	0	0	0
		2	0	1.011	10	6.426	0	0	0	0	0	0
		3	0	0	0	3.573	10	2.435	0	0	0	0
		4	0	0	0	0	0	7.564	6.700	0	0	0
		5	0	0	0	0	0	0	3.299	10	0	0
		6	0	0	0	0	0	0	0	0	10	0.648
		7	0	0	0	0	0	0	0	0	0	9.351
	P.A.		1.634	1.670	2.008	2.310	2.654	2.976	4	4.009	5.761	7.373
8	T.A.	1	10	7.486	0	0	0	0	0	0	0	0
		2	0	2.513	10	3.789	0	0	0	0	0	0
		3	0	0	0	6.210	8.934	0	0	0	0	0
		4	0	0	0	0	1.065	10	2.048	0	0	0
		5	0	0	0	0	0	0	7.951	3.428	0	0
		6	0	0	0	0	0	0	0	6.571	3.748	0
		7	0	0	0	0	0	0	0	0	6.251	2.483
		8	0	0	0	0	0	0	0	0	0	7.516
	P.A.		1.484	1.567	1.827	2.100	2.341	2.932	4	4.413	5.766	7.965

compare the PTF and ProNTO heuristics' performances to that of BCD's. We start by testing the utility and throughput improvement (over SG+TDMA) performances of the heuristics for increasing path losses. For this, we set the number of users to two, i.e., $N = 2$. The results are presented in Figure 6.2 and Figure 6.3, respectively. In both figures, the Mean Path Loss, is computed as $\widetilde{L} = \frac{1}{N}\sum_{i=1}^{N} L_i$ where L_i represents the path loss of user i. Hence, the three mean path losses seen in the figures represent the *Low*, *Moderate* and *High* cases. One can observe from Figure 6.2 that, the utility improvements of all algorithms tend to increase (or at least stay constant) when path loss increases, and the utility improvement performances of the proposed heuristics are very close to that of BCD's. For the chosen cases, ProNTO outperforms PTF. This is more obvious for the *Very Bursty* case. The corresponding throughput improvements are shown in Figure 6.3. As illustrated, for the case of $N = 2$, even with $\approx 5\%$ of utility improvement, a $\approx 65\%$ of improvement in total throughput is possible. Note that, in all cases, the performances are very close to each other.

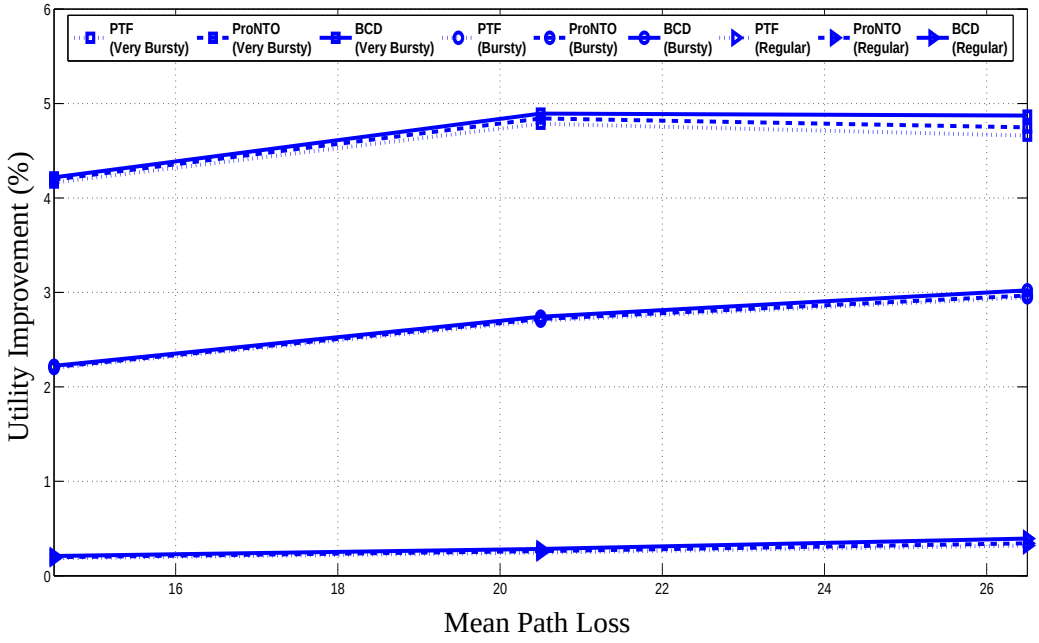

Figure 6.2: Utility improvement (BCD, PTF, ProNTO) vs. mean path loss for $N = 2$: The effect of mean path loss on utility improvement for the three energy harvesting cases; *Regular*, *Bursty*, *Very Bursty*

In order to determine the effect of number of users to the performances of our proposed heuristics, we next perform a series of simulations by considering all energy harvesting cases (*Regular*, *Bursty*, *Very Bursty*) and different number of users. By taking average over all

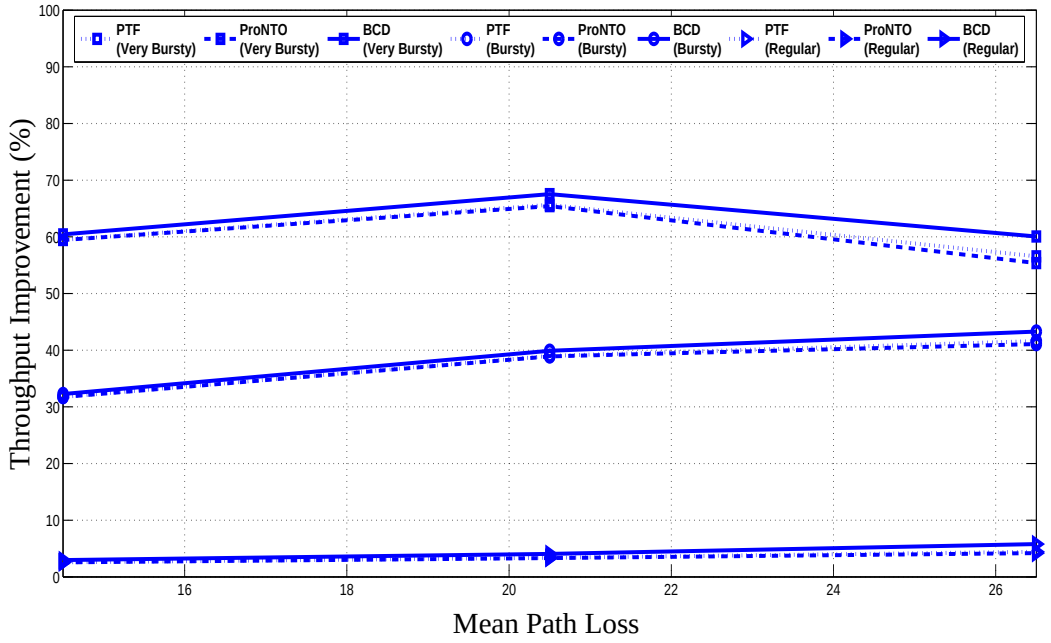

Figure 6.3: Throughput improvement (BCD, PTF, ProNTO) vs. mean path loss for $N = 2$: The effect of mean path loss on throughput improvement for the three energy harvesting cases; *Regular*, *Bursty*, *Very Bursty*

energy harvesting cases, we present the average utility improvement results in Figure 6.4, for the *Moderate* case. As illustrated in the figure, when the number of users increase, the average utility improvements of all schemes also increase. Note that, both heuristics closely track the BCD algorithm. When there are few users in the system, PTF and ProNTO are competitive. However, when there are more users, ProNTO seems to outperform PTF in terms of average utility improvement. At all instances, ProNTO is within the 1% neighbourhood of the BCD algorithm.

Although we aim at proportional fairness in this thesis, it may be interesting to analyse max-min fairnesses of the proposed algorithms, PTF and ProNTO, by using Jain's fairness index,FI. As in Chapter 5, we use Eq. (5.12) to compute FI. For computing FI, we use the no. of bits transmitted to the users, $x_i = \sum_{t=1}^{K} \tau_{it} R_{it}$ for $i = 1, \ldots, N$. From Table 6.3, it is clear that SG+TDMA is the worst choice in terms of fairness. Although low path losses embrace lower utility improvement, they mainly allow both PTF and ProNTO to be very efficient in terms of fairness. However, as observed from the table, when all three cases are considered, PTF seems to be more fair than ProNTO is. Hence, ProNTO seems to trade of fairness for utility improvement. It can also be inferred from Figure 6.4 and Table 6.3 that, when ProNTO

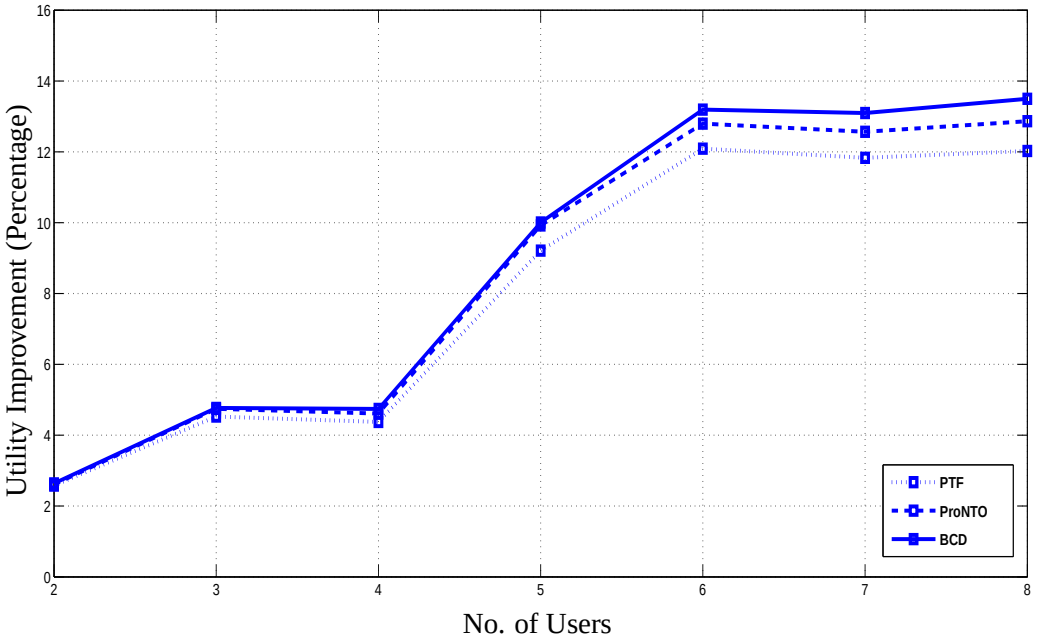

Figure 6.4: Average utility improvement (PTF, ProNTO, BCD) vs. no. of users: The average is taken over *Regular*, *Bursty*, *Very Bursty* cases. The average utility improvements of the proposed algorithms over SG+TDMA, for increasing number of users, are compared. Utility improvment increases with increasing number of users.

outperforms PTF in terms of utility improvement, the difference between two heuristics is not high. However, this is not the case for fairness,.i.e., when PTF outperforms ProNTO, the difference can be considered as high. Hence, although ProNTO seems more promising in terms of utility improvement, depending on system requirements, one can still choose PTF over ProNTO for more fairness.

Table 6.3: Fairness index (SG+TDMA, PTF, ProNTO, BCD) vs. no. of users: The fairness of PTF and ProNTO heuristics are compared to that of SG+TDMA's and BCD's, through FI.

Number of Users	Fairness Index (FI)											
	Regular				Bursty				Very Bursty			
	SG+TDMA	PTF	ProNTO	BCD	SG+TDMA	PTF	ProNTO	BCD	SG+TDMA	PTF	ProNTO	BCD
2	0.998	0.994	0.999	0.991	1.000	0.994	0.999	0.991	0.907	0.984	0.999	0.985
3	0.966	0.981	0.993	0.974	0.807	0.997	0.950	0.967	0.639	0.975	0.963	0.963
4	0.933	0.948	0.978	0.943	0.652	0.901	0.891	0.928	0.803	0.882	0.964	0.903
5	0.748	0.965	0.856	0.903	0.555	0.896	0.936	0.892	0.576	0.891	0.830	0.845
6	0.842	0.829	0.905	0.842	0.559	0.780	0.784	0.814	0.312	0.914	0.670	0.794
7	0.679	0.856	0.778	0.784	0.539	0.862	0.661	0.711	0.195	0.809	0.569	0.718
8	0.580	0.817	0.658	0.710	0.355	0.773	0.562	0.635	0.245	0.625	0.691	0.646

CHAPTER 7

PREDICTION BASED PROPORTIONAL FAIR RESOURCE ALLOCATION FOR INDUSTRIAL WIRELESS SENSOR NETWORKS

As industrial WSNs are expected to be deployed in harsh or inaccessible environments for long periods of time, a remote base station may be needed to control the operation of these networks. In industrial applications, it is very likely that an area needs to be covered with multiple WSNs. In such a case, each of these networks monitor different parameters and send data via the appropriate gateway nodes (cluster heads) to a base station node located at the central offices, where the strategic decisions about the data is taken.

In this chapter, we address the case where the base station is supplied with solar energy harvesting. Leveraging the daily periodicity of solar energy harvesting, we optimize the daily message delivery schedule from the base station to the nodes of a distributed network. The inherent differences in channel gain from the BS to the sensor nodes make it a challenge to provide service to each of them while efficiently spending the harvested energy. Leveraging a close-to-optimal algorithm developed for fair allocation of harvested energy in a wireless downlink, we develop a stand-alone algorithm, PTF-On, that operates two algorithms in tandem: A Kalman-based prediction algorithm and the modified version of the PTF algorithm. PTF-On can predict the base station's energy arrival profile throughout the day, and then, act upon this energy arrival profile to determine the best power and time allocation that will maximize the throughput (the amount of data sent to the gateway nodes) in a proportionally fair way.

7.1 System Model and Problem Statement

In our setup, there is a single base station that transmits to N gateways (or cluster heads) of several sensor networks by time sharing on a bandwith W, as shown in Figures 7.1 and 7.2. The power spectral density of the background noise is N_o. Channel conditions will be supposed to remain constant during a duration F that will be referred to as a "frame": i.e., g_n, the gain of gateway n, is chosen to be constant throughout the frame.

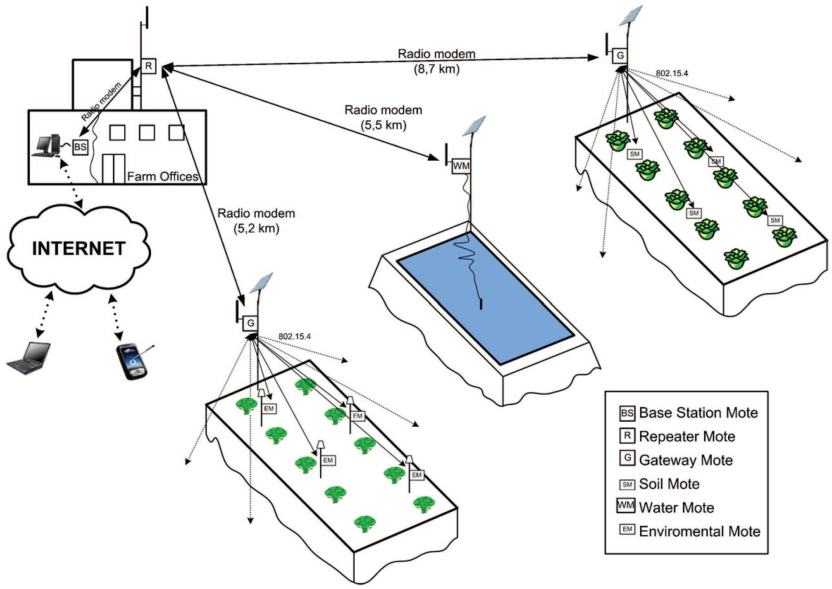

Figure 7.1: Industrial WSN application (agricultural monitoring) with remote base station. Example 1: [8]

The base station is equipped with a rechargeable battery, powered by a solar panel, such that harvested energy becomes available at distinct instances. The durations between two harvest instants will be called a "slot" (as in Chapters 5 and 6). Our system model is based on the one illustrated in Fig. 6.1. However, instead of considering only one frame, we consider multiple frames. We assume that the length of a frame is 24 hours. Note that, we restrict our attention to the case of periodic energy arrivals ($T_t = T$ for all $t \in \{1, \ldots, K\}$), as in Chapter 6). As also explained in Chapter 6, not all generality is lost, since harvest amounts are arbitrary and the absence of a harvest in a certain slot can be expressed with a harvest of amount zero for the respective slot. Thus, we define the length of a slot to be a sub-hour (half an hour). Thus, the amount of energy harvested from the environment at the beginning of time slot t of frame i is $E_{i,t}$, as illustrated in Fig. 7.3.

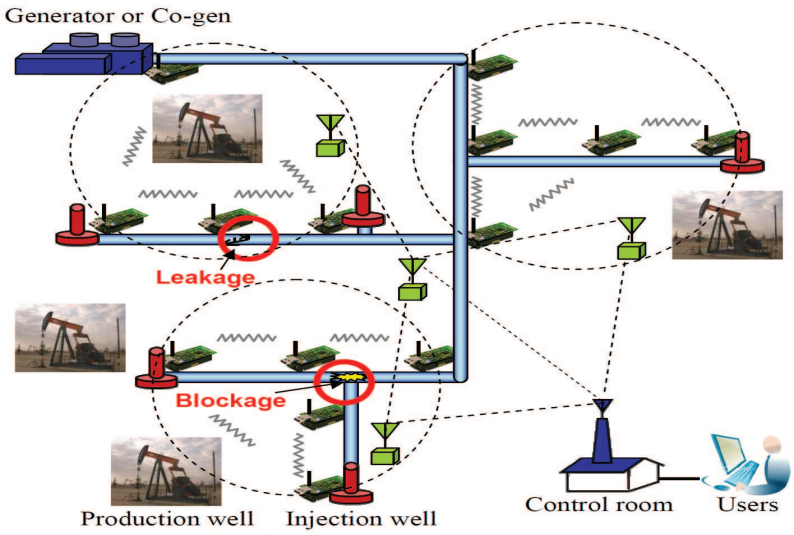

Figure 7.2: Industrial WSN application (pipeline monitoring) with remote base station. Example 2: [9]

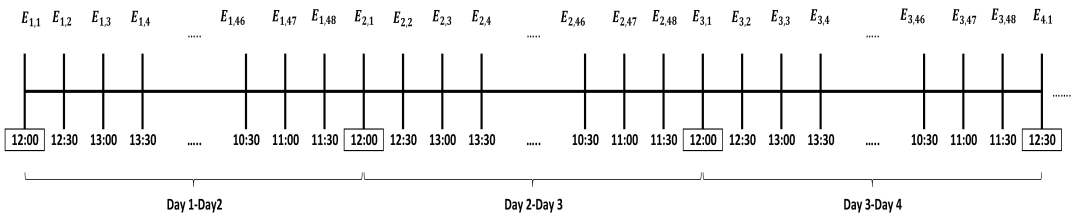

Figure 7.3: Energy arrival model

For a given frame, the base station chooses a power level p_t and a time allocation vector $\tau_t = (\tau_{1t}, ..., \tau_{Nt})$, for each time slot t of the frame, where $p_{nt} = p_t$ is the transmission power for gateway n during slot t and, τ_{nt} is the time allocated for transmission to gateway n during slot t. We use the same constrained optimization problem, Problem 1 proposed in Chapter 5. Please note that, Problem 1 is a biconvex optimization problem with multiple optima, and there exists an *offline* heuristic algorithm, PTF, that can closely track the optimal solution (solution found by BCD) of this problem. In this chapter, we modify the PTF algorithm so that we can use it in an online setting, i.e., the amounts of energy harvests within a frame are not known a priori. The modified version of the PTF algorithm will need to be combined with an energy prediction algorithm. There are some energy prediction algorithms available in the literature such as EWMA (Exponentially weighted moving average) [202] and WCMA (Weather conditioned moving average) [203]. However, these algorithms are simple moving average based prediction algorithms developed to operate on energy harvesting sensor nodes that do not have high computation capabilities. As our base station is equipped with a solar panel and is expected to harvest energy at a high rate, we developed a new energy prediction algorithm which will be explained in the next section.

7.2 Kalman-Based Solar Energy Prediction

In this section, we apply the Kalman filter algorithm to forecast the energy arrivals within a frame, for a base station powered with solar panel. We considered sub-hourly prediction of the energy arrivals for a frame of 24 hours (one day) and, formulated the Kalman filter for the following state and measurement models:

$$x(k+1) = \alpha_1 x(k) + \alpha_2 x(k-47) + \beta_1 y(k) + w(k) \tag{7.1}$$

$$z(k) = x(k) + v(k) \tag{7.2}$$

where x and z represent the state (energy level) and the measurement respectively. This model is mainly based on the idea that; due to the diurnal cycle of a day, the amount of energy that will be harvested in the $(k+1)^{th}$ sub-hour of an arbitrary day, $x(k+1)$, should be related to the energy harvested in the k^{th} sub-hour of the same day, $x(k)$, the solar irradiation received in the k^{th} sub-hour of the same day, $y(k)$, and, the energy harvested in the $(k+1)^{th}$ sub-hour

$$\begin{bmatrix} x(k+1) \\ x(k) \\ x(k-1) \\ \vdots \\ x(k-45) \\ x(k-46) \end{bmatrix} = \begin{bmatrix} \alpha_1 & 0 & 0 & \dots & 0 & 0 & \alpha_2 \\ 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 & 0 \\ \vdots & & & \ddots & & & \vdots \\ 0 & 0 & 0 & \dots & 1 & 0 & 0 \\ 0 & 0 & 0 & \dots & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x(k) \\ x(k-1) \\ x(k-2) \\ \vdots \\ x(k-46) \\ x(k-47) \end{bmatrix} + \begin{bmatrix} \beta_1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} y(k) + \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} w(k) \tag{7.3}$$

of the previous day (the energy that was harvested 48 sub-hours ago: $x((k+1)-48) = x(k-47)$), $x(k-47)$. In Eq. 7.1, $w(k)$ is a modeling error, which represents the effects of the uncontrolled events on the harvested energy (such as shadowing caused by clouds passing through, disturbance to the solar panel, or damage due to malicious act, etc.). In this paper, it is modeled as Gaussian i.i.d. with zero mean and variance σ_w^2 . The parameters α_1,α_2 and β_1 represent the weights assigned to emphasize the importance of the parameters that will be used for prediction. In the measurement model, v denotes the measurement noise and it is also modeled as Gaussian i.i.d. with zero mean and variance σ_v^2.

By considering that there are 48 sub-hours in a day, the overall state equations can be re-stated in matrix form as in Eq. (7.3). Now, we define an augmented state vector, $\overline{\xi_k}$, which contains the energy amounts harvested today:

$$\overline{\xi_k} = \begin{bmatrix} x(k) & x(k-1) & \dots & x(k-46) & x(k-47) \end{bmatrix}' \tag{7.4}$$

We define a new matrix **A**, column vectors $\overline{B}$, and $\overline{\Gamma}$ as follows:

$$\mathbf{A} = \begin{bmatrix} \alpha_1 & 0 & 0 & \dots & 0 & 0 & \alpha_2 \\ 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 & 0 \\ \vdots & & & \ddots & & & \vdots \\ 0 & 0 & 0 & \dots & 1 & 0 & 0 \\ 0 & 0 & 0 & \dots & 0 & 1 & 0 \end{bmatrix} \tag{7.5}$$

$$\overline{B} = \begin{bmatrix} \beta_1 & 0 & \dots & 0 & 0 \end{bmatrix}' \tag{7.6}$$

$$\overline{\Gamma} = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 \end{bmatrix}' \tag{7.7}$$

Thus, the state model in Eq. (7.3), and the measurement model in Eq. (7.2) reduce to

$$\overline{\xi_{k+1}} = \mathbf{A}\overline{\xi_k} + \overline{B}y(k) + \overline{\Gamma}w(k) \tag{7.8}$$

$$z(k) = x(k) + v(k) \tag{7.9}$$

which is structurally equivalent to the "truth" model described in Eq. (5.27) (in page 252) of [204]. Thus, by applying the Discrete-Time Linear Kalman Filter described in [204], we are able to predict the amount of energy arrival in the next sub-hour by only using the amount of energy arrival in this sub-hour, the solar irradiation received in this sub-hour and, the arrival in the previous day's next sub-hour. Please note that, in order to compute the best weights α_1, α_2 and β_1 that will be used for simulations, we use a data fitting method described as follows: By using the 18 days' data (real power measurements belonging to 01.10.2009-18.10.2009 for Amherst, Massachusetts, USA) provided by Navin Sharma [205], we design a Newton algorithm that aims to minimize the Mean Squared Error (MSE) between the real data and the estimated data, for 17 days ($17 days = 816 sub-hours$). Thus, the objective function that needs to be minimized by the Newton algorithm is described below:

$$\frac{1}{816}\sum_{k=48}^{863}(x(k+1) - (\alpha_1 x(k) + \alpha_2 x(k-47) + \beta_1 y(k)))^2 \tag{7.10}$$

Our simulation results, provided in Section 7.4, show that the best values for defined weights, α_1,α_2,β_1 are 0.7184, 0.1439, and, 0.0063 respectively, when the $x(k)$'s are in terms of kilo-joules.

7.3 PTF-On Algorithm

In this section, we propose an online proportional fair resource (power and time) allocation algorithm, called PTF-On. PTF-On is the online version of the PTF heuristic proposed in [61]. Note that the PTF algorithm operates in an *offline* fashion, i.e., the energy arrival amounts within a frame are known at the beginning of that frame. The main motivation of the PTF-On algorithm can be explained as follows: There are 48 sub-hours and thus, 48 energy arrivals within a frame (24 hours). At the beginning of a each slot, the current amount of residual energy and amounts of previous harvests are known. The amounts of next 47 energy arrivals

should be predicted. Thus, at the beginning of each frame we perform two prediction operations to determine the energy amounts that will be harvested during the frame. We perform this operation as follows: At the beginning of Slot 1, the energy arrives and is known to the base station. Thus, the base station can use K-SEP to predict its next energy arrival, i.e., the arrival in Slot 2. However, the arrivals other then the arrival in Slot 2 can not be predicted before a sub-hour passes. This is due to the fact that a half an hour should pass to see what is really harvested in Slot 2, so that this value can be used to predict the value in Slot 3. Thus, we adopt S-SEP, which does not use the data that was harvested in the previous slots (mainly predicts the amount of energy that will be harvested in today's k^{th} sub-hour as the average of the energy arrival amounts of the past two days' k^{th} sub-hours), to predict the next 46 arrivals. Thus, all energies (or at least their estimates) are known to the base station at the beginning of the frame. This way, at the beginning of each frame, one can run PTF algorithm to determine a close-to-optimal power and time allocation that will maximize the throughput in a proportionally fair way, for the up-coming 24 hours.

PTF-On requires past two days' data for predicting the energy arrival amounts of the day it will be used in. Assume that there are days 1,2,3,4,... etc, and, PTF-On will be used to predict the arrivals, and, determine the most proportional fair resource allocation, for the second half of day 3 and first half of day 4 (Frame of 24 hours: From 12:00 of day 3 to 12:00 of day 4). The operation of PTF-On algorithm is explained below, and illustrated in Figure 7.4.

1. For the 24-hours frame started at 12:00 of day 3, there will be 48 slots, each 30 minutes of length (Please note that this frame is called the original frame). The beginning of the whole frame will be the beginning of Slot 1. Thus, when the frame starts, the energy arrival at the beginning of slot one of day 3, $E_{3,1}$ is known. Thus, the energy arrival at the beginning of Slot 2, $E_{3,2}$, can be predicted by using the K-SEP algorithm. Then, use the S-SEP algorithm to obtain rough predictions of the others, $E_{3,3}, \ldots, E_{3,48}$, and form a predicted harvest series as follows: $\overline{E_{pred}} = [E_{3,1}, E'_{3,2}, E''_{3,3}, \ldots, E''_{3,48}]$, where E, E' and E'' represent the real, the K-SEP predicted, and the S-SEP predicted energy amounts, respectively.

2. As all the energy amounts (or at least their estimates) are known at the beginning of the frame, use the first part of the PTF algorithm to determine the best proportional fair power allocation (sub-hours) within the frame.

3. In the first slot of the frame, apply the power allocation found by the PTF algorithm for Slot 1 of that frame. Let, $B_{nt} = R_{nt}T$ be the number of bits that would be sent to gateway n if the whole slot (of length T) was allocated to that gateway. If this slot is the first slot of the original frame, assign this slot to the gateway who has the maximum rate, R_{nt}, in that slot. Otherwise, at the beginning of each slot, $t \in \{2, \ldots, K\}$, determine the gateway with the maximum β where, $\beta_n = \frac{B_{nt}}{\sum_{i=1}^{t} B_{ni}}$. Then, assign the whole slot to that gateway. If multiple gateways share the same β, then, allocate the slot to the gateway with the best channel.

4. When first slot of the frame finishes, and thus the second slot starts, assign Slot 2 of the current frame as the first slot of the upcoming frame (half an hour shifted version of the original frame), and estimate related energy amounts. Then, add the remaining energy to the energy of the first harvest of the new frame to form a new predicted harvest series. (Ex: At 12:30, $E_{3,2}$ is known and $E_{3,3}$ can be predicted by K-SEP. The remaining 46 energy harvests are predicted by S-SEP. Thus, a new predicted harvest series is formed: $\overline{E_{pred}} = [E_{3,2} + (E_{3,1} - p_1 T), E'_{3,3}, E''_{3,4}, \ldots, E''_{3,48}, E''_{4,1}]$.)

5. Apply Step 2,3, and 4 in order until the 24 hours is completed, i.e., the last slot of the original frame has been assigned a power and time allocation.

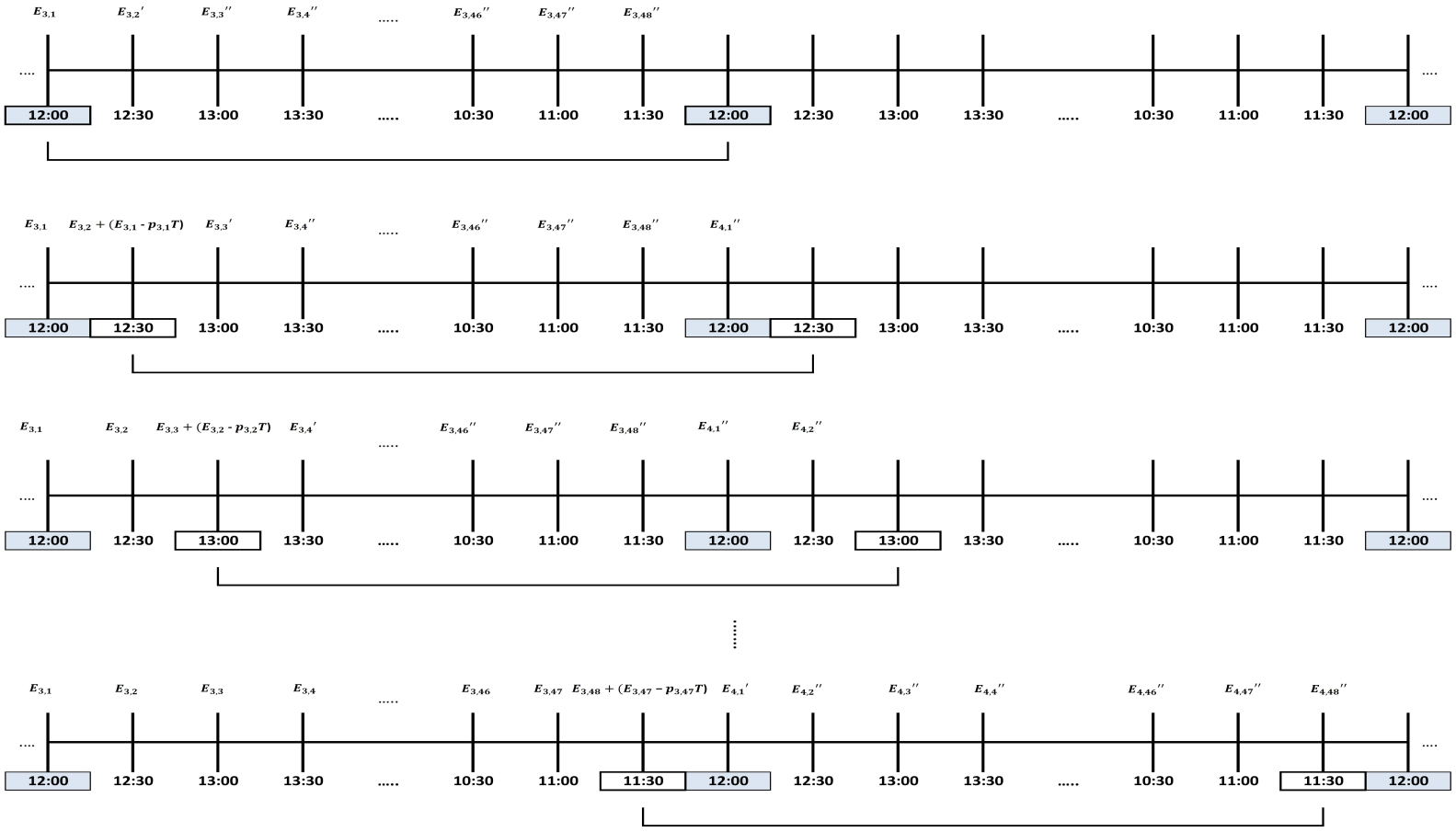

Figure 7.4: Operation of PTF-On algorithm

7.4 Numerical and Simulation Results

7.4.1 K-SEP and S-SEP Related Results

In this section, we present the numerical and simulation results related to our Kalman filter based solar energy prediction algorithm, called K-SEP, and the online resource allocation algorithm, PTF-On. By using the best weights that we computed by using the Newton algorithm, we perform numerous simulations to test our predictor. We test the performance of our predictor by the MSE criteria. We compute the MSE as follows:

$$MSE = \frac{1}{M}\sum_{i=1}^{M}(x_i - \widetilde{x}_i)^2 \tag{7.11}$$

where x and $\widetilde{x}$ represent the real and estimated energies respectively, and, M is the number of samples that will be considered. In order to compare the performance of our predictor with another one, we use a simple solar energy predictor, called S-SEP in this paper. We first let $M = 48$ (for 48 sub-hours in a day), and, compute daily MSE values for 16 days, as shown in Tables 7.1 and 7.2. Then, average MSE over 16 days of October, 2009 (03.10-2009-18.10.2009) for K-SEP and S-SEP are, $MSE_{Aver}^{K-SEP} = 4.377$ kilojoules/sub-hour/day and $MSE_{Aver}^{S-SEP} = 84.146$ kilojoules/sub-hour/day respectively. By considering that the maximum power measured in [205] was 60 Watts, one can produce maximum $E_{max} = 60.1800 = 108$ kilojoules in a sub-hour by using this system. Thus, the performance of S-SEP is much worse than the performance of K-SEP in terms of average error, i.e., $\sqrt{MSE_{Aver}^{K-SEP}} = 2.092$ kilojoules/sub-hour whereas, $\sqrt{MSE_{Aver}^{S-SEP}} = 9.173$ kilojoules/sub-hour.

Table 7.1: MSEs for the first 8 days

Days	3	4	5	6	7	8	9	10
MSE of K-SEP	0.168	12.064	9.517	17.076	6.937	3.646	0.329	2.691
MSE of S-SEP	106.689	266.271	122.629	89.941	141.008	43.907	122.341	15.874

Table 7.2: MSEs for the second 8 days

Days	11	12	13	14	15	16	17	18
MSE of K-SEP	2.028	2.328	3.451	3.612	0.287	1.093	2.569	0.194
MSE of S-SEP	90.872	20.191	69.325	26.108	32.366	5.634	57.263	135.915

The figures 7.5, 7.6, 7.7 illustrate the performances of the two predictors for three days in which S-SEP performs the best, the second best, and the worst in its 16 days's performance. As it can be seen from the figure, K-SEP outperforms S-SEP at all instances.

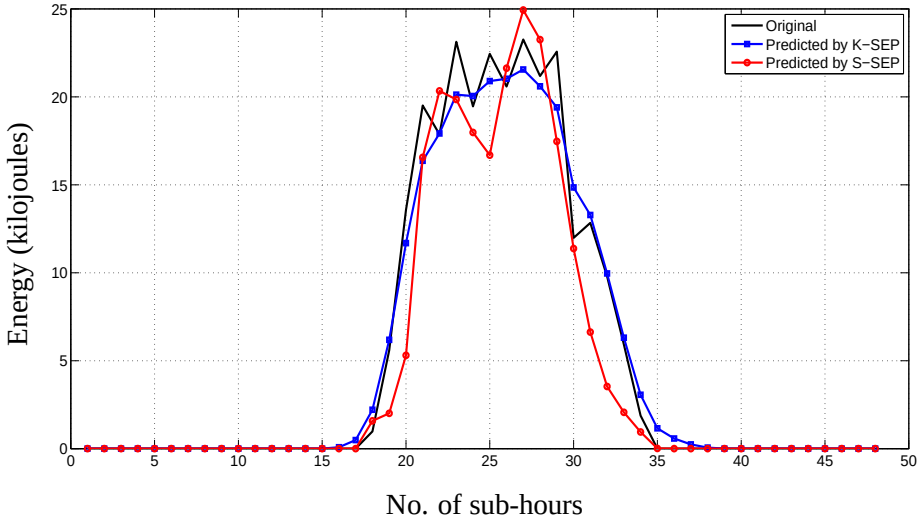

Figure 7.5: Performances of K-SEP and S-SEP for 16.10.2009

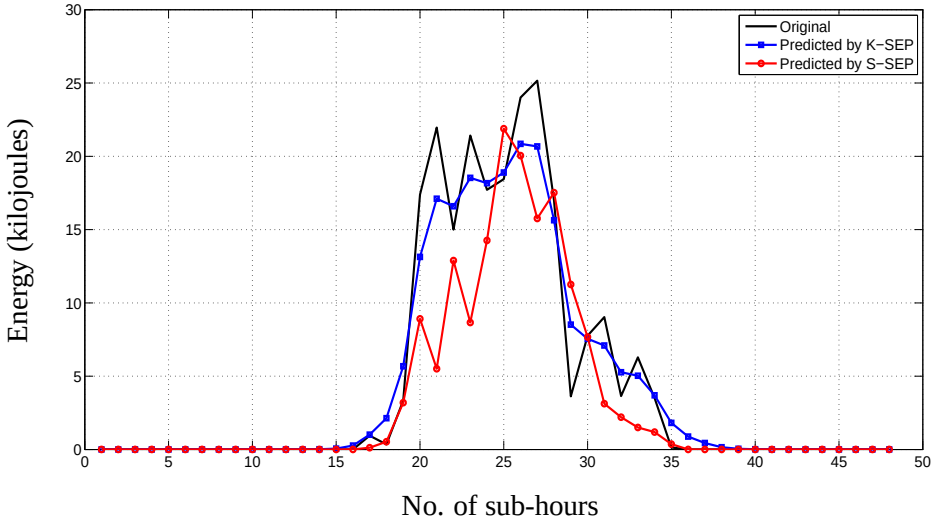

Figure 7.6: Performances of K-SEP and S-SEP for 10.10.2009

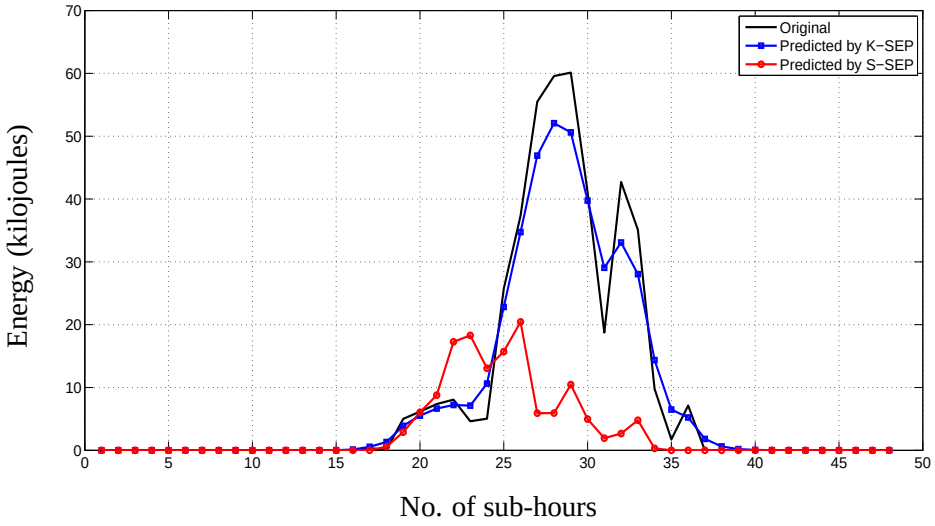

Figure 7.7: Performances of K-SEP and S-SEP for 04.10.2009

7.4.2 PTF-On Related Results

In this section, we present the numerical and simulation results related to the proposed online heuristic, PTF-On. Throughout our simulations, we use the folowing setup: $W = 10\ MHz$, $N_o = 10^{-19}\ W/Hz$. For the sake of an example, we suppose that there are three sensor networks, and thus three gateways in the system, similar to the one shown in Figure 7.1. The path loss of the gateways are 78, 92, and, 100 dB respectively. We compare the performance of the proposed algorithm with the performance of the "Spend What You Get" policy (where the amount of energy harvested at the beginning of a slot is completely spent during that slot) combined with TDMA time allocation, and with the performance of the *offline* PTF heuristic that operates close-to-optimal. We start our analysis at 12:00 pm on 03.10.2009 and finish it at 12:00 on 17.10.2009. Hence, we have 14 frames, each of which are 24 hours (48 sub-hours). For each frame, we test the performances of the PTF-On and PTF algorithms, and, the SG+TDMA scheme.

For the sake of an example, we illustrate the power allocations found by PTF and PTF-On for a frame of 24 hours (Frame 5), in Figure 7.8. As seen from the figure, the power levels of PTF and PTF-On are, most of the time, close to each other.

Utility and throughput improvement (over SG+TDMA scheme) results of PTF and PTF-On, for all 14 frames, are illustrated in Figures 7.9 and 7.10. As it can be seen from the figures, the performance of the proposed online algorithm, PTF-On, is very close to the performance

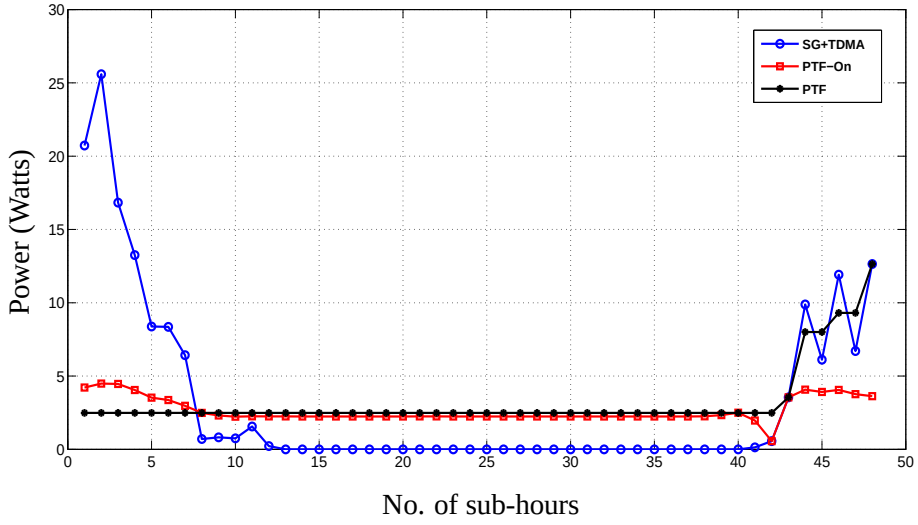

Figure 7.8: Power allocations found by PTF and PTF-On for a frame of 24 hours

of the *offline* PTF algorithm. Please note that, as the utility is defined as sum of "logarithms" of individual throughputs, even 1% improvement in utility is significant. This is more evident when the average (over 14 frames) utility and throughput improvements of PTF and PTF-On, are computed. For example; the average utility improvement for PTF is 3.310%, where the corresponding average throughput is 150.795%. Similarly, the average utility improvement for PTF-On is 3.158%, where the corresponding average throughput is 143.992%. Figure 7.11 shows the average daily throughputs of the gateways (average amount of data sent to the gateways in one day) and, the average daily total throughput.

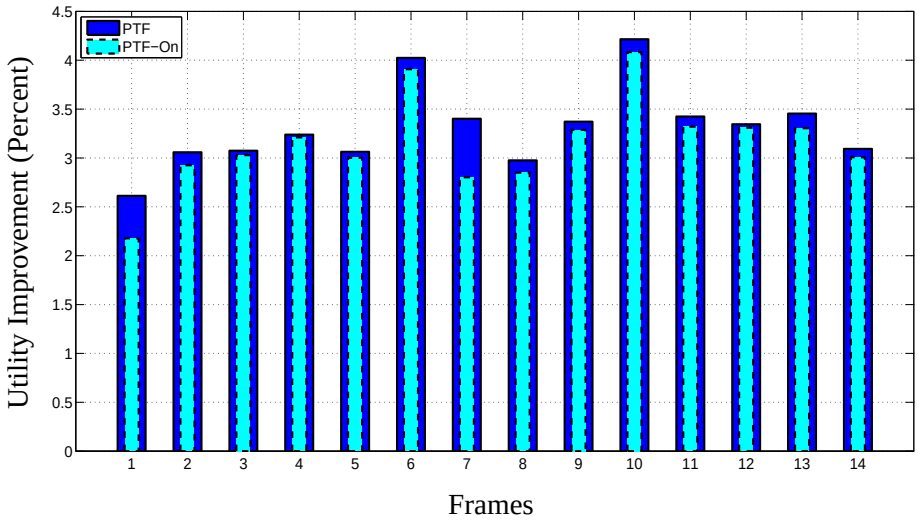

Figure 7.9: Utility improvements of PTF and PTF-On algorithms over SG+TDMA scheme for 14 frames

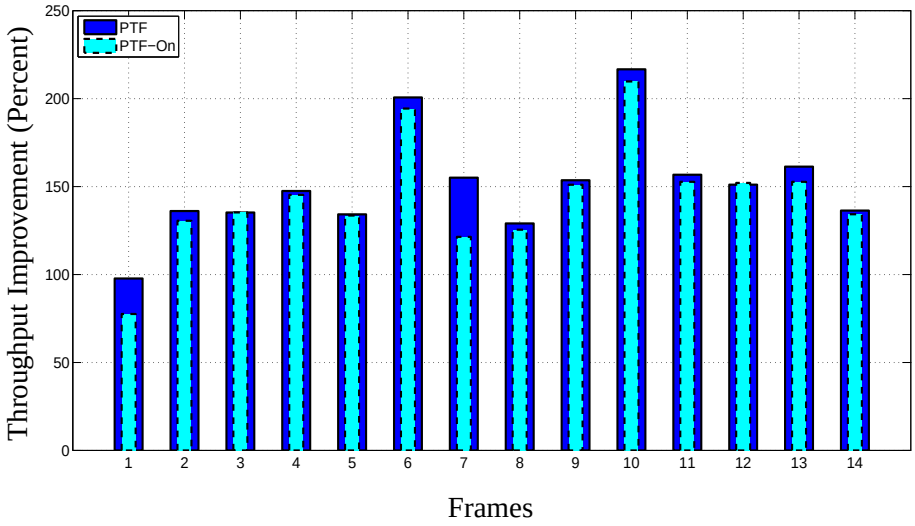

Figure 7.10: Throughput improvements of PTF and PTF-On algorithms over SG+TDMA scheme for 14 frames

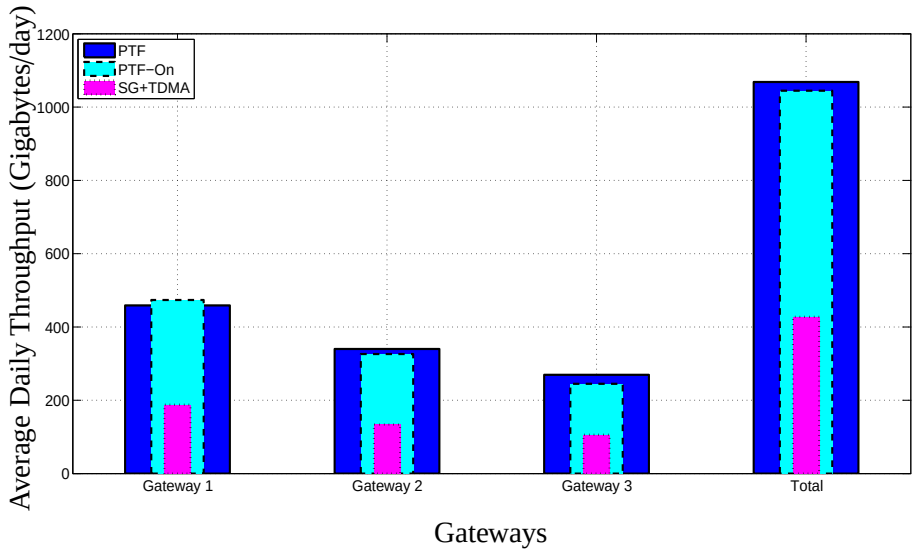

Figure 7.11: Average daily throughputs of the gateways, and average total throughput when PTF, PTF-On, and, SG+TDMA are used

CHAPTER 8

CONCLUSIONS AND FUTURE WORK

In this thesis, we first provided a classification of the current and future M2M applications and, we discussed the architecture and the design issues of M2M networks. M2M networks make use of multi-hop routing in order to route data in a wireless network. Therefore, the intelligent devices used in M2M networks should be reliable by means of availability. A key determinant of longevity and reliability of these networks is energy-efficient system architectures and algorithms. As ad hoc and sensor networks are subsets of M2M networks, we believe that forming energy efficient ad hoc and sensor networks is an important component of forming more energy efficient M2M networks. Hence, we next surveyed the shortest path based energy efficient routing algorithms developed for ad hoc and sensor networks in recent years. We also developed a classification for these algorithms, also considering the link cost metrics used in these algorithms.

By considering the fact that sustainable and environmentally friendly development of many wireless networks require increased use of renewable energy, we next investigated the state-of-the-art resource management and scheduling algorithms that can be used in energy harvesting industrial WSNs. Detailed operation of the algorithms, along with their drawbacks, advantages, and possible application areas are discussed.

By inspiring from the energy harvesting based resource allocation studies, we next investigated an original problem; the proportional fair power and time allocation problem in an energy harvesting broadcast system. This thesis focuses on finding the optimum *offline* schedule for this problem, by assuming that the energy harvesting times and the corresponding harvested energy amounts are known at the beginning of each frame. Detailed analysis of structural characteristics of the problem has been performed, which revealed that it can be

formulated as a biconvex optimization problem, and that it has multiple optima. Furthermore, an algorithm based on block coordinate descent (BCD), that surely converges to a partial optima of the problem, was showed. Building on the problem formulation and BCD, the optimal resource allocation policy was further studied and, the existence of an optimal nondecreasing power schedule and, an ordered time allocation schedule were proved. This allowed us to propose two alternative efficient and scalable heuristics, PTF and ProNTO. The computational ease of these algorithms were observed in numerical examples, while the policies they result in coincide with the structural properties we have shown the optimal to have. Simulation results indicate that, despite their simplistic design, PTF and ProNTO heuristics can closely track the performance of the optimal BCD algorithm. In our examples, which were computed for small or moderate problem sizes, both PTF and ProNTO took one or two orders of magnitude smaller time to converge than BCD, which has to compute a Hessian. Typically, ProNTO outperforms PTF in terms of utility improvement, whereas the latter is fairer. The utility improvement difference between BCD and ProNTO is shown to be less than 1% at all instances.

Finally, we developed an *online* algorithm that bypasses the need for *offline* knowledge about the energy harvesting arrivals. This algorithm employs energy harvesting prediction algorithms to predict the energy that will arrive in the future, and can be used in not only broadcast systems but also in industrial wireless sensor networks.

There are many possible future directions for the work presented in this thesis. The first direction could be developing a new PTF-On algorithm that will operate on shorter slot lengths. This will increase the number of maintenance messages and updates sent to the gateways, while improving the performance of the algorithm. However, as the number of slots and, shifting operations will increase, the ways of keeping the complexity low should be considered wisely. One idea is to shorten the frame length, from 24 hours to couple of hours, and to determine the allocation at the beginning of each frame. However, as the long term changes could not be considered in such a case, one should determine the best frame length that will provide *close-to-offline* performance.

Another direction can be modifying the ProNTO algorithm and combining it with the energy prediction algorithm, to obtain another *online* algorithm. That way the performance of the PTF-On and (possibly) ProNTO-On algorithms can be compared.

Other than changing the resource allocation method, one can modify the energy prediction algorithm so that it does not only predict the next energy arrival, but uses its own prediction as a measurement to predict the next arrivals, and this way become a stand-alone prediction algorithm. Using this method may improve the predictions of the algorithm and thus, the system performance. Moreover, the current Kalman algorithm may be modified to estimate the weights (parameters) at every step. That way the accuracy of the model would be improved.

It should be remembered that the solar energy harvesting application presented in this thesis is just an example. As shown by the simulation results, the performances of the proposed *offline* algorithms are higher when the energy arrivals are bursty. Although in solar energy harvesting applications, the base station (or transmitter) stays energy hungry during the night time (unless an efficient resource allocation method is applied), and thus, the harvesting nature becomes slightly bursty, there are some other types of energy harvesting methods, i.e., from vibrations and wind, etc. that suits this scenario better. By simple changing the prediction part of the proposed *online* algorithm, one may solve the proposed problem for different energy harvesting applications.

Until now, we considered the possible modifications to the proposed *online* setup and algorithm. However, there are some other possible directions that may require the modification of the proposed *offline* problem. For example, the proposed problem can be modified to consider battery inefficiencies such as leakage. Moreover, the finite battery case can be considered.

Priority is another issue that can be included to the problem formulation. In the case of industrial sensor network application presented in this thesis, sending update and maintenance messages to some of the gateways can be more important than sending to others. For example, among many sensor networks, one may need faster topology updates due to mobility/availability of its nodes. In such a case, the problem can be modified to favor some gateways in arbitrary slots.

Another possible direction would be considering existence of multiple base stations equipped with energy harvesting capabilities. In that case, data intended for some of the gateways can be sent from either of the base stations, and the resource allocation for these base stations can be done by considering both of the base stations' harvesting processes and the channel qualities of the users.

REFERENCES

[1] C. Pandana and K. Liu, "Robust connectivity-aware energy-efficient routing for wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 7, pp. 3904–3916, October 2008.

[2] Electsol, "Building automation," February 2010. http://www.electsol.com.sg/2.html.

[3] LCPC, February 2010. http://media.lcpc.fr/ext/img/rech/secteurs-01.jpg.

[4] SHRM, "Health care reform," February 2010. http://www.shrm.org/Advocacy/Issues/HealthCare/Pages/Default.aspx.

[5] SCADALink, "Scadalink remote video surveillance - industrial video monitoring over the internet," February 2010. http://www.scadalink.com/products/img/site-video-surveillance.jpg.

[6] E. Utopia, "The "smart grid" is all the buzz!," February 2010. http://energyutopia.wordpress.com/2009/06/07/the-smart-grid-is-all-the-buzz/.

[7] EnviroZine, "Track pollution levels in your community," February 2010. http://www.environment-canada.ca/EnviroZine/images/Issue42/industrial_plant_l.jpg.

[8] J. L. Riquelme, F. Soto, J. Suardiaz, P. Sanchez, A. Iborra, and J. Vera, "Wireless sensor networks for precision horticulture in southern spain," *Computers and Electronics in Agriculture*, vol. 68, no. 1, pp. 25 – 35, 2009.

[9] S. Yoon, W. Ye, J. S. Heidemann, B. Littlefield, and C. Shahabi, "Swats: Wireless sensor networks for steamflood and waterflood pipeline monitoring," *IEEE Network*, vol. 25, no. 1, pp. 50–56, 2011.

[10] G. Lawton, "Machine-to-machine technology gears up for growth," *Computer*, vol. 37, no. 9, pp. 12–15, 2004.

[11] J. Brazell, L. Donoho, J. Dexheimer, R. Hanneman, and G. Langdon, *M2M: The Wireless Revolution*. Texas State Technical College (TSTC) Publishing, 2005.

[12] A. Hansmann, "Wireless and green with cellular M2M." Asia-Pacific III, 2010.

[13] E. Oh and B. Krishnamachari, "Towards dynamic energy-efficient operation of cellular network infrastructure," *IEEE Communications Magazine*, November. , 2010.

[14] C. McVeigh, G. Sandieson, C. Burgess, T. Newitt, B. Webb, P. Lacy, J. Keeble, S. Yurisich, A. Gordon, and X. Veillard, "Carbon connections: Quantifying mobile's role in tackling climate change," tech. rep., Accenture and Vodafone, July 2009.

[15] M. Ye, C. Li, G. Chen, and J. Wu, "An energy efficient clustering scheme in wireless sensor networks," *Ad Hoc & Sensor Wireless Networks*, vol. 3, pp. 99–119, 2006.

[16] R. Kwok, "Use of 1) sensors and 2) radio frequency ID (RFID) for the national children's study," tech. rep., RTI International, August 2004.

[17] R. C. Shah and J. M. Rabaey, "Energy aware routing for low energy ad hoc sensor networks," in *IEEE Wireless Communications and Networking Conference (WCNC)*, March 2002.

[18] L. T. Saswat Misra and A. Ephremides, "Application dependent shortest path routing in ad-hoc sensor networks," in *Wireless Sensor Networks: Signal Processing and Communications Perspectives* (A. Swami, Q. Zhao, Y. Hong, and L. Tong, eds.), John Wiley and Sons, Ltd, 2007.

[19] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," *Computer Networks*, vol. 38, pp. 393–422, 2002.

[20] S. Tilak, N. B. Abu-ghazaleh, and W. Heinzelman, "A taxonomy of wireless microsensor network models," *ACM Mobile Computing and Communications Review*, vol. 6, pp. 28–36, 2002.

[21] K. Akkaya and M. F. Younis, "A survey on routing protocols for wireless sensor networks," *Ad Hoc Networks*, vol. 3, no. 3, pp. 325–349, 2005.

[22] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: a survey," *IEEE Wireless Communications*, vol. 11, no. 6, pp. 6–28, 2004.

[23] V. C. Gungor and G. P. Hancke, "Industrial wireless sensor networks: Challenges, design principles, and technical approaches," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 10, pp. 4258–4265, 2009.

[24] M. R. Akhondi, A. Talevski, S. Carlsen, and S. Petersen, "Applications of wireless sensor networks in the oil, gas and resources industries," in *Proceedings of the 24th IEEE International Conference on Advanced Information Networking and Applications (AINA)*, pp. 941–948, 2010.

[25] L. Krishnamurthy, R. Adler, P. Buonadonna, J. Chhabra, M. Flanigan, A. Kushalnagar, L. Nachman, and M. Yarvis, "Design and deployment of industrial sensor networks: experiences from a semiconductor plant and the north sea," in *Proceedings of the 3rd international conference on Embedded networked sensor systems*, pp. 64–75, ACM Press, 2005.

[26] R. Bogue, "Solar-powered sensors: a review of products and applications," *Sensor Review*, vol. 32, no. 2, pp. 95–100, 2012.

[27] A. Khaligh, P. Zeng, and C. Zheng, "Kinetic energy harvesting using piezoelectric and electromagnetic technologies - state of the art," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 3, pp. 850–860, 2010.

[28] M. I. Mohamed, Y. W. Wu, and M. Moniri, "Power harvesting for smart sensor networks in monitoring water distribution system," in *IEEE International Conference on Networking, Sensing and Control (ICNSC)*, pp. 393–398, 2011.

[29] K. T. Yen and S. K. Panda, "Energy harvesting for autonomous wireless sensor networks," *IEEE Transactions on Power Electronics*, vol. 26, no. 1, pp. 38–50, 2011.

[30] R. J. M. Vullers, R. V. Schaijk, H. J. Visser, J. Penders, and C. V. Hoof, "Energy harvesting for autonomous wireless sensor networks," *IEEE Solid-State Circuit Magazine*, vol. 2, no. 2, pp. 29–38, 2010.

[31] D. K. Noh and K. Kang, "A practical flow control scheme considering optimal energy allocation in solar-powered wsns," in *Proceedings of 18th Internatonal Conference on Computer Communications and Networks (ICCCN)*, pp. 1–6, 2009.

[32] X. Jiang, J. Polastre, and D. E. Culler, "Perpetual environmentally powered sensor networks," in *Proceedings of the Fourth International Symposium on Information Processing in Sensor Networks (IPSN)*, pp. 463–468, 2005.

[33] C. Park and P. Chou, "Ambimax: Autonomous energy harvesting platform for multi-supply wireless sensor nodes," in *Proceedings of the Sensor and Ad Hoc Communications and Networks (SECON)*, pp. 168–177, 2006.

[34] F. Simjee and P. H. Chou, "Everlast: long-life, supercapacitor-operated wireless sensor node," in *International symposium on Low power electronics and design (ISLPED)*, pp. 197–202, 2006.

[35] H. Yu and Q. Yue, "Indoor light energy harvesting system for energy-aware wireless sensor node," in *International Conference on Future Energy, Environment, and Materials*, pp. 1027–1032, 2012.

[36] H. Raisigel, G. Chabanis, I. Ressejac, and M. Trouillon, "Autonomous wireless sensor node for building climate conditioning application," in *Fourth International Conference on Sensor Technologies and Applications (SENSORCOMM)*, pp. 68–73, 2010.

[37] C. Moser, J.-J. Chen, and L. Thiele, "Optimal service level allocation in environmentally powered embedded systems," in *Proceedings of the 2009 ACM symposium on Applied Computing*, pp. 1650–1657, ACM, 2009.

[38] A. Hande, T. Polk, W. Walker, and D. Bhatia, "Indoor solar energy harvesting for sensor network router nodes," *Microprocess. Microsyst.*, vol. 31, no. 6, pp. 420–432, 2007.

[39] Z. Mao, C. E. Koksal, and N. B. Shroff, "Near optimal power and rate control of multi-hop sensor networks with energy replenishment: Basic limitations with finite energy and data storage," *IEEE Trans. Automatic Control*, vol. 57, no. 4, pp. 815–829, 2012.

[40] M. Paavola and K. Leiviska, "Wireless sensor networks in industrial automation," in *Factory Automation*, InTech, 2010.

[41] E. Uysal-Biyikoglu, B. Prabhakar, and A. E. Gamal, "Energy-efficient packet transmission over a wireless link," *IEEE/ACM Transactions on Networking*, vol. 10, pp. 487–499, August 2002.

[42] A. E. Gamal, C. Nair, B. Prabhakar, E. Uysal-Biyikoglu, and S. Zahedi, "Energy-efficient scheduling of packet transmissions over wireless networks," in *IEEE INFOCOM*, pp. 1773–1782, November 2002.

[43] E. Uysal-Biyikoglu and A. E. Gamal, "On adaptive transmission for energy efficiency in wireless data networks," *IEEE Transactions on Information Theory*, vol. 50, pp. 3081–3094, August 2004.

[44] R. Knopp and P. A. Humblet, "Information capacity and control in single-cell multiuser communications," in *Proceedings of IEEE International Communications Conference (ICC)*, pp. 331–335, June 1995.

[45] D. N. C. Tse, "Optimal power allocation over parallel gaussian broadcast channels," in *IEEE ISIT*, p. 27, June 1997.

[46] X. Liu, E. K. P. Chong, and N. B. Shroff, "Opportunistic transmission scheduling with resource-sharing constraints in wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 10, pp. 2053–2064, 2001.

[47] J. Yang and S. Ulukus, "Transmission completion time minimization in an energy harvesting system," in *44th Annual Conference on Information Sciences and Systems (CISS)*, pp. 1–6, March 2010.

[48] J. Yang, O. Ozel, and S. Ulukus, "Broadcasting with an energy harvesting rechargeable transmitter," *IEEE Transactions on Wireless Communications*, vol. 11, pp. 571–583, February 2012.

[49] M. A. Antepli, E. Uysal-Biyikoglu, and H. Erkal, "Optimal packet scheduling on an energy harvesting broadcast link," *IEEE Journal on Selected Areas in Communication*, vol. 29, pp. 1712–1731, September 2011.

[50] O. Ozel, J. Yang, and S. Ulukus, "Broadcasting with a battery limited energy harvesting rechargeable transmitter.," in *IEEE WiOpt*, pp. 205–212, 2011.

[51] K. Tutuncuoglu and A. Yener, "Optimum transmission policies for battery limited energy harvesting nodes," *IEEE Transactions on Wireless Communications*, vol. 11, no. 3, pp. 1180–1189, 2012.

[52] O. Ozel, K. Tutuncuoglu, J. Yang, S. Ulukus, and A. Yener, "Resource management for fading wireless channels with energy harvesting nodes.," in *INFOCOM*, pp. 456–460, IEEE, 2011.

[53] J. Yang and S. Ulukus, "Optimal packet scheduling in a multiple access channel with energy harvesting transmitters," *Journal of Communications and Networks, special issue on Energy Harvesting in Wireless Networks*, vol. 14, no. 2, pp. 140–150, 2012.

[54] K. Tutuncuoglu and A. Yener, "Short-term throughput maximization for battery limited energy harvesting nodes," in *IEEE ICC*, pp. 1–5, 2011.

[55] C. Ho and R. Zhang, "Optimal energy allocation for wireless communications powered by energy harvesters," in *IEEE ISIT*, pp. 2368–2372, June 2010.

[56] M. Gatzianas, L. Georgiadis, and L. Tassiulas, "Control of wireless networks with rechargeable batteries," *IEEE Transactions on Wireless Communications*, vol. 9, pp. 581–593, February 2010.

[57] S. Chen, P. Sinha, N. B. Shroff, and C. Joo, "Finite-horizon energy allocation and routing scheme in rechargeable sensor networks," in *IEEE INFOCOM*, pp. 2273–2281, April 2011.

[58] H. Pirsiavash, D. Ramanan, and C. Fowlkes, "Bilinear classifiers for visual recognition," in *IEEE Computer Vision and Pattern Recognition (CVPR)*, pp. 1482–1490, 2010.

[59] N. Tekbiyik and E. Uysal-Biyikoglu, "Energy efficient wireless unicast routing alternatives for machine-to-machine networks," *Journal of Network and Computer Applications (JNCA)*, vol. 34, no. 5, pp. 1587–1614, 2011.

[60] N. Tekbiyik and E. Uysal-Biyikoglu, "Resource management and scheduling in wsns powered by ambient energy harvesting," in *Industrial Wireless Sensor Networks* (V. C. Gungor and G. P. Hancke, eds.), CRC Press, (to appear in March 2013).

[61] N. Tekbiyik, T. Girici, E. Uysal-Biyikoglu, and K. Leblebicioglu, "Proportional fair resource allocation on an energy harvesting downlink," *IEEE Transactions on Wireless Communications*, (Accepted for publication, December 2012).

[62] N. Tekbiyik, E. Uysal-Biyikoglu, T. Girici, and K. Leblebicioglu, "Utility-based time and power allocation on an energy harvesting downlink: The optimal solution," in *27th International Symposium on Computer and Information Sciences (ISCIS)*, pp. 83–91, 2012.

[63] N. Tekbiyik, E. Uysal-Biyikoglu, T. Girici, and K. Leblebicioglu, "An algorithm for proportional-fair downlink scheduling in the presence of energy harvesting," in *27th International Symposium on Computer and Information Sciences (ISCIS)*, pp. 93–101, 2012.

[64] M. Huff, "M2M device networking: components & strategies," tech. rep., MSI Tec, Inc., September 2007.

[65] D. Boswarthick and U. Mulligan, "M2M activities in ETSI." Presentation, July 2009. SCS Conference, Sophia.

[66] M. Starsinic, "System architecture challenges in the home M2M network," in *Long Island Systems Applications and Technology Conference (LISAT)*, pp. 1–7, May 2010.

[67] M. Kim and S. Choi, "Effective M2M gateway selection algorithms for geographical region-based query," in *International Conference on Information and Communication Technology Convergence (ICTC)*, pp. 413 –414, 2010.

[68] ETSI TS 102 689 V1.1.1, "Machine-to-machine communications (m2m); m2m service requirements." Technical Specification, August 2010.

[69] S. K. Tan, M. Sooriyabandara, and Z. Fan, "M2m communications in the smart grid: Applications, standards, enabling technologies, and research challenges.," *International Journal of Digital Multimedia Broadcasting*, vol. 2011, 2011.

[70] R. Lu, X. Li, X. Liang, X. Shen, and X. Lin, "Grs: The green, reliability, and security of emerging machine to machine communications," *IEEE Communications Magazine*, vol. 49, pp. 28–35, april 2011.

[71] W. K. Seah, Z. A. Euy, and H.-P. Tan, "Wireless sensor networks powered by ambient energy harvesting (WSN-HEAP) - survey and challenges," in *1st International Conference on Wireless Communication, Vehicular Technology, Information Theory, and Aerospace and Electronic Systems Technology (Wireless VITAE)*, pp. 1–5, May 2009.

[72] K. Lin, J. Yu, J. Hsu, S. Zahedi, D. Lee, J. Friedman, A. Kansal, V. Raghunathan, and M. Srivastava, "Heliomote: Enabling long-lived sensor networks through solar energy harvesting," in *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems (SenSys)*, (San Diego, CA, USA), p. 309, November 2005.

[73] R. Torah, P. Glynne-Jones, M. Tudor, T. O'Donnell, S. Roy, and S. Beeby, "Self-powered autonomous wireless sensor node using vibration energy harvesting," *Measurement Science and Technology*, vol. 19, October 2008.

[74] L. Hardesty, "Self-powered sensors." MTL Micronets (The annual news magazine of the Microsystems Technology Laboratories), Fall 2010. Massachusetts Institute of Technology.

[75] Z. Fadlullah, M. Fouda, N. Kato, A. Takeuchi, N. Iwasaki, and Y. Nozaki, "Toward intelligent machine-to-machine communications in smart grid," *IEEE Communications Magazine*, vol. 49, pp. 60–65, april 2011.

[76] D. Gislason, *ZigBee Wireless Networking*. Newton, MA, USA: Newnes, 2008.

[77] G. Anastasi, M. Conti, M. D. Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: A survey," *Ad hoc Networks*, vol. 7, pp. 537–568, May 2009.

[78] S. Halawani and A. W. Khan, "Sensors' lifetime enhancement techniques in wireless sensor networks - a survey," *Journal of Computing*, vol. 2, May 2010.

[79] J. Lorincz, A. Capone, and M. Bogarelli, "Energy savings in wireless access networks through optimized network management," in *5th IEEE International Symposium on Wireless Pervasive Computing (ISWPC)*, pp. 449–454, May 2010.

[80] M. A. Marsan and M. Meo, "Energy efficient wireless internet access with cooperative cellular networks," *Computer Networks*, vol. 55, pp. 386–398, February 2011.

[81] M. A. Marsan and M. Meo, "Energy efficient management of two cellular access networks," *ACM SIGMETRICS Performance Evaluation Review*, vol. 37, pp. 69–73, March 2010.

[82] V. Chandrasekhar, J. Andrews, and A. Gatherer, "Femtocell networks: a survey," *IEEE Communications Magazine*, vol. 46, pp. 59–67, 2008.

[83] H. Leem, S. Y. Baek, and D. K. Sung, "The effects of cell size on energy saving, system capacity, and per-energy capacity," in *IEEE WCNC*, pp. 18–21, April 2010.

[84] S. Bhaumik, G. Narlikar, S. Chattopadhyay, and S. Kanugovi, "Breathe to stay cool: Adjusting cell sizes to reduce energy consumption," in *Proceedings of the first ACM SIGCOMM workshop on Green networking*, pp. 41–46, 2010.

[85] K. Son, E. Oh, and B. Krishnamachari, "Energy-aware hierarchical cell configuration: from deployment to operation," in *IEEE INFOCOM 2011 Workshop on Green Communications and Networking*, pp. 289–294, April 2011.

[86] Z. Niu, Y. Wu, J. Gong, and Z. Yang, "Cell zooming for cost-efficient green cellular networks," *IEEE Communications Magazine*, vol. 48, pp. 74–79, November 2010.

[87] ETSI TS 102 706 V1.1.1, "Energy efficiency of wireless access network equipment." Technical Specification, August 2009.

[88] L. Saker, S.-E. Elayoubi, and T. Chahed, "Minimizing energy consumption via sleep mode in green base station," in *WCNC*, pp. 1–6, 2010.

[89] G. Micallef, P. Mogensen, and H. O. Scheck, "Cell size breathing and possibilities to introduce cell sleep mode," in *European Wireless Conference*, pp. 111–115, 2010.

[90] M. Marsan, L. Chiaraviglio, D. Ciullo, and M. Meo, "Optimal energy savings in cellular access networks," in *IEEE International Conference on Communications (ICC) Workshops*, pp. 1–5, June 2009.

[91] E. Oh and B. Krishnamachari, "Energy savings through dynamic base station switching in cellular wireless access networks," in *GLOBECOM*, pp. 1–5, 2010.

[92] R. Litjens and L. Jorguseski, "Potential of energy-oriented network optimisation: Switching off over-capacity in off-peak hours," in *IEEE 21st International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, pp. 1660–1664, September 2010.

[93] Ericsson, "Energy-saving solutions helping mobile operators meet commercial and sustainability goals worldwide." Press Information, June 2008.

[94] Vodafone, "Vodafone group announces commitment to reduce CO_2 emissions by 50%." Press Release, April 2008.

[95] ETSI TR 102 530 V1.1.1, "The reduction of energy consumption in telecommunications equipment and related infrastructure," tech. rep., June 2008.

[96] F. Richter, A. Fehske, and G. Fettweis, "Energy efficiency aspects of base station deployment strategies for cellular networks," in *IEEE 70th Vehicular Technology Conference Fall (VTC 2009-Fall)*, pp. 1–5, September 2009.

[97] B. Badic, T. O'Farrrell, P. Loskot, and J. He, "Energy efficient radio access architectures for green radio: Large versus small cell size deployment," in *IEEE 70th Vehicular Technology Conference Fall (VTC 2009-Fall)*, pp. 1–5, Sept. 2009.

[98] W. Wang and G. Shen, "Energy efficiency of heterogeneous cellular network," in *IEEE 72nd Vehicular Technology Conference Fall (VTC 2010-Fall)*, pp. 1 –5, September 2010.

[99] K. Christensen, P. Reviriego, B. Nordman, M. Bennett, M. Mostowfi, and J. Maestro, "IEEE 802.3az: the road to energy efficient ethernet," *IEEE Communications Magazine*, vol. 48, no. 11, pp. 50–56, 2010.

[100] ARM, "World's most energy-efficient processor from arm targets low-cost mcu, sensor and control markets." Press Release, March 2012.

[101] J. Balfour, W. Dally, D. Black-Schaffer, V. Parikh, and J. Park, "An energy-efficient processor architecture for embedded systems," *IEEE Computer Architecture Letters*, vol. 7, pp. 29–32, Jan. 2008.

[102] V. Fodale and S. Scherer, "Green networking equipment: Who leads and who lags," tech. rep., In-Stat, February 2008.

[103] A. Berl, E. Gelenbe, M. di Girolamo, G. Giuliani, H. de Meer, M.-Q. Dang, and K. Pentikousis, "Energy-efficient cloud computing," *The Computer Journal*, pp. 1–7, August 2009.

[104] E. Gelenbe and S. Silvestri, "Reducing power consumption in wired networks," in *Proceedings of the 24th International Symposium on Computer and Information Sciences (ISCIS)*, (North Cyprus), pp. 292–297, September 2009.

[105] D. Niyato, L. Xiao, and P. Wang, "Machine-to-machine communications for home energy management system in smart grid," *IEEE Communications Magazine*, vol. 49, pp. 53–59, april 2011.

[106] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "Energy-efficient broadcast and multicast trees in wireless networks," *Mobile Network Appl.*, vol. 7, no. 6, pp. 481–492, 2002.

[107] A. Ephremides, "Energy concerns in wireless networks," *Wireless Communications, IEEE*, vol. 9, no. 4, pp. 48–59, 2002.

[108] L. Feeney and M. Nilson, "Investigating the energy consumption of a wireless network interface in an ad hoc networking environment," in *INFOCOM*, 2001.

[109] A. Bachir, D. Barthel, M. Heusse, and A. Duda, "A synthetic function for energy-delay mapping in energy efficient routing," in *WONS*, 2006.

[110] E. Uysal-Biyikoglu, *Energy-Efficient Packet Transmission Scheduling on Wireless Networks*. PhD thesis, Stanford University, 2003.

[111] M. Perillo and W. Heinzelman, "Dapr: A protocol for wireless sensor networks utilizing an application-based routing cost," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1528–1533, 2004.

[112] S. Gupta and A. Hirdesh, "Overview of m2m," tech. rep., University of Florida, 2007.

[113] A. Spyropoulos and C. S. Raghavendra, "Energy efficient communications in ad hoc networks using directional antennas.," in *INFOCOM*, 2002.

[114] J. Aslam, Q. Li, and D. Rus, "Three power-aware routing algorithms for sensor networks," *Wireless Communications and Mobile Computing*, vol. 3, pp. 187–208, 2003.

[115] J. H. Chang and L. Tassiulas, "Routing for maximum system lifetime in wireless ad-hoc networks," in *in 37-th Annual Allerton Conference on Communication, Control, and Computing*, 1999.

[116] T. Girici and A. Ephremides, "Joint routing and scheduling metrics for ad hoc wireless networks," in *in Proc. of the 36th Asilomar Conference on Signals, Systems and Computers, 2002*, pp. 1155–1159, 2002.

[117] A. Avudainayagam, W. Lou, and Y. Fang, "Dear: A device and energy aware routing protocol for heterogeneous ad hoc networks," *J. Parallel Distrib. Comput.*, vol. 63, no. 2, pp. 228–236, 2003.

[118] D. Medhi and K. Ramasamy, *Network Routing: Algorithms, Protocols, and Architectures*. Morgan Kaufmann Publishers, 2007.

[119] J. Park and S. Sahni, "An online heuristic for maximum lifetime routing in wireless sensor networks," *IEEE Trans. Computers*, vol. 55, no. 8, pp. 1048–1056, 2006.

[120] C.-K. Toh, H. Cobb, and D. Scott, "Performance evaluation of battery-life-aware routing schemes for wireless ad hoc networks," in *International Conference on Communications (ICC)*, vol. 9, pp. 2824–2829, 2001.

[121] Q. Li, J. Aslam, and D. Rus, "Online power-aware routing in wireless ad-hoc networks," in *In MOBICOM*, pp. 97–107, 2001.

[122] Q. Li, J. Aslam, and D. Rus, "Hierarchical power-aware routing in sensor networks," in *Proceedings of the DIMACS Workshop on Pervasive Networking*, 2001.

[123] K. Kar, M. S. Kodialam, T. V. Lakshman, and L. Tassiulas, "Routing for network capacity maximization in energy-constrained ad-hoc networks," in *INFOCOM*, 2003.

[124] L. Lin, N. B.Shroff, and R. Srikant, "Asymptotically optimal energy-aware routing for multihop wireless networks with renewable energy sources," *IEEE/ACM Trans. Netw.*, vol. 15, no. 5, pp. 1021–1034, 2007.

[125] A. B. Mohanoor, S. Radhakrishnan, and V. Sarangan, "On energy aware routing in wireless networks.," in *BROADNETS*, pp. 690–697, 2007.

[126] A. B. Mohanoor, S. Radhakrishnan, and V. Sarangan, "Online energy aware routing in wireless networks," *Ad Hoc Netw.*, vol. 7, no. 5, pp. 918–931, 2009.

[127] C.-K. Toh, "Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks," vol. 39, pp. 138–147, 2001.

[128] S. Singh and C. Raghavendra, "Pamas - power aware multi-access protocol with signalling for ad hoc networks," *ACM Computer Communication Review*, vol. 28, pp. 5–26, 1999.

[129] V. Rodoplu and T. H. Meng, "Minimum energy mobile wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 1333–1344, 1999.

[130] S. Singh, M. Woo, and C. S. Raghavendra, "Power-aware routing in mobile ad hoc networks," in *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking, MobiCom*, pp. 181–190, 1998.

[131] N. Vassileva and F. Barcelo-Arroyo, "A survey of routing protocols for maximizing the lifetime of ad hoc wireless networks," *International Journal of Software Engineering and Its Applications*, vol. 2, no. 3, 2008.

[132] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," *IEEE Workshop on Mobile Computing Systems and Applications*, 1999.

[133] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks," in *ACM International Conference on Mobile Computing and Networking, MOBICOM'00*, pp. 56–67, 2000.

[134] J.-H. Chang and L. Tassiulas, "Energy conserving routing in wireless ad-hoc networks," in *INFOCOM*, pp. 22–31, 2000.

[135] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*, pp. 153–181, Kluwer Academic Publishers, 1996.

[136] S. Murthy and J. J. Garcia-Luna-Aceves, "An efficient routing protocol for wireless networks," *Mob. Netw. Appl.*, vol. 1, no. 2, pp. 183–197, 1996.

[137] S. Tarannum, B. Aravinda, L. Nalini, K. Venugopal, and L. Patnaik, "Routing protocol for lifetime maximization of wireless sensor networks," in *International Conference on Advanced Computing and Communications, 2006. ADCOM 2006.*, pp. 401–406, Dec. 2006.

[138] J. hwan Chang and L. Tassiulas, "Maximum lifetime routing in wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 12, pp. 609–619, 2004.

[139] T. Clausen and P. Jacquet, "Optimized Link State Routing Protocol (OLSR)." RFC 3626, October 2003.

[140] S. Mahfoudh and P. Minet, "An energy efficient routing based on OLSR in wireless ad hoc and sensor networks," in *22nd International Conference on Advanced Information Networking and Applications - Workshops, AINAW*, pp. 1253 –1259, 2008.

[141] S. Mahfoudh and P. Minet, "Energy-aware routing in wireless ad hoc and sensor networks," in *Proceedings of the 6th International Wireless Communications and Mobile Computing Conference*, IWCMC, (New York, NY, USA), pp. 1126–1130, ACM, 2010.

[142] M. R. Minhas, S. Gopalakrishnan, and V. Leung, "Fuzzy algorithms for maximum lifetime routing in wireless sensor networks," in *GLOBECOM*, pp. 1–6, 2008.

[143] M. R. Minhas, S. Gopalakrishnan, and V. Leung, "An online multipath routing algorithm for maximizing lifetime in wireless sensor networks," in *Sixth International Conference on Information Technology: New Generations, ITNG*, pp. 581–586, 2009.

[144] D. P. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.

[145] C.-K. Toh, "Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks," *IEEE Communications Magazine*, vol. 39, pp. 138–147, 2001.

[146] C. Pandana and K. Liu, "Near-optimal reinforcement learning framework for-energy aware sensor communications," *IEEE Journal on Selected Areas of Communications*, vol. 23, no. 4, 2005.

[147] M. Younis, M. Youssef, and K. Arisha, "Energy-aware routing in cluster-based sensor networks," in *MASCOTS '02: Proceedings of the 10th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems (MASCOTS'02)*, (Washington, DC, USA), pp. 129–136, 2002.

[148] K. Akkaya and M. Younis, "An energy-aware qos routing protocol for wireless sensor networks," in *Proceedings. 23rd International Conference on Distributed Computing Systems Workshops*, pp. 710–715, 2003.

[149] M. Jafarian and M. Jaseemuddin, "Routing of emergency data in a wireless sensor network for mines," in *Proceedings of IEEE International Conference on Communications,ICC*, pp. 2813–2818, 2008.

[150] D. S. J. D. Couto, D. Aguayo, J. Bicketand, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," in *MOBICOM '03*, pp. 134–146, 2003.

[151] S. C. Ergen and P. Varaiya, "Energy efficient routing with delay guarantee for sensor networks," *Wireless Networks*, vol. 13, no. 5, pp. 679–690, 2007.

[152] M. R. Minhas, S. Gopalakrishnan, and V. C. Leung, "Multiobjective routing for simultaneously optimizing system lifetime and source-to-sink delay in wireless sensor networks," *29th IEEE International Conference on Distributed Computing Systems Workshops*, vol. 0, pp. 123–129, 2009.

[153] Wikipedia, "Matching (graph theory)," 2009. http://en.wikipedia.org/wiki/Matching_(graph_theory)#Maximum_bipartite_matchings.

[154] S. Kwon and N. B. Shroff, "Energy-efficient interference-based routing for multi-hop wireless networks.," in *INFOCOM*, 2006.

[155] S. Kwon and N. B. Shroff, "Energy-efficient sinr-based routing for multihop wireless networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 5, 2009.

[156] H.-Y. Wei, S. Ganguly, R. Izmailov, and Z. J. Haas, "Interference-aware ieee 802.16 wimax mesh networks," vol. 5, pp. 3102–3106, 2005.

[157] S. Kwon and N. B. Shroff, "Unified energy-efficient routing for multi-hop wireless networks.," in *INFOCOM*, pp. 430–438, 2008.

[158] S. Banerjee and A. Misra, "Minimum energy paths for reliable communication in multi-hop wireless networks," in *Proceedings of Mobihoc*, pp. 146–156, 2002.

[159] A. Misra and S. Banerjee, "Mrpc: Maximizing network lifetime for reliable routing in wireless environments," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, vol. 2, pp. 800–806, 2002.

[160] Q. Dong, S. Banerjee, M. Adler, and A. Misra, "Minimum energy reliable paths using unreliable wireless links," in *MobiHoc '05: Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, pp. 449–459, 2005.

[161] A. Srinivas and E. Modiano, "Minimum energy disjoint path routing in wireless ad-hoc networks," in *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking*, pp. 122–133, 2003.

[162] K. Sohrabi, J. Gao, V. Ailawadhi, and G. Pottie, "Protocols for self-organization of a wireless sensor network," *Personal Communications, IEEE*, vol. 7, no. 5, pp. 16–27, 2000.

[163] Y. Zhang, *Wireless sensor network for volcano monitoring*. PhD thesis, IMIT/LCN, 2005.

[164] Y. Lin and Q. Wu, "Energy-conserving dynamic routing in multi-sink heterogeneous sensor networks," in *International Conference on Communications and Mobile Computing, CMC*, pp. 269–273, 2010.

[165] G. Treplan, L. Tran-Thanh, and A. Olah, "Reliable and energy aware routing protocols for wireless sensor networks," in *17th International Conference on Software, Telecommunications and Computer Networks, SoftCOM*, pp. 171–175, 2009.

[166] E.-S. Sung and M. Potkonjak, "Energy balancing routing schemes for low-power wireless networks," *Annual Conference on Communication Networks and Services Research*, vol. 0, pp. 408–415, 2009.

[167] X. S. Zhu, L. Shen, and T. Yum, "Hausdorff clustering and minimum energy routing for wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 2, pp. 990–997, 2009.

[168] W. Yang, W. Liang, J. Luo, and W. Dou, "Energy-aware online routing with qos constraints in multi-rate wireless ad hoc networks," in *Proceedings of the 6th International Wireless Communications and Mobile Computing Conference, IWCMC*, (New York, NY, USA), pp. 721–725, 2010.

[169] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," *Wireless Networks*, vol. 11, no. 4, pp. 419–434, 2005.

[170] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*, pp. 114–128, 2004.

[171] B. Awerbuch, D. Holmer, and H. Rubens, "The medium time metric: High throughput route selection in multi-rate ad hoc wireless networks.," *MONET*, vol. 11, no. 2, pp. 253–266, 2006.

[172] Y. Yang, J. Wang, and R. Kravets, "Designing routing metrics for mesh networks," *Proceedings of the IEEE Workshop on Wireless Mesh Networks (WiMesh). IEEE Press*, 2005.

[173] C. E. Koksal and H. Balakrishnan, "Quality-aware routing metrics for time-varying wireless mesh networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 11, pp. 1984–1994, 2006.

[174] D. Lal, A. Manjeshwar, F. Herrmann, E. Uysal-Biyikoglu, and A. Keshavarzian, "Measurement and characterization of link quality metrics in energy constrained wireless sensor networks," in *GLOBECOM'03*, pp. 446–452, 2003.

[175] V. Gungor, C. Sastry, S. Zhen, and R. Integlia, "Resource-aware and link quality based routing metric for wireless sensor and actor networks," in *IEEE International Conference on Communications, ICC '07*, pp. 3364–3369, 2007.

[176] D. K. Noh and K. Kang, "Balanced energy allocation scheme for a solar-powered sensor system and its effects on network-wide performance," *Journal of Computer and System Sciences*, vol. 77, no. 5, pp. 917–932, 2010.

[177] L. Georgiadis and L. Tassiulas, "Optimal overload response in sensor networks," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2684–2696, 2006.

[178] H.-P. Tan, A. Valera, and W. Koh, "Transmission power control in 2-d wireless sensor networks powered by ambient energy harvesting," in *IEEE 21st International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, pp. 1671–1676, 2010.

[179] R.-S. Liu, P. Sinha, and C. Koksal, "Joint energy management and resource allocation in rechargeable sensor networks," in *INFOCOM*, pp. 1–9, 2010.

[180] S. Rangwala, R. Gummadi, R. Govindan, and K. Psounis, "Interference-aware fair rate control in wireless sensor networks," in *SIGCOMM Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communications*, pp. 63–74, 2006.

[181] M. Gatzianas, L. Georgiadis, and L. Tassiulas, "Control of wireless networks with rechargeable batteries," *IEEE Transactions on Wireless Communications*, vol. 9, no. 2, pp. 581–593, 2010.

[182] L. Georgiadis, M. J. Neely, and L. Tassiulas, "Resource allocation and cross-layer control in wireless networks," *Foundations and Trends in Networking*, vol. 1, no. 1, pp. 1–144, 2006.

[183] M. Gatzianas, L. Georgiadis, and L. Tassiulas, "Asymptotically optimal control of wireless networks with rechargeable batteries." Technical report. Available: http://users.auth.gr/~leonid/public/TechReports/tecreport_rabp.pdf.

[184] S. Reddy and C. R. Murthy, "Duty cycling and power management with a network of energy harvesting sensors," in *4th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pp. 205–208, 2011.

[185] B. Zhang, R. Simon, and H. Aydin, "Maximal utility rate allocation for energy harvesting wireless sensor networks," in *Proceedings of the 14th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM)*, pp. 902–910, 2011.

[186] S. Chen, P. Sinha, N. B. Shroff, and C. Joo, "Finite-horizon energy allocation and routing scheme in rechargeable sensor networks," in *IEEE INFOCOM*, pp. 2273–2281, 2011.

[187] D. T. Lee and F. P. Preparata, "Euclidean shortest paths in the presence of rectilinear barriers," *Networks*, vol. 14, no. 3, pp. 393–410, 1984.

[188] J.-H. Hoepman, "Simple distributed weighted matchings," in *CoRR*, 2004.

[189] L. Chen, S. H. Low, M. Chiang, and J. C. Doyle, "Cross-layer congestion control, routing and scheduling design in ad hoc wireless networks," in *IEEE INFOCOM*, pp. 1–13, 2006.

[190] J. Gorski, F. Pfeuffer, and K. Klamroth, "Biconvex sets and optimization with biconvex functions: a survey and extensions," *Mathematical Methods of Operations Research*, vol. 66, no. 3, pp. 373–407, 2007.

[191] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear Programming Theory and Algorithms*. John Wiley & Sons, 2006.

[192] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[193] D. G. Luenberger and Y. Ye, *Linear and Nonlinear Programming*. Springer Science, 2008.

[194] C.-J. Lin, "Projected gradient methods for nonnegative matrix factorization," *Neural Comput.*, vol. 19, pp. 2756–2779, October 2007.

[195] M. J. D. Powell, "On search directions for minimization," *Mathematical Programming*, vol. 4, pp. 193–201, 1973.

[196] L. Grippo and M. Sciandrone, "On the convergence of the block nonlinear gauss-seidel method under convex constraints," *Operations Research Letters*, vol. 26, pp. 127–136, 2000.

[197] R. E. Wendell and A. Hurter, "Minimization of a non-separable objective function subject to disjoint constraints," *Operations Research*, vol. 24, no. 4, pp. 643–657, 1976.

[198] W. Wiesemann, *Optimization Of Temporal Networks Under Uncertainty*. Springer Heidelberg Dordrecht, 2012.

[199] M. Gorlatova, A. Berstein, and G. Zussman, "Performance evaluation of resource allocation policies for energy harvesting devices," in *IEEE WiOpt*, pp. 189–196, May 2011.

[200] R. Jain, D. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," tech. rep., September 1984.

[201] T. Mahmoodi, V. Friderikos, O. Holland, and H. Aghvami, "Balancing sum rate and tcp throughput in OFDMA based wireless networks," in *IEEE ICC*, pp. 1–6, 2010.

[202] D. Cox, "Prediction by exponentially weighted moving averages and related methods," *Journal of the Royal Statistical Society*, vol. 23, pp. 414–422, 1961.

[203] J. Piorno, C. Bergonzini, D. Atienza, and T. Rosing, "Prediction and management in energy harvested wireless sensor nodes," in *Wireless Vitae*, 2009.

[204] J. L. Crassidis and J. L. Junkins, *Optimal Estimation of Dynamic Systems*. Abingdon: Chapman and Hall, 2004.

[205] N. Sharma, J. Gummeson, D. Irwin, and P. J. Shenoy, "Cloudy computing: Leveraging weather forecasts in energy harvesting sensor systems," in *SECON'10*, pp. 136–144, 2010.

[206] L. Tassiulas and S. Sarkar, "Maxmin fair scheduling in wireless networks," in *IEEE INFOCOM*, pp. 763–772, 2002.

[207] F. Kelly, "Charging and rate control for elastic traffic," *European Transactions on Telecommunications*, vol. 8, pp. 33–37, 1997.

[208] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Transactions on Networking*, vol. 8, no. 5, pp. 556 –567, 2000.

APPENDIX A

THE CONCEPT OF PROPORTIONAL FAIRNESS AND THE REASONING BEHIND THE CHOSEN UTILITY FUNCTION

In its simplest form, fairness is to allocate the same share to all. However, in wireless networks, such a simple idea does not generally make sense. Thus, for wireless networks, fairness is defined in a number of different ways, such as max-min fairness [206], and, proportional fairness [207]. The main focus of this thesis is proportional fairness. The concept of proportional fairness is proposed by Kelly [207], and, generalized by Mo and Walrand [208]. According to [207]: A vector of rates, x, is *proportionally fair* if it is feasible – that is $x \geq 0$ and $A^T x \leq c$ – and, if for any other feasible vector x^*, the aggregate of proportional changes is zero or negative[1]:

$$\sum_i \frac{x_i^* - x_i}{x_i} \leq 0 \tag{A.1}$$

There are many ways to impose the proportional fairness constraint on a network. Using the logarithm utility is one option adopted in this thesis. There are other valid and relevant utility functions, some ensuring fairness as well, such as weighted sum-rate maximization. Howewer, maximizing $\sum log(R)$, where R is the rate, provides proportional fairness by definition. The weighted sum-rate maximization provides only an approximation. This approximation converges to the real value for very long term optimization, keeping track of the rate received by each user in time (Please check the proof provided below). Inherently this is a long-term and online algorithm. On the other hand, we are interested in finite horizon offline optimization and our time slots are determined according to the energy harvesting process,

[1] The published version of this paper has strict inequality in this relation. However, Kelly later provided a corrected version of this paper (For the corrected version, please check http://research.microsoft.com/en-us/events/networkeconomics/kelly_1997.pdf).

and thus, the time slots are not necessarily small. Hence, the chosen log-sum utility function seems to be the best choice for our case.

Proof.

$$max \sum_{n} log(R_n(t)) = max \sum_{n=1} log\left(\alpha R_n(t-1) + (1-\alpha) r_n(t)\right) \tag{A.2}$$

$$= max \sum_{n=1} log(\alpha R_n(t-1)) + log\left(1 + \frac{(1-\alpha) r_n(t)}{\alpha R_n(t-1)}\right) \tag{A.3}$$

Given the average received rate up until the current slot $R_n(t-1)$

$$max \sum_{n} log(R_n(t)) = max \sum_{n=1} log\left(1 + \frac{(1-\alpha) r_n(t)}{\alpha R_n(t-1)}\right) \tag{A.4}$$

Applying the approximation $log(1+x) \approx x$ for small x,

$$max \sum_{n} log(R_n(t)) \approx max \sum_{n=1} \frac{(1-\alpha) r_n(t)}{\alpha R_n(t-1)} \tag{A.5}$$

This can be considered as a weighted total rate maximization, where the weights of each user at time t are,

$$w_n(t) = \frac{(1-\alpha)}{\alpha R_n(t-1)}\ ,\ \forall n \tag{A.6}$$

Weighted sum rate maximization only approximately provides proportional fairness. At each time slot it gives the whole time slot to a single user. The approximation $log(1+x) \approx x$ is only accurate if the number of time slots are very high and time slot durations are very low. Thus, the expression $\sum_n log(R_n(t))$ (the log-sum of user rates) is the true proportional fairness metric. ■

APPENDIX B

PROOFS OF THE THEOREMS, LEMMAS, AND CORROLLARY MENTIONED IN CHAPTER 5

B.1 Proof of Lemma 5.2.1

i) Let us define $l = \sum_{i=1}^{K} c_i h_i$ where h_i is a strictly concave function of p_i, and, $c_i \geq 0$. For l to be concave, it needs to satisy the concavity condition, i.e., $l(\lambda\overline{p_1} + (1-\lambda)\overline{p_2}) \geq \lambda l(\overline{p_1}) + (1-\lambda) l(\overline{p_2})$ where $0 \leq \lambda \leq 1$, for any point $\overline{p_1}, \overline{p_2}$ in the domain of l. Thus, proving that l satisfies the concavity condition completes the proof of part (i). We start by

$$l(\lambda\overline{p_1} + (1-\lambda)\overline{p_2}) = \sum_{i=1}^{K} c_i h_i(\lambda p_{1i} + (1-\lambda)p_{2i}) \geq \sum_{i=1}^{K} c_i(\lambda h_i(p_{1i}) + (1-\lambda)h_i(p_{2i})) \quad \text{(B.1)}$$

$$= \lambda \sum_{i=1}^{K} c_i h_i(p_{1i}) + (1-\lambda) \sum_{i=1}^{K} c_i h_i(p_{2i}) = \lambda l(\overline{p_1}) + (1-\lambda) l(\overline{p_2}) \quad \text{(B.2)}$$

where Eq. (B.1) follows from the strict concavity of h_i function. From the set of equations described above, one can observe that l is a concave function of $\overline{p}$. Note that in Eq. (B.1), equality may happen only if $c_i = 0$ for all i. Hence, if there exists an index j such that $c_i = 0$ for $i = 1, \ldots, j-1, j+1, \ldots, K$ and, $c_i > 0$ for $i = j$, Eq. (B.1) is satisfied with ">". Then, l is strictly concave in $\overline{p_k}$. This completes the proof of part (i).

ii) From part (i) we know that, l is a strictly concave function if Eq. (B.1) is satisfied. Assume that Eq. (B.1) is satisfied and, let there be an increasing concave function f. We need to show that $f(l(\overline{p}))$ is always strictly concave. The proof is as follows:

$$l(\lambda\overline{p_1} + (1-\lambda)\overline{p_2}) > \lambda l(\overline{p_1}) + (1-\lambda) l(\overline{p_2}) \quad \text{(B.3)}$$

$$f(l(\lambda\overline{p_1} + (1-\lambda)\overline{p_2})) > f(\lambda l(\overline{p_1}) + (1-\lambda) l(\overline{p_2})) \quad \text{(B.4)}$$

where Eq. (B.3) and Eq. (B.4) follow from strict concavity of l and increasing property of the f function respectively. By using concavity of the f function, one can further write the following expression

$$f(\lambda l(\overline{p_1}) + (1-\lambda) l(\overline{p_2})) \geq \lambda f(l(\overline{p_1})) + (1-\lambda) f(l(\overline{p_2})) \tag{B.5}$$

Then, combining Eqns. (B.3)-(B.5), leads us to the final result of strict concavity, as follows:

$$f(l(\lambda\overline{p_1} + (1-\lambda)\overline{p_2})) > \lambda f(l(\overline{p_1})) + (1-\lambda) f(l(\overline{p_2})) \tag{B.6}$$

Thus, we conclude that increasing concave functions of strictly concave functions are strictly concave, which completes the proof.

B.2 Proof of Theorem 5.2.2

Since the constraints of the problem are linear, and, maximizing $U(\overline{p})$ is equivalent to minimizing $-U(\overline{p})$, showing that the utility function, $U(\overline{p})$, is strictly concave will be enough to show that Problem 2 can be formulated as a strictly convex optimization problem. We start by checking the concavity of R_{nt}. As R_{nt} is a function of p_t, let $h_t = R_{nt}$ (as in the proof of Lemma 5.2.1's part (i)). The first and second derivatives of h_t are defined as $\frac{\partial h_t}{\partial p_t} = \frac{WL_n/(ln2)}{(1+L_n p_t)}$ and $\frac{\partial^2 h_t}{\partial {p_t}^2} = \frac{-WL_n^2/(ln2)}{(1+L_n p_t)^2}$ respectively. As, W, L_n, and $(1+L_n p_t)^2$ are all positive, $\frac{\partial^2 h_t}{\partial {p_t}^2}$ is definitely negative for all $t = 1, \ldots, K$. Hence, from the second derivative test [191], h_t, thus R_{nt}, is strictly concave in p_t. Furthermore, let $l_n = \sum_{t=1}^{K} \tau_{nt} R_{nt}$ for an arbitrary user n. As all τ_{nt}'s (for $t = 1, \ldots, K$) are nonnegative, and, at least one τ_{nt} is positive for user n, from part (i) of Lemma 5.2.1, l_n is a strictly concave function of $\overline{p}$. Note that this is true for all users, i.e., l_n is strictly concave for all $n = 1, \ldots, N$. Thus, from part (ii) of Lemma 5.2.1, f_n is strictly concave in $\overline{p}$ for all $n = 1, \ldots, N$. The rest of the proof is straight-forward, since the utility function, $U(\overline{p})$, is a positive linear combination of f_n's and thus (from part (i) of Lemma 5.2.1) is strictly concave in $\overline{p}$. Hence, the proof is complete.

B.3 Proof of Lemma 5.2.3

i) Let us define $m = \sum_{i=1}^{K} d_i q_i$ where q_i is an affine function of τ_{ni}, and, $d_i \geq 0$. For m to be affine, it needs to satisfy the affinity condition, i.e., $m(\lambda\overline{\tau_1} + (1-\lambda)\overline{\tau_2}) = \lambda m(\overline{\tau_1}) + (1-\lambda)m(\overline{\tau_2})$, where $0 \leq \lambda \leq 1$ and $\overline{\tau_k} = [\tau_{k,n1} \ \ldots \ \tau_{k,nK}]^T$ is a point on m. Thus, proving that m satisfies the condition completes the proof of part (i). We start by

$$m(\lambda\overline{\tau_1} + (1-\lambda)\overline{\tau_2}) = \sum_{i=1}^{K} d_i q_i(\lambda\tau_{1i} + (1-\lambda)\tau_{2i}) = \sum_{i=1}^{K} d_i(\lambda q_i(\tau_{1i}) + (1-\lambda)q_i(\tau_{2i})) \quad \text{(B.7)}$$

$$= \lambda \sum_{i=1}^{K} d_i q_i(\tau_{1i}) + (1-\lambda) \sum_{i=1}^{K} d_i q_i(\tau_{2i}) = \lambda m(\overline{\tau_1}) + (1-\lambda)m(\overline{\tau_2}) \quad \text{(B.8)}$$

where τ_{1i}, τ_{2i} are the i^{th} entries of the $\overline{\tau_1}$ vector and $\overline{\tau_2}$ vector respectively, and, Eq. (B.7) follows from the "affine" property of the m_i function. From Eq. (B.8), one can clearly observe that m function is an affine function of time variables. Hence, any nonnegative linear combination of affine functions is affine, and, the proof of part (i) is complete.

ii) From part (i) we know that m is an affine function. In this part, we need to prove that an increasing function, s, of m, $s(m(\overline{\tau}))$, is a concave function. Hence, we start by

$$m(\lambda\overline{\tau_1} + (1-\lambda)\overline{\tau_2}) = \lambda m(\overline{\tau_1}) + (1-\lambda)m(\overline{\tau_2}) \quad \text{(B.9)}$$

$$s(m(\lambda\overline{\tau_1} + (1-\lambda)\overline{\tau_2})) = s(\lambda m(\overline{\tau_1}) + (1-\lambda)m(\overline{\tau_2})) \quad \text{(B.10)}$$

where Eq. (B.9) and Eq. (B.10) follow from affine property of m and increasing property of the s function respectively. By using concavity of the s function, one can further write the following expression

$$s(\lambda m(\overline{\tau_1}) + (1-\lambda)m(\overline{\tau_2})) \geq \lambda s(m(\overline{\tau_1})) + (1-\lambda)s(m(\overline{\tau_2})) \quad \text{(B.11)}$$

Then, combining Eqns. (B.9)-(B.11), leads us to the final result of concavity, as follows:

$$s(m(\lambda\overline{\tau_1} + (1-\lambda)\overline{\tau_2})) \geq \lambda s(m(\overline{\tau_1})) + (1-\lambda)s(m(\overline{\tau_2})) \quad \text{(B.12)}$$

Thus, we conclude that increasing concave functions of affine functions are concave, which completes the proof.

B.4 Proof of Theorem 5.2.4

Similar to the proof of Theorem 5.2.2, showing that the utility function, $U(\overline{\tau})$, is concave will be enough to show that Problem 3 can be formulated as a convex optimization problem. We start by checking the concavity of s_n for $n = 1, \ldots, N$. As R_{nt} is a known constant for every $n \in \{1, \ldots, N\}$ and $t \in \{1, \ldots, K\}$, let $d_t = R_{nt}$, and, $q_t = \tau_{nt}$ for an arbitrary user n (as in the proof of Lemma 5.2.3's part (i)). It is well-known that a linear function is an affine function. As q_t is a linear function of τ_{nt} for any user n, q_t is affine. Now, let $m_n = \sum_{t=1}^{K} \tau_{nt} R_{nt}$ for an arbitrary user n. Note that all R_{nt}'s are nonnegative constants known apriori. Thus, from part (ii) of Lemma 5.2.3, s_n is concave in $\overline{\tau}$ for all $n = 1, \ldots, N$. The rest of the proof is straight-forward, since the utility function, $U(\overline{\tau})$, is a nonnegative linear combination of s_n's and thus (from part (i) of Lemma 5.2.1) is concave in $\overline{\tau}$. Hence, the proof is complete.

B.5 Proof of Corollary 1

A function $f : X \times Y \rightarrow \Re$ is called biconvex if $f(x, y)$ is convex in y for fixed $x \in X$ and is convex in x for fixed $y \in Y$ [58]. Since the constraints of the problem are linear, showing that $-U(\overline{\tau}, \overline{p})$ is biconvex will be enough to show that Problem 1 can be formulated as a biconvex optimization problem. $-U(\overline{\tau}, \overline{p})$ is a function of two set of variables; $\overline{\tau}$ and $\overline{p}$. From Theorem 5.2.2, given $\overline{\tau}$, $-U(\overline{p})$ is convex. Similarly, from Theorem 5.2.4, given $\overline{p}$, $-U(\overline{\tau})$ is convex. Hence, $-U(\overline{\tau}, \overline{p})$ is biconvex, which completes the proof.

APPENDIX C

PROOFS OF THE THEOREMS AND LEMMAS MENTIONED IN CHAPTER 6

C.1 Proof of Theorem 6.2.1

The proof is done by contradiction. For any given time allocation $\overline{\tau}$, consider a given power sequence, $\mathcal{P}_C = (p_1, ..., p_{d-1}, p_d, ..., p_K)$, in which the power level decreases at some time, say $d > 1$. In such a case, we can defer some energy, $0 < \Delta \leq p_{d-1}T_{d-1}$, from the $(d-1)^{th}$ slot to the d^{th} slot forming a modified schedule, $\mathcal{P}'_C = (p_1, ..., p'_{d-1}, p'_d, ..., p_K)$, that will not violate the energy causality conditions (as shown in Fig. C.1). Clearly, we can continue this deferral operation until $p'_{d-1} < p'_d$ and still not violate the energy causality conditions. Applying the same method for every possible decrease leads us to a nondecreasing schedule, $\mathcal{P}^{\uparrow}_C = (p'_1, ..., p'_{d-1}, p'_d, ..., p'_K)$, where $p'_1 \leq p'_2 \leq ... \leq p'_K$.

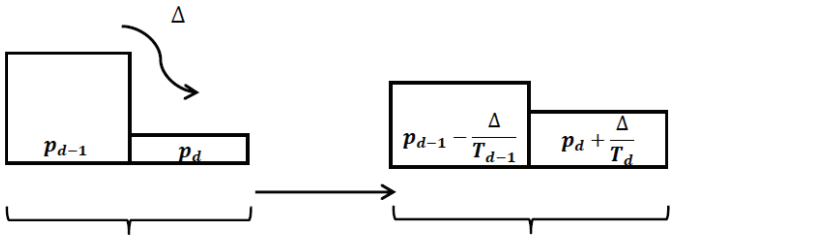

Figure C.1: Maintaining Energy Causality After Energy Deferral

From Lemma 6.2.3, $U(\overline{\tau}, \mathcal{P}_C) = U(\overline{\tau}^{\mathcal{P}^{\uparrow}_C}, \mathcal{P}^{\uparrow}_C)$. Thus, for time allocation $\overline{\tau}^* = \overline{\tau}^{\mathcal{P}^{\uparrow}_C}$, $\mathcal{P}^{\uparrow}_C$ is

optimal. This completes the proof.

C.2 Proof of Lemma 6.2.3

Let, $\overline{R'_n} = \overline{R_n}^{\uparrow}$ where $\overline{R_n}^{\uparrow}$ is as defined in Definition 6.2.2. Note that Eq. (6.7) forces

$$log_2(1 + L_n p'_1) \leq \ldots \leq log_2(1 + L_n p'_l) \leq \ldots \leq log_2(1 + L_n p'_K) \tag{C.1}$$

$$1 + L_n p'_1 \leq \ldots \leq 1 + L_n p'_l \leq \ldots \leq 1 + L_n p'_K \tag{C.2}$$

$$p'_1 \leq \ldots \leq p'_l \leq \ldots \leq p'_K \tag{C.3}$$

Hence, sorting $\overline{R_n}$ in increasing order, forces nondecreasing powers (ordered schedule $\mathcal{P}_C^{\uparrow}$ mentioned previously), which indeed forces all other $\overline{R_i}$ (where $i \in \{1, \ldots, i-1, i+1, \ldots, N\}$) to be sorted in increasing order, to form $\overline{R'_i}$. Now, we have new rates, $\overline{R'_i}$ for all users $i = 1, \ldots, N$. Remember that the utility of a user is defined as in Eq. (6.4). Thus, changing the order of $\overline{R_i}$ vector does not change the value of U_i if the order of $\overline{\tau_i}$ is also changed so that the previous element pairs are matched again. Let us explain this, with an example. Let $R_{i2} < R_{i1}$, $R_{NK} < R_{i2}$, and, $R_{i1} \leq R_{i3} \leq \ldots \leq R_{i(K-1)}$. Then, $\overline{\tau'_i}$, and, $\overline{R'_i}$ vectors are defined as $\overline{R'_i} = [R_{iK}\ R_{i2}\ R_{i1}\ R_{i3}\ \ldots\ R_{i(K-1)}]^T$ and $\overline{\tau'_i} = [\tau_{iK}\ \tau_{i2}\ \tau_{i1}\ \tau_{i3}\ \ldots\ \tau_{i(K-1)}]^T$. Hence, it is straight forward to write that

$$\begin{aligned}
\overline{\tau'_i}^T \overline{R'_i} &= \tau_{iK}R_{iK} + \tau_{i2}R_{i2} + \tau_{i1}R_{i1} + \ldots + \tau_{i(K-1)}R_{i(K-1)} \\
&= \tau_{i1}R_{i1} + \tau_{i2}R_{i2} + \ldots + \tau_{i(K-1)}R_{i(K-1)} + \tau_{iK}R_{iK} \\
&= \overline{\tau_i}^T \overline{R_i}
\end{aligned} \tag{C.4}$$

where $\overline{\tau_i}$ and $\overline{R_i}$ are as defined in Eq. (6.2). As it can be observed, $U_i = U'_i$ as long as $\overline{R'_i} = \overline{R_i}^{\uparrow}$ and $\overline{\tau'_i} = (\overline{\tau_i})^{\overline{R_i}^{\uparrow}}$. Here, $\overline{\tau_i}^{\overline{R_i}^{\uparrow}}$ indicates the $\overline{\tau_i}$ vector ordered according to $\overline{R_i}^{\uparrow}$. Under these circumstances, $U_i = U'_i$ for all $i = 1, \ldots, N$, and, the overall utility does not change, $U = U'$. This completes the proof.

C.3 Proof of Lemma 6.2.4

For the proof of Lemma 6.2.4, we use the KKT optimality conditions. After defining the Lagrangian as in Eq. (6.11), one can list the KKT conditions, for the optimal solution, as follows:

C1: $\nabla_\tau L(\overline{\tau}^*, \overline{\mu}^*, \overline{\lambda}^*) = 0$ which is equivalent to $\left[\frac{\partial L}{\partial \tau_{11}}, \ldots, \frac{\partial L}{\partial \tau_{NK}}\right] = [0, \ldots, 0]$, and thus, leads us to:

$$\frac{\partial L}{\partial \tau_{nt}} = -\frac{\partial U(\overline{\tau}^*)}{\partial \tau_{nt}} + \mu^*_{N(t-1)+n} + \mu^*_{n+NK} - \lambda^*_t = 0 \tag{C.5}$$

where for $\forall n \in \{1, \ldots, N\}$ and $\forall t \in \{1, \ldots, K\}$,

$$\frac{\partial U(\overline{\tau}^*)}{\partial \tau_{nt}} = \frac{1}{ln2} \frac{R_{nt}}{\sum_{i=1}^{K} \tau_{ni} R_{ni}} \tag{C.6}$$

Substituting Eq. (C.6) into Eq. (C.5), we obtain the following set of optimality conditions for $\forall n \in \{1, \ldots, N\}$ and $\forall t \in \{1, \ldots, K\}$:

$$\lambda^*_t = -\frac{1}{ln2} \frac{R_{nt}}{\sum_{i=1}^{K} \tau_{ni} R_{ni}} + \mu^*_{N(t-1)+n} + \mu^*_{n+NK} \tag{C.7}$$

C2: Due to the nonnegativity property of the Lagrange multipliers, the second set of optimality conditions are defined as:

$$\mu^*_j \geq 0 \quad for\ j = 1, \ldots, NK + N \tag{C.8}$$

C3: Similarly, $v_j(\overline{\tau}^*) \geq 0$ for $j = 1, \ldots, NK$ necessitates:

$$\tau^*_{nt} \geq 0 \tag{C.9}$$

$$\sum_{t=1}^{K} \tau_{nt} \geq \epsilon \quad for\ n = 1, \ldots, N,\ and,\ t = 1, \ldots, K \tag{C.10}$$

C4: In order for the equality conditions to be satisfied, we need to have $w_j(\overline{\tau}^*) = 0$ for all j corresponding the equality constraints. This leads us to the following set of conditions for $j = 1, \ldots, K$:

$$\sum_{n=1}^{N} \tau_{nj}^* = T_j \tag{C.11}$$

C5: The last set of conditions are due to $\mu_j^* v_j(\overline{\tau}^*) = 0$ for $j = 1, \ldots, NK + N$ and are defined as:

$$\mu_{N(t-1)+n}^* \tau_{nt}^* = 0 \tag{C.12}$$

$$\mu_{n+NK}^* (\sum_{t=1}^{K} \tau_{nt}^* - \epsilon) = 0 \quad for\ n = 1, \ldots, N,\ and,\ t = 1, \ldots, K \tag{C.13}$$

The optimal time allocation should jointly satisfy the set of conditions described in Eqns. (C.7)-(C.13).

Let, $A_n = \tau_{n1}^* R_{n1} + \tau_{n2}^* R_{n2}$. Then, for the special case, ($N = 2$, $K = 2$), the set of KKT conditions described in Section 6.2.2 reduces to Eqns. (C.14a)-(C.14g).

$$\frac{\partial L}{\partial \tau_{nt}} = \frac{1}{ln2} \frac{R_{nt}}{A_n} + \mu_{2(t-1)+n}^* + \mu_{n+4}^* - \lambda_t^* = 0 \tag{C.14a}$$

$$\mu_i^* \geq 0 \tag{C.14b}$$

$$\tau_{nt}^* \geq 0 \tag{C.14c}$$

$$\tau_{n1}^* + \tau_{n2}^* \geq \epsilon \tag{C.14d}$$

$$\tau_{1t}^* + \tau_{2t}^* = T_t \tag{C.14e}$$

$$\mu_{2(t-1)+n}^* \tau_{nt}^* = 0 \tag{C.14f}$$

$$\mu_{4+n}^* (\tau_{n1}^* + \tau_{n2}^* - \epsilon) = 0 \tag{C.14g}$$

where μ_i's for $i = 1, \ldots, 6$, are the lagrange multipliers and $n = 1, 2$ and $t = 1, 2$. Combining the set of equations described above leads us to the following optimality conditions for the time allocation:

$$\mu^*_{2t-1}\tau^*_{1t} = 0 \tag{C.15a}$$

$$\left(\frac{R_{1t}}{A_1 ln2} - \frac{R_{2t}}{A_2 ln2} + \mu^*_{2t-1}\right)\left(T_t - \tau^*_{1t}\right) = 0 \tag{C.15b}$$

Solving the set of equations in Eq. (C.15), one can obtain the optimal time allocation described in Table C.1. Due to the convex nature of the problem, the solutions presented in Table C.1 represent the global optimums, when the rate improvements of the users, $\Gamma_n = \frac{R_{n2}}{R_{n1}}$, are equal. Note that, as the optimal solutions for all cases depend on the lengths of the slots, when $T_1 \neq T_2$, it is hard to develop a direct relation between power allocation and time allocation. Therefore, we next analyze the case of equal slots to reveal the link between these two. When all slot lengths are equal ($T_1 = T_2 = T$), the optimal time allocation illustrated in Table C.1 reduces to the one presented in Table C.2, posing the desired relation. By inspecting Table C.2, one can observe the properties mentioned in Lemma 6.2.4. All cases are summarized in Table C.2, which completes the proof.

Table C.1: Overall Optimality Conditions for the Special Case of Two Users and Two Slots.

Users' Rate Improvement Relation (User 1 vs. User 2)	Slot Length & Rate Improvement Relation	Slot 1		Slot 2	
		User 1	User 2	User 1	User 2
		τ_{11}	τ_{21}	τ_{12}	τ_{22}
$\Gamma_1 < \Gamma_2$	$\frac{T_1}{T_2} < \Gamma_1$	T_1	0	$\frac{1}{2}\left(T_2 - \frac{T_1}{\Gamma_1}\right)$	$\frac{1}{2}\left(T_2 + \frac{T_1}{\Gamma_1}\right)$
	$\frac{T_1}{T_2} = \Gamma_1$	T_1	0	0	T_2
	$\Gamma_1 < \frac{T_1}{T_2} < \Gamma_2$	T_1	0	0	T_2
	$\frac{T_1}{T_2} = \Gamma_2$	T_1	0	0	T_2
	$\frac{T_1}{T_2} > \Gamma_2$	$\frac{1}{2}(T_1 + T_2\Gamma_2)$	$\frac{1}{2}(T_1 - T_2\Gamma_2)$	0	T_2
$\Gamma_1 = \Gamma_2$	$\frac{T_1}{T_2} < \Gamma_1 = \Gamma_2$	T_1	0	$\frac{1}{2}\left(T_2 - \frac{T_1}{\Gamma_1}\right)$	$\frac{1}{2}\left(T_2 + \frac{T_1}{\Gamma_1}\right)$
		0	T_1	$\frac{1}{2}\left(T_2 + \frac{T_1}{\Gamma_2}\right)$	$\frac{1}{2}\left(T_2 - \frac{T_1}{\Gamma_2}\right)$
	$\frac{T_1}{T_2} = \Gamma_1 = \Gamma_2$	T_1	0	0	T_2
		0	T_1	T_2	0
	$\frac{T_1}{T_2} > \Gamma_1 = \Gamma_2$	$\frac{1}{2}(T_1 - T_2\Gamma_1)$	$\frac{1}{2}(T_1 + T_2\Gamma_1)$	T_2	0
		$\frac{1}{2}(T_1 + T_2\Gamma_2)$	$\frac{1}{2}(T_1 - T_2\Gamma_2)$	0	T_2
$\Gamma_1 > \Gamma_2$	$\frac{T_1}{T_2} < \Gamma_2$	0	T_1	$\frac{1}{2}\left(T_2 + \frac{T_1}{\Gamma_2}\right)$	$\frac{1}{2}\left(T_2 - \frac{T_1}{\Gamma_2}\right)$
	$\frac{T_1}{T_2} = \Gamma_2$	0	T_1	T_2	0
	$\Gamma_2 < \frac{T_1}{T_2} < \Gamma_1$	0	T_1	T_2	0
	$\frac{T_1}{T_2} = \Gamma_1$	0	T_1	T_2	0
	$\frac{T_1}{T_2} > \Gamma_1$	$\frac{1}{2}(T_1 - T_2\Gamma_1)$	$\frac{1}{2}(T_1 + T_2\Gamma_1)$	T_2	0

Table C.2: Overall Optimality Conditions for the Special Case of Two Users and Two Slots ($T_1 = T_2$): Categorized according to the relation between the powers allocated in the first and second slots. For a given power allocation, the optimal time allocation differs according to the relation between the rate improvements of the users.

Power Relation (Slot 1 vs. Slot 2)	Users' Rate Improvement Relation (User 1 vs. User 2)	Slot 1		Slot 2		Utility
		User 1	User 2	User 1	User 2	
		τ_{11}	τ_{21}	τ_{12}	τ_{22}	
$p_1 < p_2$	$\Gamma_1 < \Gamma_2$	T	0	$\frac{T}{2}\left(1-\frac{1}{\Gamma_1}\right)$	$\frac{T}{2}\left(1+\frac{1}{\Gamma_1}\right)$	$log_2\left(\frac{R_{22}}{R_{12}}(R_{11}+R_{12})^2\right)+2log_2\left(\frac{T}{2}\right)$
	$\Gamma_1 = \Gamma_2$	T	0	$\frac{T}{2}\left(1-\frac{1}{\Gamma_1}\right)$	$\frac{T}{2}\left(1+\frac{1}{\Gamma_1}\right)$	$log_2((R_{11}+R_{12})(R_{21}+R_{22}))+2log_2\left(\frac{T}{2}\right)$
		0	T	$\frac{T}{2}\left(1+\frac{1}{\Gamma_2}\right)$	$\frac{T}{2}\left(1-\frac{1}{\Gamma_2}\right)$	$log_2((R_{11}+R_{12})(R_{21}+R_{22}))+2log_2\left(\frac{T}{2}\right)$
	$\Gamma_1 > \Gamma_2$	0	T	$\frac{T}{2}\left(1+\frac{1}{\Gamma_2}\right)$	$\frac{T}{2}\left(1-\frac{1}{\Gamma_2}\right)$	$log_2\left(\frac{R_{12}}{R_{22}}(R_{21}+R_{22})^2\right)+2log_2\left(\frac{T}{2}\right)$
$p_1 = p_2$	$\Gamma_1 < \Gamma_2$	T	0	0	T	$log_2(R_{11}R_{22})+2log_2(T)$
	$\Gamma_1 = \Gamma_2$	T	0	0	T	$log_2(R_{11}R_{22})+2log_2(T)$
		0	T	T	0	$log_2(R_{11}R_{22})+2log_2(T)$
	$\Gamma_1 > \Gamma_2$	0	T	T	0	$log_2(R_{12}R_{21})+2log_2(T)$
$p_1 > p_2$	$\Gamma_1 < \Gamma_2$	$\frac{T}{2}(1+\Gamma_2)$	$\frac{T}{2}(1-\Gamma_2)$	0	T	$log_2\left(\frac{R_{11}}{R_{21}}(R_{21}+R_{22})^2\right)+2log_2\left(\frac{T}{2}\right)$
	$\Gamma_1 = \Gamma_2$	$\frac{T}{2}(1-\Gamma_2)$	$\frac{T}{2}(1+\Gamma_2)$	T	0	$log_2((R_{11}+R_{12})(R_{21}+R_{22}))+2log_2\left(\frac{T}{2}\right)$
		$\frac{T}{2}(1+\Gamma_1)$	$\frac{T}{2}(1-\Gamma_1)$	0	T	$log_2((R_{11}+R_{12})(R_{21}+R_{22}))+2log_2\left(\frac{T}{2}\right)$
	$\Gamma_1 > \Gamma_2$	$\frac{T}{2}(1-\Gamma_1)$	$\frac{T}{2}(1+\Gamma_1)$	T	0	$log_2\left(\frac{R_{21}}{R_{11}}(R_{11}+R_{12})^2\right)+2log_2\left(\frac{T}{2}\right)$

CURRICULUM VITAE

Neyre Tekbıyık Ersoy

Department of Electrical and Electronics Engineering,
Middle East Technical University,
Ankara, 06800 Turkey,
E-mail: ntekbiyik@eee.metu.edu.tr

EDUCATION

Ph.D. with High Honours (2008-2012), Department of Electrical and Electronics Engineering, Middle East Technical University, Ankara, Turkey.

- Ph.D. Thesis Defense. with Success (December 2012).
- Thesis: “Efficient Resource Allocation In Energy Harvesting Wireless Networks”. Advisor: Assoc. Prof. Dr. Elif Uysal-Bıyıkoğlu. The thesis investigates the proportional fairness based utility maximization problem in energy harvesting wireless networks, where the battery of the transmitter/ base station gets recharged periodically. The main contributions of this thesis are; an optimal offline algorithm, close-to-optimal offline heuristic algorithms, and, a prediction based online algorithm.
- Doctoral Qualifying Examination. with Success (2010).

M.Sc. with High Honours (2005-2007), Department of Electrical and Electronics Engineering, Eastern Mediterranean University, Famagusta, North Cyprus.

B.Sc. with High Honours (2001-2005), Department of Electrical and Electronics Engineering, Eastern Mediterranean University, Famagusta, North Cyprus.

PROFESSIONAL EXPERIENCE

Department of Electrical and Electronics Engineering, Middle East Technical University

April 11 - to date: Research assistant in the TUBITAK project “Principles and Experimental Implementation Toward Energy-Efficient Design of Wireless Networks”.

March 08 - April 11: Research assistant in the TUBITAK project “Design of Minimum Energy High Performance Wireless Communication Networks: Inter-layer Optimization and Algorithms”.

Department of Electrical and Electronics Engineering, Eastern Mediterranean University

September 07 - February 08: Research Assistant (during PhD)

September 05 - September 07: Research Assistant (during M.Sc.)

Cyprus Turkish Electricity Authority (KIBTEK)

August 04 - September 04: Intern

COMPUTER SKILLS

MATLAB, Latex, MS Word, MS Excell, MS Power Point.

PUBLICATIONS

Book Chapters

- N. Tekbiyik and E. Uysal-Biyikoglu, “Resource management and scheduling in WSNS powered by ambient energy harvesting,” *in Industrial Wireless Sensor Networks* (V. C. Gungor and G. P. Hancke, eds.), CRC Press, (to appear in March 2013).

Journal Papers

- N. Tekbiyik and E. Uysal-Biyikoglu, “Energy efficient wireless unicast routing alternatives for machine-to-machine networks,” *Journal of Network and Computer Applications (JNCA)*, vol. 34, no. 5, pp. 1587-1614, 2011.
- N. Tekbiyik, T. Girici, E. Uysal-Biyikoglu, and K. Leblebicioglu, “Proportional fair resource allocation on an energy harvesting downlink,” *IEEE Transactions on Wireless Communications*, (Accepted for publication, December 2012).

Conference Papers

- N. Tekbiyik, E. Uysal-Biyikoglu, T. Girici, and K. Leblebicioglu, “Utility-based time and power allocation on an energy harvesting downlink: The optimal solution,” *in 27th International Symposium on Computer and Information Sciences (ISCIS)*, pp. 83-91, 2012.
- N. Tekbiyik, E. Uysal-Biyikoglu, T. Girici, and K. Leblebicioglu, “An Algorithm for Proportional-Fair Downlink Scheduling in the Presence of Energy Harvesting,” *in 27th International Symposium on Computer and Information Sciences (ISCIS)*, pp. 93-101, 2012.

Theses

- N. Tekbiyik, “Efficient Resource Allocation In Energy Harvesting Wireless Networks”, PhD Thesis, Middle East Technical University, Ankara, Turkey, Dec. 2012.
- N. Tekbiyik, “Closed-Loop Power Control with Fixed Step Size in DS-CDMA Cellular Systems”, M.Sc. Thesis, Eastern Mediterranean University, North Cyprus, Sept. 2007.
- N. Tekbiyik and H. S. Tozkoparan, “Embedded Zerotree Wavelet Compression”, Undergraduate Project Report, Eastern Mediterranean University, North Cyprus, Jun. 2005.