Old Dominion University
ODU Digital Commons

Computer Science Theses & Dissertations | Computer Science

Summer 2015

Energy Harvesting-Aware Design for Wireless Nanonetworks

Shahram Mohrehkesh
Old Dominion University, smohr003@odu.edu

Follow this and additional works at: https://digitalcommons.odu.edu/computerscience_etds

 Part of the Computer Sciences Commons, and the Nanoscience and Nanotechnology Commons

Recommended Citation

Mohrehkesh, Shahram. "Energy Harvesting-Aware Design for Wireless Nanonetworks" (2015). Doctor of Philosophy (PhD), Dissertation, Computer Science, Old Dominion University, DOI: 10.25777/nabb-dn67
https://digitalcommons.odu.edu/computerscience_etds/2

This Dissertation is brought to you for free and open access by the Computer Science at ODU Digital Commons. It has been accepted for inclusion in Computer Science Theses & Dissertations by an authorized administrator of ODU Digital Commons. For more information, please contact digitalcommons@odu.edu.

ENERGY HARVESTING-AWARE DESIGN FOR WIRELESS NANONETWORKS

by

Shahram Mohrehkesh
B.S. August 2003, AmirKabir University of Technology
M.S. August 2006, Iran University of Science and Technology

A Dissertation Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

COMPUTER SCIENCE

OLD DOMINION UNIVERSITY
August 2015

Approved by:

_____
Michele C. Weigle (Director)

_____
Kurt Maly (Member)

_____
Stephan Olariu (Member)

_____
Tamer Nadeem (Member)

_____
Norou Diawara (Member)

_____
Sajal K. Das (Member)

ABSTRACT

ENERGY HARVESTING-AWARE DESIGN FOR WIRELESS NANONETWORKS

Shahram Mohrehkesh
Old Dominion University, 2015
Director: Dr. Michele C. Weigle

Nanotechnology advancement promises to enable a new era of computing and communication devices by shifting micro scale chip design to nano scale chip design. Nanonetworks are envisioned as artifacts of nanotechnology in the domain of networking and communication. These networks will consist of nodes of nanometer to micrometer in size, with a communication range up to 1 meter. These nodes could be used in various biomedical, industrial, and environmental monitoring applications, where a nanoscale level of sensing, monitoring, control and communication is required. The special characteristics of nanonetworks require the revisiting of network design. More specifically, nanoscale limitations, new paradigms of THz communication, and power supply via energy harvesting are the main issues that are not included in traditional network design methods. In this regard, this dissertation investigates and develops some solutions in the realization of nanonetworks. Particularly, the following major solutions are investigated. (I) The energy harvesting and energy consumption processes are modeled and evaluated simultaneously. This model includes the stochastic nature of energy arrival as well as the pulse-based communication model for energy consumption. The model identifies the effect of various parameters in this joint process. (II) Next, an optimization problem is developed to find the best combination of these parameters. Specifically, optimum values for packet size, code weight, and repetition are found in order to minimize the energy consumption while satisfying some application requirements (i.e., delay and reliability). (III) An optimum policy for energy consumption to achieve the maximum utilization of harvested energy is developed. The goal of this scheme is to take advantage of available harvested energy as much as possible while satisfying defined performance metrics. (IV) A communication scheme that tries to maximize the data throughput via a distributed and scalable coordination while avoiding the collision among neighbors is the last problem to be investigated. The goal is to design an energy harvesting-aware and distributed mechanism that could coordinate

data transmission among neighbors. (V) Finally, all these solutions are combined together to create a data link layer model for nanonodes. We believe resolving these issues could be the first step towards an energy harvesting-aware network design for wireless nanosensor networks.

Copyright, 2015, by Shahram Mohrehkesh, All Rights Reserved.

ACKNOWLEDGEMENTS

This dissertation would not have been possible without the advice and the help of several individuals who in one way or another contributed and extended their valuable assistance in the preparation and completion of this study.

First and foremost, I would like to express my sincere gratitude to my advisor Dr. Michele C. Weigle for the continuous support of my Ph.D. study and research, for her patience, motivation, enthusiasm, and immense knowledge. She helped me in all moments of my research and creating of this dissertation. I could not have imagined having a better advisor and mentor for my Ph.D study.

My sincere thanks also goes to Professor Stephan Olariu for the invaluable feedbacks, support, motivation and guidance. Special thanks to Prof. Sajal K. Das for being a great mentor and for teaching me how to improve my work.

Also, I would like to thank the rest of my dissertation committee: Prof. Kurt Maly and Dr. Tamer Nadeem, and Dr. Norou Diawara for their encouragement and insightful comments. I would also like to thank Prof. Nikos P. Chrisochoides, who was a great mentor for me since the first day I joined the ODU.

My colleagues in the intelligent network System (iNets) Lab, as well as center for real time computing (CRTC), and the staff of the Department of Computer Science, thanks for you help and support.

Finally, I would like to thank my family. My special thanks to my father, Hassan Mohrehkesh, as well as my mother, Fatemeh Norouzi, for supporting me throughout my life. My sister, Shirin Mohrehkesh, and my brother, Shahab Mohrehkesh, thank you for accompanying me in this journey.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The last decade has witnessed significant advances in nanotechnology as one of the promising approaches to overcome the limitations in downscaling microelectronics. The development of materials at nano scales now enables the creation of nanomachines. It is envisioned [1, 2] that in coming years fabrication of biosensors, nano-memories, and other nanocomponents together will create devices with sensing, computing, actuating, and communication at nanoscale, called nanosensor motes, or nanonodes.

There are exciting sensing applications among others [1, 2] that could be enabled by the deployment of nanonodes. They will enable molecular level monitoring of chemicals or bacteria as well as intra-body drug delivery systems [2]. In addition, the integration of nanonodes in every single object will allow the networking of almost everything in our daily life, from cooking utensils to every element in our offices.

Similar to adding communication to traditional sensors, which opened the door for innovative applications such as remote environmental monitoring, allowing nanonodes to communicate will enable application development of nanosensors. One simple application of nanosensors would be to monitor the level of various ions such a glucose in blood constantly. As shown in Figure 1, nanosensors could sense and monitor conditions inside the body. There are many situations in which communication among these nodes is required. The simplest scenario is to transfer their measurements, e.g., the level of glucose, through a network to a micro-gateway. Next, information can be transmitted to the micro domain via that micro-gateway[1].

Methods for communication among traditional sensors are not applicable at nanoscale for several reasons. First, the nanoscale properties of these nodes limit the complexity level of schemes and protocols that could be run on a nanonode. Second, due to their size limitation, these nodes rely on energy harvested from the

[1]Since the energy that is used for communication is on the order of picojoules, power intensity would not have thermal effects on human tissues [3] and could provide communication inside the body which is composed of 53% water [4] and other molecules structures, including connective tissue, fats, protein, apatite (in bones), carbohydrates and DNAs. More details will be described in Section 2.3.2

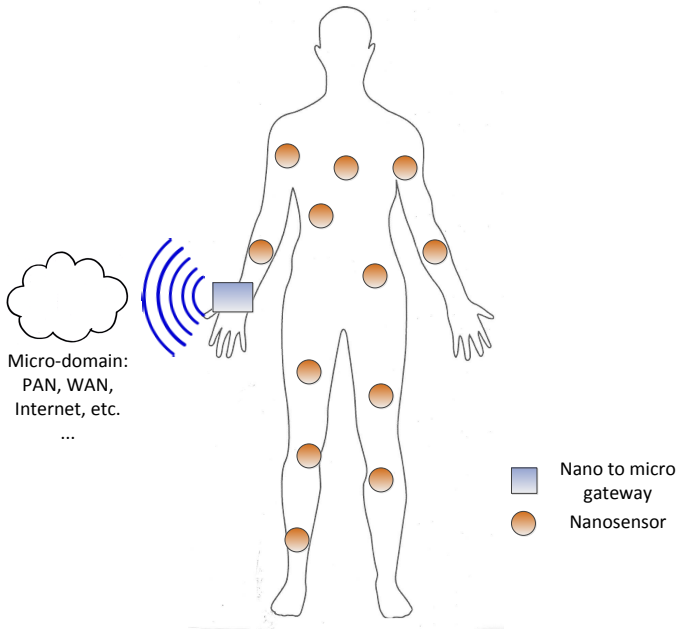


FIG. 1: Nanosensors Inside Body, Communication with Outside Through a Gateway.

environment and stored in a ultra-nanocapacitor [5]. Because energy will be so precious, a new network design that considers both an energy-efficient communication model and the realities of energy harvesting (e.g., stochastic arrival, variable spatio-temporal properties) is required. Finally, it is envisioned that electromagnetic communication for nanonodes operates in the 0.1-10 THz band, which is different from traditional wireless carrier based communication model. Therefore, communication protocols among nanonodes should be revisited for properties of pulse-based communication in Terahertz band. In conclusion, many questions need to be addressed to design wireless nanonetworks, which involve all of the new challenges of nanonodes, i.e., nanoscale limitation, energy harvesting process, and THz communication.

1.1 STATE-OF-ART OF ENERGY HARVESTING-BASED NANONETWORKS

There are many issues to be answered before nanonetworks can be realized [1, 6, 7, 8]. Figure 2 illustrates the state of the art in electromagnetic nanonetworks along various layers. Although a network of nanonodes does not need to have this protocol stack, here for the lack of a better categorization method, we use this

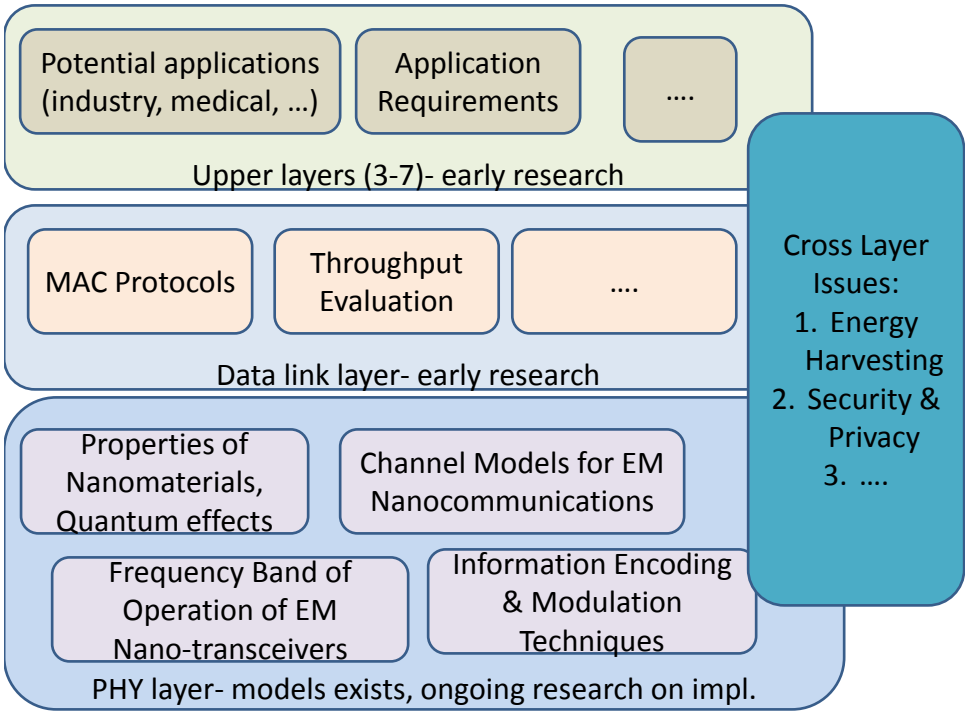


FIG. 2: State of Art for Electromagnetic Nanonetworks.

structure to describe existing work in nanonetworks. At the physical layer, properties of nanomaterials are known and the modeling of wireless communication in the THz band has been studied [9, 10, 11, 12]. However, the real implementation of a nanoantenna is ongoing research. At the data link layer, some initial work such as [5, 13] has been done, including work on energy harvesting design. Dressler and Kargl [8] investigated the security and privacy challenges in nanonetworks. Some other issues about various layers and applications of nanonetworks in the releam of Internet of Nanothings have been studied by Balasubramaniam and Kangasharju [7]. The focus of this dissertation will be on energy harvesting issues, the data link layer and their interaction with each other.

1.1.1 ENERGY HARVESTING

Energy harvesting plays the main role in the realization of nanonetworks. Due to the limited size of nanonodes as well as their applications in environments with no light or heat, new sources of energy, such as blood sugar, which is harvested by biofuel cells [14], electrical differences in the inner ear [15], and ambient vibration[2], are

introduced as the main methods for energy harvesting. A piezoelectric nanogenerator prototype [16] has shown promising results in harvesting energy from vibration at nano scale. The amount of harvested energy depends on the vibration rate. This means that the variation in the vibration rate will result in a stochastic model for available energy for a node at different times and different locations. Moreover, energy storage in a nanobattery/ultracapacitor is not a linear process. Therefore, the first issue is to understand and model the energy harvesting process where the stochastic and nonlinear behavior of harvesting is included.

1.1.2 COMMUNICATION

Electromagnetic communication in the 0.1-10 THz band is proposed [12] as the main communication method for nanonodes due to their limited energy budget and nano scale properties. In this frequency range, pulse-based communication is used rather than the regular carrier-based communication of traditional wireless networks. The pulses could provide communication at millimeter to one meter scale. Among the possible pulse modulation methods, Rate Division Time Spread On-Off Keying (RD TS-OOK) [12] is speculated as the simplest method. In this method, a logical `1` is transmitted as a femto-second long pulse, and a logical `0` is transmitted as silence. The duration of each pulse is T_p and the time between two symbols is T_s, producing a symbol rate of $\beta = \frac{T_s}{T_p}$. Since silence does not consume energy, any scheme that could produce fewer `1`s is preferred. For example, using *code weight* [10] has been proposed [5] to reduce energy consumption. The code weight basically reduces the number of `1`s by adding extra bits so that data is coded in a way that a fewer number of `1`s are present in the coded bits. This results in lower energy in transmission and higher energy in reception. The reception of either a `0` or `1` costs the same energy, so sending more bits results in higher energy consumption for the receiver. Energy savings could happen only if the energy for the reception is lower than the transmission, which is the typical case in wireless transmission [12] and [5]. Not only should the optimum value for this trade-off be identified, but also other methods of coding information regarding the limitation of nanonodes are of interest.

In addition to coding and modulation of pulses, the transmission of pulses in the THz band encounters special channel behaviors. While the probability of collision between symbols is low due to the fact that there can be no collisions for `0` symbols (silences) and that the length of T_s is much longer than T_p (typically 1000 times

larger), the probability of path loss (i.e., absorption of pulses) exists. This probability increases exponentially with a growth in the communication distance or in the environmental molecular absorption conditions. For example, in a 10 mm communication distance with 10% water vapor, the transmission will face 10^{-4} bit error rate [5, 12].

1.1.3 NETWORKING

Nanosensor networks inherit some of the known challenges in sensor networks, such as unknown and not fully manageable topologies, large scale networks (i.e., thousands of nodes), and difficult central management. A new communication mechanism, as well as energy harvesting based nanonodes, make all the networking challenges more difficult to address. In other words, any solution such as a medium access method should be topology-independent, decentralized, scalable, and energy harvesting-aware.

1.2 RESEARCH OBJECTIVES

Due to the characteristics of nanonetworks, there are several challenges in the realization of this new networking paradigm that require novel solutions and even to rethink some well-established concepts in communication and network theory. These challenges range from the design of novel nanoantennas to the development of new communication models and protocols for nanonodes.

Because of the importance of energy harvesting for nanonodes, we focus mainly on the development of communication and network models with consideration of energy harvesting. Therefore, the goal of this work is to design, develop, and evaluate energy harvesting-aware solutions for the realization of nanonetworks. In this dissertation, we particularly focus on addressing the following problems:

- Modeling the energy harvesting and consumption of nanonodes
- Optimizing the energy consumption with regards to energy harvesting
- Maximizing the utilization of harvested energy
- Enable data link layer communication among nanonodes

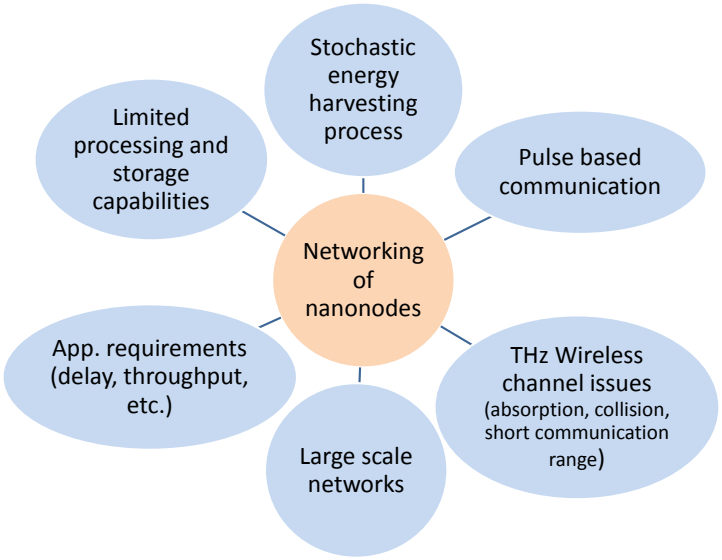


FIG. 3: Challenges in the Networking of Nanonodes.

1.3 THESIS STATEMENT AND CONTRIBUTIONS

The main challenges in the networking of nanonodes are illustrated in Figure 3. A network design that includes all the characteristics of nanonetworks is required. The main network design principle is energy harvesting-aware. With this approach, we can write the thesis statement as follows.

Thesis Statement: A network of resource-limited THz operating nanonodes requires distributed and energy harvesting-aware data link layer mechanisms to increase the data rate of successful communication between nanonodes. Maximizing the utilization of harvested energy, minimizing the amount of energy consumption, and a distributed medium access method for communication among neighbor nanondes make this increase possible.

To address this statement, we took the following steps:

- We started with a general and simple model to understand the combined process of energy harvesting and energy consumption. This process showed that many parameters (e.g., packet size, number of neighbors, rate of energy consumption, code weight) can affect the process significantly, which results in an inefficient performance in energy utilization for communication (Chapter 3 and [17]).

- We then evaluated how the packet should be formed to minimize energy consumption while satisfying other functions, such as delay and reliability, simultaneously. We found that a combination of packet size, code weight and repetition can meet the requirements of these functions. We developed a general multi-objective optimization problem that can be customized by nanonodes for their application environment (Chapter 4 and [17]).

- Next, we evaluated the optimum utilization of harvested energy in relation to the energy harvesting problem. We designed a Markov decision process model for this purpose, where we include the number of receptions and transmissions per timelsot for each nanonode. Our model not only includes the utilization of energy, but also avoids going to *full* and *out* of energy states. We showed the minimum capacity requirement for energy storage to avoid going to these states. Our model is general enough to be used for any harvesting model, as well as both linear and nonlinear energy storage. We also developed a heuristic model that can perform close to the optimal solution. The heuristic solution adapts the energy consumption rate based on the level of available energy in storage (Chapter 5 and [18, 19, 20]).

- To enable communication among nanonodes, we developed a receiver-initiated MAC protocol (RIH-MAC) that can operate both in centralized and distributed topologies. RIH-MAC is distributed and thus is scalable. In the centralized solution, RIH-MAC uses a probabilistic approach to coordinate the communication between nanonodes and a central node, called a nanocontroller. In the distributed topologies, we used a distributed edge-coloring method to determine the channel access mechanism. RIH-MAC can also adapt to various energy harvesting rates. Combining our energy utilization model (Chapter 5) with our prediction-based method for coordinating the energy consumption, called CECS, we could achieve a higher performance in terms of energy efficiency in communication with neighbors (Chapter 6 and [21]).

- Finally, we combined all the protocols and schemes to present their efficiency as energy-harvesting-aware solutions for networking among nanonodes. We simulated a simple application for medical monitoring by nanosensors on and inside the body. These nanosensors measure parameters, such as glucose, and transfer the measurements to the micro domain for further processing. We

showed that our scheme can provide these measurements with low delay even in low energy harvesting rates (Chapter 7 and [22]).

In this work, we propose several algorithms and mechanisms to provide energy harvesting-aware communication between nanonodes. Particularly, our main **contributions** are:

- *Optimum energy consumption for packets*: We developed a model that include both stochastic energy harvesting and energy consumption processes together. The model reveals the parameters that affect these two processes. Then, we developed an optimization model to identify the optimal combination of these parameters to satisfy several metrics of energy consumption, delay, error rate and throughput, simultaneously.

- *Optimal Policy for Energy Consumption*: We developed a model that finds the optimal policy for energy consumption where the stochastic properties of energy harvesting is included. This optimal policy includes maximizing the utilization of available energy while avoiding the over-consumption of energy. Moreover, this model enables us to analyze the process of energy consumption and harvesting to understand the energy storage capacity requirements.

- *Data Link Layer Communication Model*: We developed a receiver-initiated MAC protocol for nanonodes. This energy harvesting-aware protocol operates in both centralized and distributed topologies of nanonodes.

1.4 OUTLINE

The rest of this work is organized as follows.

- Chapter 2 introduces nanonodes and nanonetworks in more detail. Applications, communication models, and networking of nanonodes are reviewed. Moreover, a survey of the energy harvesting state-of-art is provided. We also highlight the differences between our contribution and the most similar related work.

- Chapter 3 introduces our model for the joint energy harvesting and consumption process. The model enables us to identify the main parameters that affect this joint process.

- Chapter 4 describes our multi-objective model to optimize several functions in packet design for nanonetworks.
- Chapter 5 defines our Markov decision process (MDP) which maximizes the utilization of harvested energy. In addition to the MDP model, several other simple and heuristic models are evaluated.
- Chapter 6 introduces our receiver-initiated harvesting-aware MAC protocol. The protocol operates in both centralized and distributed topologies of nanonodes while considering an energy harvesting-aware and distributed solution.
- Chapter 7 demonstrates the use of the developed algorithms and protocols in some basic nanonetwork applications.
- Chapter 8 summarizes the work and discusses some open questions for future work.

CHAPTER 2

BACKGROUND AND RELATED WORK

2.1 NANONODES AND NANONETWORKS

Downscaling of electronic devices has always been a goal since the introduction of transistors in 1950. The trend of downscaling is now approaching the nanoscale as illustrated in Figure 4. After 2010, several research directions have been created to shift the design of microchips to nanoscale [23]. Since each silicon atom is 0.3 nm size, a gate size that is composed of at least 40 atoms would be at least 12 nm, not including the heating and current leakage problems. Due to limitations in silicon technology, researchers are investigating new materials and new fabrication methods. Short term solutions, such as using silicon/germanium helium material or deep-ultraviolet excimer laser photolithography, as well as long term solution such as nanotechnology, quantum computing, and DNA computing, are topics of ongoing research to enable nanoscale design. For example, Intel currently is trying to move from 22 nm transistors to 14 nm transistors. Although it was predicted in 2001 [24] that for the next 20 years, semiconductor modifications will still play a major role, technology and materials such as carbon nanotubes (Figure 5) and graphene nanoribbons (Figure 6) are the promising new enablers in design of nanochips. These materials in combination with advanced manufacturing techniques, such as electron beam lithography [25] enables the introduction of nanomachines, such as nano-electromechanical systems (NEMS) components, e.g., nanomemory, nanosensor, etc. [26]. However, the fabrication and assembly of these nanomachines is still at an early stage. In contrast to the top-down approach of nanomachine fabrication, a bottom-up approach or a hybrid approach is also envisioned as the method for producing nanonodes (Figure 7).

Nanotechnology advancement promises a significant rise in small scale communication. The reception of radio waves through a nanoantenna by using nanotubes [27] and the development of graphene-based nanoantennas [11] has enlightened the vision for the feasibility of nanonode production with communication capabilities. It is envisioned [2] that in the coming years, nanosensor nodes at nano to millimeter

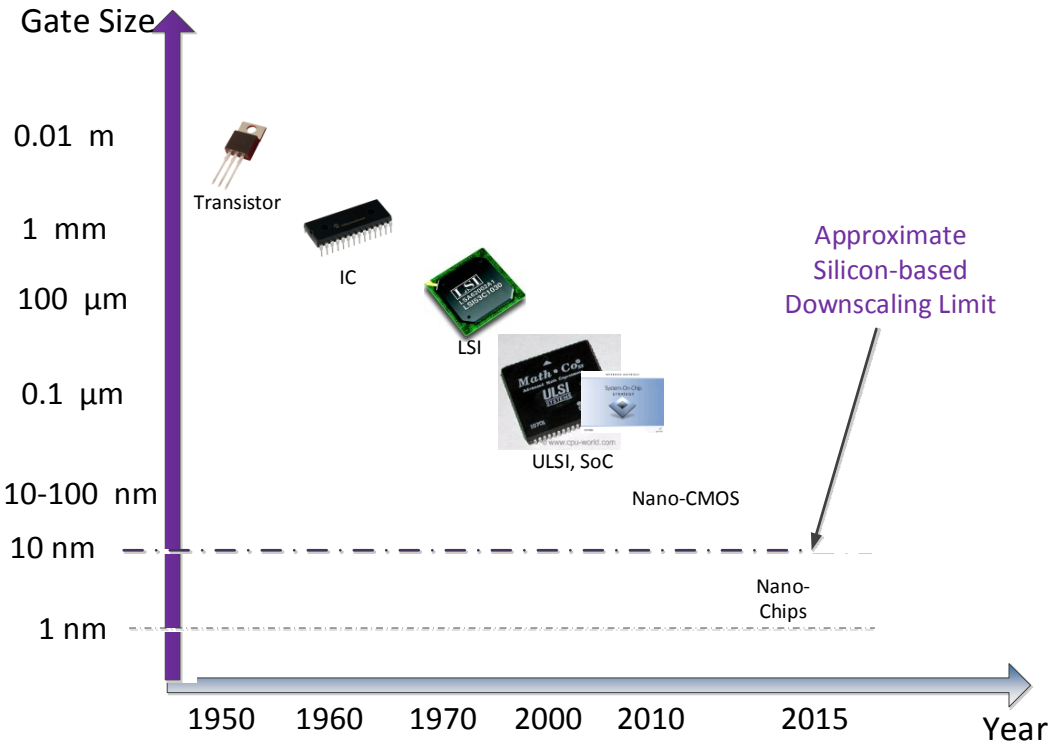


FIG. 4: Trend of Downscaling in Electronic Components Technology [24, 23].

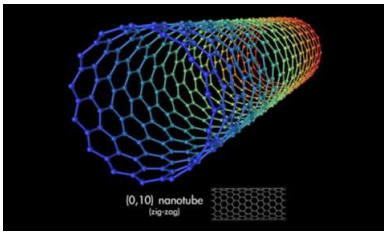


FIG. 5: Carbon Nanotubes (CNT) [1].

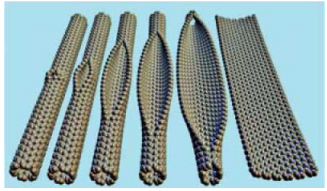


FIG. 6: Graphene Nanoribbons (GNR) [1].

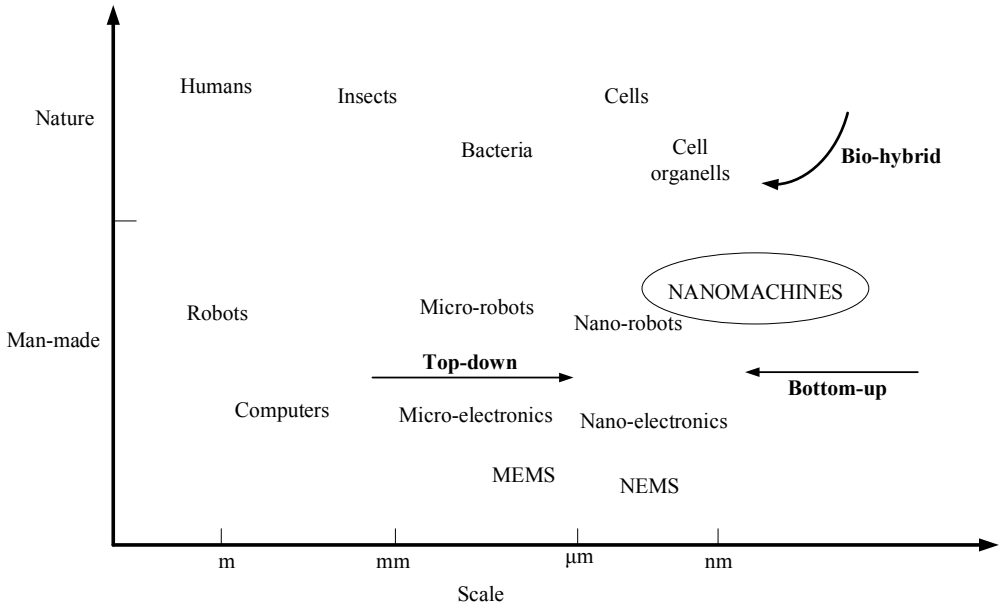


FIG. 7: Fabrication of Nanonodes [1].

scale, consisting of nano-memory, nano-processor, nano-batteries, etc. will be produced. These nodes will be equipped with a wireless communication module that can provide communication among nanonodes and other networks (e.g., traditional sensor network, home networks, wireless local networks).

Nanonetworks [2] are the new generation of networks at nano scale. Each nanonode is composed of nanosensors, nanoantenna, nano-memory, nano-processor, etc. Each nanonode, as illustrated in Figure 8, will have nanometer to micrometer size. Nanosensors are more than the just scaling of sensors. In fact, they take advantage of the unique properties of nanomaterials and nanoparticles to detect and measure new types of events at the nanoscale. For example, nanosensors can detect chemical compounds in concentrations as low as one part per billion [2] or the presence of different infectious agents, such as viruses or harmful bacteria. They will collect useful information that must be sent outside of their sensing environment for storage and additional processing. In other words, they need a communication mechanism between themselves as well as communication with nodes in the micro and macro domains. For this purpose, two main methods of communication have been proposed

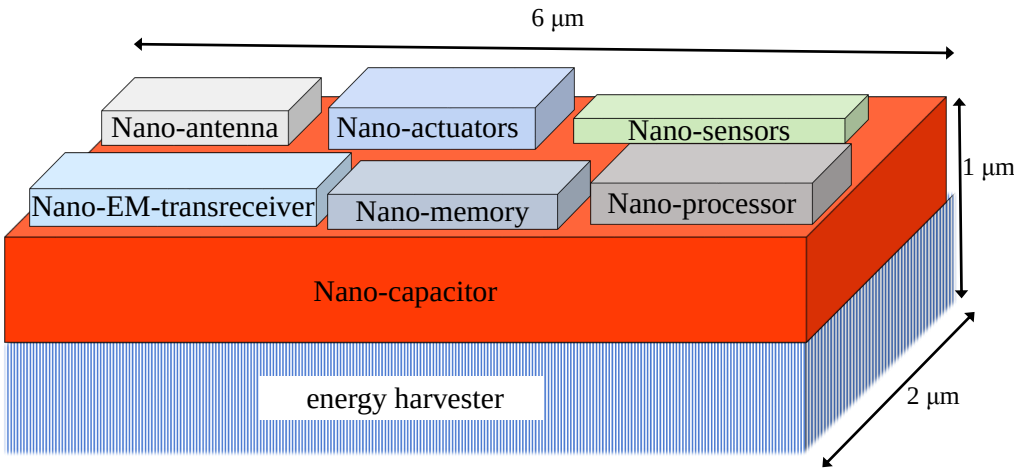


FIG. 8: Structure of a Nanonode [2].

by the research community [1]: molecular communication and electromagnetic communication. We focus on electromagnetic communication, the background of which will be discussed in Section 2.3.

2.2 APPLICATIONS OF NANONETWORKS

The most common application areas of nanonetworks will be biology, medicine, chemistry, environmental science, and the development of military, industrial and consumer goods [2]. In the area of biomedicine, applications such as health monitoring systems for monitoring the amount of sodium, glucose and other ions in the blood or drug delivery systems to deliver drugs to a special part of the body with controlled doses are envisioned. Plant monitoring systems and plague defeating systems are preliminary environmental applications. Nanosensors could also be used in developing new touch surfaces or haptic interfaces as the sample application of nanosensors in industry. In addition, nanosensors can help the realization of Internet of Things (IoT) or even Internet of NanoThings [6]. Moreover, nanosenors could be used to design equipment that is required for augmented reality or game applications.

Even though many potential applications could be imagined for nanosensor networks, the future may reveal some new applications that now are not even imagined. In the following, we describe some potential applications of nanonetworks.

2.2.1 INDUSTRIAL APPLICATIONS

The development of nanodevices and nanomaterials for agriculture and plant research would allow various novel applications, ranging from treatments with agrochemicals to delivery of nucleic acids for genetic transformation [28]. Nanosensors and nano-based delivery systems help in efficient use of water, chemicals and nutrients through precision farming and will help the agricultural industry combat viruses and other crop pathogens. Nanotechnology has the potential to revolutionize the agricultural and food industry with new tools for the molecular treatment of diseases, rapid disease detection, enhancing the ability of plants to absorb nutrients, etc. [29, 30]. Control and monitoring of these molecular level treatments can be enabled through a communication mechanism.

Nanonetworks can help with the development of new materials, manufacturing processes, and quality control procedures. More specifically, these applications have already been proposed. Food and fluid quality control can take advantage of nanonetworks. Nano-sensor networks can help in detecting small bacteria and toxic components that can affect product quality and cannot be detected using traditional sensing technologies [31].

Nanonetworks can be included in advanced fabrics and materials to get new and improved functionalities. Antimicrobial and stain-repellent textiles are being developed using nano-functionalized materials [32]. For instance, nano-actuators can help to improve airflow in smart fabrics. These nano-actuators can communicate to nano-sensors to control the proper reaction based on the external conditions.

As another nanosensor equipped fabric, the Sensoria anklet, as shown in Figure 9, is composed of an e-textile sensor instrumented sock and a snap-on Bluetooth-enabled anklet bracelet [33]. Since very few non-professional runners learn specific running techniques, poor running behavior leads again and again to injuries. Heapsylon's Sensoria Socks technology has been developed specifically to help with this problem and prevent injuries even before they occur by providing feedback based on real world information measured during the user's running session. Nanotechnology communication will enable removing of the bracelet and putting the nanonodes in the sock, which is more convenient and comfortable. A similar miniaturization can occur for electroencephalography (EEG) headsets (Figure 10). Nanonodes can enable the production of invisible EEG headsets.

Smart environments, as a super definition of IoT and cyber physical systems

FIG. 9: The Sensoria Anklet [33].

FIG. 10: Current Size of EEG Headsets.

(CPS), is beyond regular localization or data collection. Smart objects will interact with human in various ways. Apart from the method of interaction and communication with human (e.g., voice, light, etc.), they will provide real-time accurate information for users. For example, one can imagine a scenario where a plant will report its health status, e.g., if it needs more sun or water.

Recently, the deployment of nanonodes to create Network-on-Chip (NoC) has been proposed [34]. Rather than traditional wire solutions, wireless networking of a multi-core system is favored. The unique properties of graphene enables producing 5 μm long and 1 μm wide antennas to radiate in the Terahertz band [35]. This antenna enables integration of one antenna per core as well as providing data rates up to tens of Terabits per second (Tbps) [35]. In this way, nanonetworks can be used to create WNoC [36].

2.2.2 MEDICAL APPLICATIONS

The most important and immediate applications of nanonodes are in the biomedical area. Nanonodes can interact with organs and tissues. This is clearly provided due to nanosize, biocompatibility and biostability. In the following, we describe some potential biomedical applications of nanonodes.

An immune system can be composed of several nanomachines that protect an organism against disease. These nanomachines, including sensors and actuators, can act in a coordinated way to identify and control foreign and pathogen elements. nanomachines can be used to help the detection and elimination of those elements. They could realize tasks of localization and response to malicious agents and cells, such as cancer cells [37], resulting in a less aggressive and invasive treatments compared to the existing ones. Coordination between elements to protect against organisms as well the control and monitoring them remotely could be provided by the creation of nanonetworks.

The monitoring of oxygen and cholesterol levels, hormonal disorders, and early diagnosis are some examples of possible applications that can take advantage of in-body nano-sensor networks. The information retrieved by these systems must be accessible outside the body to doctors, nurses, etc. Thus, nanonetworks must provide the proper level of connectivity to deliver the sensed information.

The use of bio-nanosensors to monitor levels of glucose via implant [38, 31] is one example of a medical application. Nanosensors are implanted in mouse ears (Figure 11) and the level of glucose is monitored through the signal of flourescence. Nanocommunication could enable devices to communicate through electromagnetic signals rather than fluorescence signals, which make it easier for data collection and monitoring. More frequent glucose sensing helps people with diabetes tightly monitor their blood glucose, thus effectively preventing chronic diabetic complications. Once the nanonodes are injected in the body, they can start to sense the environment and possibly communicate with each other using the embedded communication technology (i.e., electromagnetic wireless communication or molecular communication) [39]. If the concentration of any ions goes beyond the identified and preprogrammed threshold, the equilibrium is violated and the nanonode will react as programmed. The nanonode that detects the problem will propagate the information to other nanonodes to trigger a global response. Moreover, a drug delivery system that is

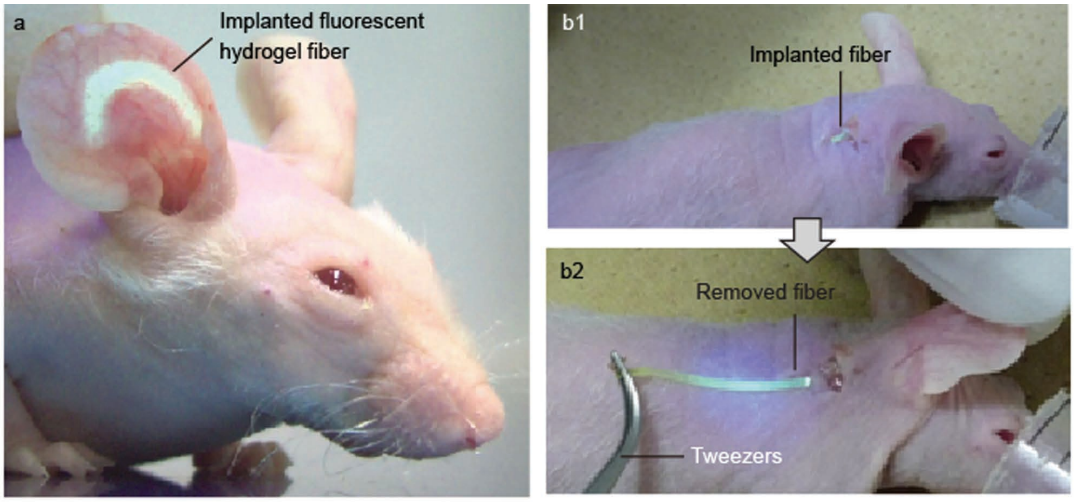


FIG. 11: Biosensors Implanted to Monitor Glucose Level. The fluorescence fiber is injected under the skin (b1) implanted fiber and can be removed (b2) [38].

composed of nanonodes could help to compensate metabolic diseases such as diabetes. In this scenario, nano-sensors and smart glucose reservoirs or producers can work in a cooperative manner to support regulating mechanisms. Nanonetworks will provide the infrastructure for this cooperation.

One other promising application of the nanosensors is checking for bacteria or viruses in hospitals [7]. If contaminating bacteria can be located, it is possible to reduce the number of patients who develop complications such as contagious infections.

Finally, manipulation and modification of nano-structures such as molecular sequences and genes can be achieved by nanomachines. The use of nanonetworks will allow expanding the potential applications in genetic engineering. Nanonetworks enable the control of the nanomachines for performing the genetic operation.

2.3 COMMUNICATION

Two possible communication mechanisms are envisioned [1] for communications among nanonodes: molecular communication and electromagnetic communication.

Figure 12 illustrates the hierarchical architecture that enables microgateways to communicate with molecular and EM nanonetworks. In the case of EM nanonetworks, each microgateway will require dual transceivers: one to communicate with

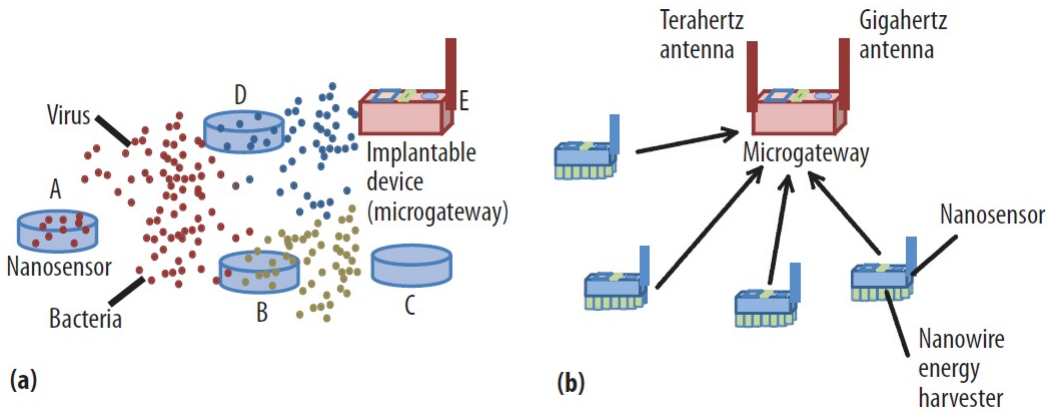


FIG. 12: Nanoscale to Microscale Connection- (a) Molecular Nanonetworks and (b) Electromagnetic (EM) [7].

nanonetworks in the THz band and another to communicate with peer microgateways in another band, e.g., GHz band.

Through this architecture, data can be available in the microscale, which is simpler for transfer and further processing with longer range communication devices and more powerful processors.

2.3.1 MOLECULAR COMMUNICATION

Molecular communication [1] is developed based on chemical interactions of molecules to transfer information/bits. This model complies with the biological environment and has low energy consumption. However, the speed of information transfer is significantly low for most molecular communication models, i.e., bits per hour [1]. A comparison of the speed of several molecular communication methods is shown in Table 1. Except for neuronal signaling, all other methods of molecular communication have a very low speed in transferring molecules.

The various molecular communication mechanisms can be categorized according to the type of molecule propagation as follows.

- **Walkway-based**: In walkway-based molecular communication, the molecules are transmitted via pre-defined pathways by using carrier substances, such as molecular motors [41]. For example, E. coli bacteria can play the role of molecular motors [42].

TABLE 1: Comparison of Molecular Communication Mechanisms [40].

Type	**Distance**	**Propagation Speed**
Vesicular Trafficking	Intra-cell (2 μm)	1 $\mu m/s$
Quorum Sensing	Inter-cell (40 μm)	1 $\mu m/s$
Chemotactic Signalling	Intra-cell (2 μm)	10 $\mu m/s$
Bacterial Migration	Inter-cell (50 μm)	14 $\mu m/s$
Calcium Signalling	Inter-cell (200 μm)	20 $\mu m/s$
IP_3 Signalling	Intra-cell (20 μm)	280 $\mu m/s$
Morphogen Signaling	Inter-org (0.1 cm)	5×10^{-1} $\mu m/s$
Hormonal Signaling	Inter-org (1 m)	5 cm/s
Neuronal Signaling	Inter-org (2 m)	100 m/s

- **Flow-based**: In flow-based molecular communication, the molecules propagate through diffusion in a medium whose flow and turbulence are guided and predictable. Hormonal communication through the bloodstream inside the human body is an example of this type of propagation. Flow-based propagation can also be realized by using carrier entities whose motion can be constrained on the average along specific paths, despite showing a random component. A good example of this case is given by the pheromonal long range molecular communications [43].

- **Diffusion-based**: In diffusion-based molecular communication, the molecules propagate through instant diffusion in a medium. Pheromonal communication [1], calcium signaling [1], or quorum sensing among bacteria [44] are known methods of diffusion-based mechanism.

Since we are interested more in applications that need high data rates for communication, we focus on electromagnetic communication, specifically pulse-based communication [1].

2.3.2 ELECTROMAGNETIC COMMUNICATION

Electromagnetic communication has been proposed [1, 2] as another communication method for nanonetworks. More specifically, pulse based communication in the 0.1-10 THz band has been studied.

There are several drawbacks in existing silicon-based manufacturing techniques that make the downscaling of existing electromagnetic (EM) transceivers infeasible

[45]. Alternatively, nanomaterials are envisioned to solve parts of building a new generation of electronic components that overcome the shortcomings of current technology [46]. Carbon Nanotubes (CNTs) and Graphene Nanoribbons (GNRs) among other graphene-based materials are expected to be the silicon of the 21st century [47]. EM properties of these nanomaterials should be evaluated in terms of bandwidth for emission of EM, the time lag of the emission, and the magnitude of the emitted power for a given input energy, amongst others. Ongoing research on the EM emission on graphene are indicating the Terahertz band (0.1 - 10.0 THz) as the expected frequency range of operation of future nano EM transceivers [11, 48]. In particular, it is determined that a 1 μm long graphene-based nanoantenna can only efficiently radiate in the Terahertz range. This matches the initial predictions for the frequency of operation of graphene-based RF transistors [49].

The emergence of femtosecond lasers and photoconductive antennas during the 1980s made it possible to use THz waves for various applications [50, 51], such as biological and medical science, pharmacology, and security [3, 52, 53, 54]. Since electromagnetic communication can have thermal effects on human tissues, the amount of power that can be safe with no significant temperature rise, i.e., less than 1°C, is considered. Wang et al. [3] show that communication is possible with low power transmission, e.g., order of pJ, and it would not harm human tissues.

The potential and feasibility of THz use in the range of 100 GHz - 10 THz for future wireless communications as an enabling technology has been discussed recently [55, 51, 56, 57]. The IEEE recently created a working group to evaluate the potential applications of the THz band. The Tera Hertz Interest Group (THz IG) in the IEEE 802.15 WPAN wireless standards committee proposes communication in short range applications such as nanocell, short-range fast downloads, etc. [58]. The THz band could not be used for far field communication since it needs high power for transmission [57]. However, it can be easily used for low distance communication, even with non-line-of-sight (NLOS) with low power. Since low power is used, it would not be harmful to the body and could be used for intra-body communication in short distances even with the presence of water [59, 60, 61, 62, 48]. Theoretical measurements suggest the use of 0.1-1 THz for communication on the order of several centimeters for intrabody communication [61]. The challenge is to produce a nanoscale antenna that can be attached to a nanosensor to result in a nanonode. However, researchers have shown that due to the emergence of nanotubes and nanoribbons, nanoantenna

will be available in coming years [2, 48].

Pulse-based communication is a known method [63] in Ultra-Wide-Band (UWB) networks as Impulse Radio Ultra-Wide-Band (IR-UWB) systems. The use of pulses rather than continuous waves requires novel modulation techniques. Conventionally, these are the pulse-based communication modulations [64]: (i) the amplitude of the transmitted pulses (Pulse Amplitude Modulation - PAM), (ii) the temporal position of the pulse (Pulse Position Modulation - PPM), (iii) the pulse width (Pulse Width Modulation - PWM), and (iv) the rate of pulses (Pulse Rate Modulation - PRM). However, these methods are not directly deployable for communication among nanosensor devices, mainly due to the limited capabilities of nanonodes [2]. Apart from that, it does not seem very feasible to detect information from pulse shape (e.g., in PAM) at Terahertz channels [2]. Furthermore, placing the information in the temporal position of pulses (e.g., in PPM) requires accurate synchronization between nanosensor devices [65], which does not seem feasible due to limited resources of nanonodes and the random topology nature of nanonetworks. Finally, PWM and PRM seem to be difficult to implement in nanonodes with limited capabilities. Therefore, the simplest method based on the basic On-off Keying (OOK) looks to be the best choice, at least for the current modeling purpose of nanonetworks [2]. In this modulation, the presence or absence of these pulses is detected by sensing and detecting of energy. Transmitting a pulse represents a logical `1`, and being silent transmits a logical `0` [2].

2.3.3 PULSE-BASED COMMUNICATION MODEL FOR NANONETWORKS

The pulse-based communication model for nodes in nanonetworks, based on the model proposed by Jornet and Akyildiz [2, 5], operates at the THz band communication, which results in a micrometer to millimeter communication range [2]. The nodes use the pulse-based communication and Rate Division Time Spread On-Off Keying (RD TS-OOK) [12] as the modulation mechanism. A logical `1` is transmitted as a femto-second long pulse, and a logical `0` is transmitted as silence. The duration of each pulse is T_p and the time between two symbols is T_s, producing a symbol rate of $\beta = \frac{T_s}{T_p}$.

The probability of collision between symbols is extremely low due to the fact that there can be no collisions for `0` symbols (silences) and that the length of T_s is much

longer than T_p (typically 1000 times larger). However, in spite of other frequency ranges of electromagnetic signals, there is molecular absorption noise, for example 10^{-4} BER for 10 % water vapor. To mitigate the effect of these problems, repetition and code weight techniques have been proposed in [10] and [5].

Repetition is a simple mechanism for error detection and correction. With this method, the sender simply repeats the symbol several times, typically 1 to 9 times. For example, in 3-repetition, a `1` would be transmitted as `111`. In this case, if one or two of these `1`s were not received, the problem could be detected at receiver, and the information (i.e., a bit of `1`) would still be received. Although it is not the most efficient method, it is the simplest method. It is interesting to investigate other methods of coding and error detection and correction. Works such as [66] and [67] are sample of ongoing research in this domain. However, this is not our focus. So, we just use the simple repetition and code weight.

Using symbol repetition necessarily increases the energy required to send data. As `0` symbols take no energy to transmit (because they are silences), it would be most energy-efficient to send as many `0`s as possible. Applying the code weight technique to a packet can result in reducing the number of energy-consuming `1`s that are transmitted. The code weight is defined as the proportion of `1`s to the total number of `1`s and `0`s in the packets [10].

Since transmission of `0`s in RD TS-OOK pulse-based modulation is equal to silences that do not consume energy, the lower code weight can lower the energy consumption. Moreover, the code weight can lower the collisions since fewer `1`s, which are the only pulses that can face collision, are transmitted. A code weight of 0.5 means that, on average, there are an equal number of `1`s and `0`s in the packets. A lower weight, such as 0.4, means that there are fewer `1`s. However, it also means that more bits should be used to send the same of amount of information. For example, Table 2 shows how the number of `1`s for sending two bits of information could be reduced by using three bits. The code weight in this example is decreased from 0.5 to 0.25.

For a more realistic example, for sending $n = 64$ bits of information with a code weight of 0.4, at least $a = 6$ more bits will be added to each packet. In this case, the total number of encoded bits would be $m = 70$ and the number of `1`s, u, is less than or equal to 28.

To make sure that for a target code weight, there are at most u `1`s independent

TABLE 2: Code Weight Example.

information value	**coding with 2 bits (weight = 0.5)**	**coding with 3 bits (weight = 0.25)**
0	00	000
1	01	001
2	10	010
3	11	100

of the original bit values, for n bits of information, the $\frac{m!}{(m-u)!u!} \geq 2^n$ condition must be satisfied with the minimum a additional bits, where the total number of bits is $m = n + a$.

The method to determine the additional number of required bits is as follows. First, for a specific code weight W, u is specified as

$$u = \lceil W \cdot m \rceil \tag{1}$$

and the following condition must be satisfied with the minimum a, where $m = n + a$.

$$\frac{m!}{(m-u)!u!} \geq 2^n \tag{2}$$

Note that sending fewer `1`s consumes less energy in the sender while it consumes more energy in the receiver. Energy is consumed when receiving any bit, `0`s or `1`s. Decreasing the code weight necessarily increases the packet size, increasing the cost to the receiver. Depending on the packet length and the ratio of energy required for reception to that for transmission of a pulse, named as α, the code weight may or may not save energy in total. Here, the assumption is that α is small, e.g., 0.1. Therefore, the aim is to find the optimum values for *packet length* and *code weight*, which we address in Chapter 4.

2.4 BIOCOMPABILITY

One of the main dominant applications for nanonetworks would be in the medical and health domains. Therefore, the biocompability and safety of these devices, especially for the human body, is important. There are two type of devices: (I) implantable devices which reside in the human body and are expected to stay for tens of years; and (II) disposable devices which will exit the body after they are depleted.

In the following, we take a look at some of recent advances in biocompability and safety issues for these type of devices.

In disposable devices, to provide solubility for carbon nanotubes, several categories of biomodification (i.e., covalent attachment, noncovalent attachment, and hybrid) have been proposed [68]. This is one of the methods which is used for the disposal of nanonodes.

Toxicity and possible damage of graphene based material is one of the concerns for nanomedicine when they reside inside the human body. However, recent modifications of the structure of these materials enable their safe implantation. For example, a type of biocompatible coating (e.g., PEGylation) is mainly localized in the reticuloendothelial system, including the liver and spleen after intravenous injection, and could be gradually excreted from mice without causing noticeable toxicity to the treated animals at a dose of 20 mg/kg over a course of 3 months [69].

Some other approaches such as e-skin, or safe implantable nanodevices are ongoing research efforts. E-skin [70] has been developed recently, which can be worn on the skin, is very flexible, and can be used to monitor body temperature and other vital signs. They can be bend and form as required.

Webster [71] introduces the advances in the development of a safe implantable nanosensor that can measure cellular function. While the applications for this technology are numerous, orthopedics is the first target. The device can be bundled with, say, a spinal implant or a hip implant to help repair damaged bone. As soon as the implant is inserted in the body, the sensor can determine if a bone cell that attached to the implant. It can detect presence of bacteria or an inflammatory cell. The technology detects the type of cells attached to implants by measuring their conductivity. Each one of those cells, a bone cell, a bacteria, or an inflammatory cell, has different conductivity level, which will be measured by these nanosensors.

Using radio frequency communication, the sensor can communicate how well an implant is faring in the body to a handheld device. The program within the handheld device would interpret that signal and provide feedback to the patient. A patient, for instance, might be informed that the bone growth surrounding the implant is healthy. Or, if bacteria is growing on the implant or inflammation is setting in, the patient could be instructed to make an appointment with their orthopedic surgeon. Alternately, upon detecting bacterial growth or inflammation, the device could trigger the release of either an antibiotic or an anti-inflammatory agent.

2.5 ENERGY HARVESTING

Energy harvesting has attracted researchers for several years ever since devices were first built to harvest solar energy. However, not only is solar energy limited to specific times and locations, but also the capacity for storage of the energy is limited. Therefore, researchers have investigated new methods of energy harvesting such as ambient vibration or heat. Independent of the type of resource for energy harvesting, they mainly share a common property: the arrival of energy follows a stochastic process.

Energy harvesting has been investigated by researchers from various points of view. In the following, we first introduce the energy harvesting taxonomy. Next, the methods for modeling the energy harvesting process are introduced. Finally, we describe the energy harvesting model for nanonetworks, developed by Jornet and Akyildiz [5].

2.5.1 SOURCES

Energy sources are categorized broadly into (I) ambient energy sources such as solar, wind, RF, and ambient vibration; and (II) human power [72]. Human power could be passive such as blood pressure, body heat, heartbeat and breath, which are not user controllable, or it could be an active type that is controllable, such as finger motion, paddling, and walking.

There are three main metrics for the evaluation of harvesting methods ([73],[74], [72]):

- conversion efficiency: This is the amount of energy that is harvested compared to the amount of available energy.

- energy harvest rate: This parameter specifies how fast the energy can be harvested. This metric is dependent on various factors. For example, in solar systems, the size of solar panel and weather conditions (e.g., sunny, cloudy) can affect this parameter. In vibration energy harvesting, the rate of vibration affects the rate of energy harvesting.

- power density: This indicates the amount of power (time rate of energy transfer) per unit volume, measured in $\frac{Watt}{m^3}$. It mainly defines the specification of the system's energy storage.

Harvested energy is used in two ways:

- harvest-use: In this method when energy is produced, it is used immediately. An example of this method is pushing a key/button. Pushing produces some energy that can be used to transfer an electronic signal.

- harvest-store-use: In this method, energy is harvested whenever possible and is stored for future use. Obviously, this architecture is more useful since there is some energy available most of the time if it is consumed wisely. The limitation comes only from the capacity of storage. Most studies in the domain of networking use the harvest-store-use method. In these situations, two approaches are taken: (I) finding the required capacity of storage to meet the application requirement; and/or (II) trying to optimize usage of this energy. In both cases, the modeling of the energy harvesting process plays the key role. In this next section, we introduce some of modeling methods.

2.5.2 VIBRATION

Among all sources of energy, in this thesis, we focus on energy harvesting from vibration since it is very useful for medical as well as indoor industrial applications. Table 3 represents some of the potential sources for harvesting energy from vibration, including their frequencies and acceleration amplitudes. New generations of piezoelectric-nanowire are sensitive to very low acceleration [16]. Therefore, the main parameter that affects the energy harvesting amount is the frequency.

The amount of power that can be harvested through vibration is compared with other sources of energy in Table 4 in terms of power density. Power density is the amount of power (time rate of energy transfer) per unit volume or surface [78]. Volume is expressed as W/m^3, and surface power density is expressed as W/m^2.

As it can be seen, piezoelectric nanowire provides a significant amount of power density. The limitations in fully utilizing this power density comes from size limitations for nanonodes (scale of nanometers to micrometers) as well as vibration source availability. For example, from arm motion, at maximum 330 $\mu W/cm^3$ can be extracted.

TABLE 3: Peak Frequency and Acceleration Amplitude for Various Vibration Sources [75, 76, 77].

Vibration Source	**Peak Frequency (Hz)**	**Acceleration Amplitude ($\frac{m}{s^2}$)**
Refrigerator	240	0.1
Car engine compartment	200	12
Door frame just after door closes	125	3
Kitchen blender casing	121	6.4
Clothes dryer machine	121	3.5
Small microwave oven	121	2.25
Washing machine	109	0.5
External windows next to a busy street	100	0.7
Second story of wood frame office building	100	0.2
HVAC in office buildings	60	0.2-1.5
Vehicles	5-2000	0.5-110
Person nervously tapping their heel	1	3

2.5.3 EVALUATIONS AND MODELS

There are many works in the literature about modeling energy harvesting (e.g., [80, 84]). Sharma et al. [84] model energy harvesting and energy consumption as a queuing system. Then based on stationary analysis, they propose a transmission strategy to optimize the throughput of a sensor node. Their model considers only one node and the energy required for transmission. The model also assumes that the data buffer and energy storage are infinite, which might not be the case in many situations such as nanoscale nodes. Gorlatova et al. [80] find the best spending rate of energy consumption for a node/link through optimization and lexicographic frameworks. They develop their algorithm for predictable energy inputs as well as stochastic models. The model has been evaluated in a network of RFID active tags.

Optimized algorithms for energy harvesting can be categorized according to various aspects as follows:

- *energy model profile*: Several parameters such as energy source (e.g., solar,

TABLE 4: Comparison of Power Density for Various Harvesting Sources and Technologies [79, 80, 81, 75, 82, 16, 83].

Source/Technology	**Power density ($\mu W/cm^3$)**
Solar (outdoors)	15,000 direct sun, 150 cloudy day
Piezoelectric-nanowire	2800
Arm motion	330
Shoe inserts (pizoelectric vibration)	330
Running	300- Max from kinetic
Vibrations (piezoelectric conversion)	250
Vibration (small microwave oven)	116
Vibrations (electrostatic conversion)	50
Batteries (non-rechargeable lithium)	45
Walking	30- Max from kinetic
Light	25- outdoor at night, 100- indoor
Temperature gradient	10-60, depends on temperatures and difference known as Carnot efficiency
Batteries (rechargeable lithium)	7
Solar (indoors)	6 office desk
RF	0.02 -40, depends on source and distance
Acoustic noise	0.003 at 75 dB, 0.96 at 100 dB

vibration, RF) and environment (e.g., indoor/outdoor, vibration rate, temperature) can produce different energy model profiles. *Predictable*, *partially predictable*, *stochastic*, and *model free* are known categories that have been identified and studied [73, 72].

- *ratio of energy storage capacity to energy harvested*: This parameter specifies how fast the energy storage is filled. It depends both on the capacity of energy storage and the availability of energy. In other words, it connects the *energy harvest rate* to *power density*.

- *time granularity*: This specifies the timescale of decision making and designing schemes, algorithms, and protocols. The timescale can be in the range of seconds to days. It is related to the *storage-harvesting ratio* as well as the *energy profile model*. The higher the time granularity, the more accurate a design is required. This is important in applications where there are QoS requirements for data transfer.

- *problem size*: When we are solving any problem for efficient energy harvesting, the design can be evaluated in the domain of a node, pairwise nodes (link), or network wide (e.g., routing).

In the following, we describe some of works in modeling the energy harvesting process. Table 5 compares the works based on various design aspects. These models can be categorized into two general types: lexicographic and stochastic.

Lexicographic[1]: Gorlatova et al. [80] investigate solar power for active tag RFID nodes. The authors propose various time fair energy allocations for both predictable energy inputs as well as stochastic inputs. Based on real environment measurements, they develop a prediction model for energy arrivals. Next, they use the lexicographic maximization and utility maximization framework for modeling their energy spending rate. They achieve fair allocation of resources among nodes over a one day duration. Then, they consider a stochastic energy arrival and claim that, based on a developed

[1]Lexicographic optimization is a form of multi-criteria (multi-objective) optimization in which the various objectives under consideration cannot be quantitatively traded off between each other, at least not in a meaningful and numerically tractable way. Lexicographic method assumes that the objectives can be ranked in the order of importance. It can be assumed, without loss of generality, that the among k objective functions are in the order of importance so that f_1 is the most important and f_k the least important to the decision maker. Then, the lexicographic method consists of solving a sequence of single objective optimization problems [85].

Markov Decision Process, they achieve an optimal energy spending policy for a single node or link. Liu et al. [86] design a fair and high throughput data extraction as well as a routing path solution among all nodes, where the energy model is developed for solar power. They develop a centralized solution and two distributed solutions. The main idea is to adapt the extraction rate (sensing and sending rate of information) based on the available energy. A rate assignment for data transfer is found through lexicographical optimization. Even though the strength of the scheme is that it is independent of the energy arrival profile, the optimization solution works only on a large time scale, such as a day.

Stochastic: Sharma et al. [84] model energy arrival and consumption as a $G/G/1$ queue. After finding the stationary state of the model for some specific conditions, they try to find the optimum throughput (largest possible data rate of packets) based on their energy management policy. Later, they try to minimize the delay of packets in the buffer. The optimization model is called α-discount optimal and is developed based on the stationary state of Markov model. The main weakness of their model is that they assume that energy and data buffer are infinite. The goal of the scheme developed by Khouzani et al. [87] is to achieve the highest data rate that results in a long term optimal solution. The advantage of the scheme is that it requires no explicit knowledge of the energy harvesting profile or traffic generation process. In fact, it is a learning system that adapts itself based on the environment (i.e., available energy) and network circumstances. Their scheme works at the node level as well as the network layer. Their work is limited to analysis, and no simulation or test-bed results are provided. The main goal of the model developed by Luo et al. [88] is to develop an optimized training model. Then, the model will be used for a transmission policy that specifies the energy spending based on channel state information (CSI). The paper assumes an infinite buffer level. Finally, Wu et al. [89] use a new method of evaluating the stochastic properties of energy harvesting while they evaluate the network performance. They try to support a soft QoS. However, they develop only a framework and it is not clear how efficient the model would be.

There are some other works involving stochastic modeling of energy consumption that focus on other aspects of energy harvesting. For example, Gatzianas et al. [90] use a stochastic optimization framework for modeling the problem. The focus of the work is on the variation of the channel, so they try to develop a model for the stochastic behavior of the channel while achieving the best policy on transmission

TABLE 5: Comparison of Energy Harvesting and Consumption Modeling.

Ref.	**Energy Model Profile**	**Time Granularity**	**Problem Size**	**Solution Method**	**Network**	**Energy Source**
[80]	predictable and stochastic	day	node, link	lexicographical	RFID	light
[86]	almost independent	day	node, network	lexicographical	sensor networks	solar
[84]	independent (General arrival)	seconds-minutes	node, partially network	queueing	sensor network	any
[87]	almost independent (General arrival)	seconds-minutes	node, network	time discrete	sensor network	any
[89]	independent	seconds	network	stochastic network calculus	N/A	N/A

and energy consumption. Kar et al. [91] model the duty cycle of sensor nodes. In this model, it is assumed that nodes cannot harvest energy and communicate simultaneously. Therefore, they need to switch between active and passive states. The goal is to optimize the timing of sleep/awake to maximize a utility function such as throughput.

Energy-Neutral Operation (ENO) [92, 93] is defined as how to operate such that the energy used is always less than the energy harvested. This concept is used to find an estimate for the battery size based on an average approach for the rates of energy harvesting and consumption, where energy storage is not 100% efficient and there is energy leakage. Also a power management system is developed to optimize the harvested energy. They use an exponentially weighted moving-average (EWMA) filter to predict the arrival of energy and then compute the consumption rate based on the prediction. In the next time slots, the prediction is adjusted based on real values. Niyato et al. [94] consider the problem of duty cycling for sleep and awake times when charging occurs during sleep periods. Finding the optimum sleep and awake time is solved through a game-theoretic approach. Noh et al. [95] develop an optimal distribution of energy consumption on defined intervals. However, their

model does not incorporate the stochastic nature of energy harvesting. They use historical data to model energy harvesting. Finally, they optimize the flow control based on their energy harvesting model.

2.5.4 MODEL FOR NANONETWORKS

Most of current energy harvesting and consumption models lack a complete view of this joint process. Even some of work such as [84] and [5] have limitations. Sharma et al. [84] assume an unlimited energy buffer that is not a true assumption in nanonodes with size limitations. Also, the range of optimization (i.e., days) with solar power does not comply with nanosensor network scenarios with limited access to solar power. As will be discussed in Section 2.5.5, vibration is envisioned as the main source of energy harvesting for nanonodes. Moreover, in most of the models, it is assumed that energy storage is linear while it has been shown that this would not be the model for ultra-nanocapacitors in nanondes [5]. The only work that includes these properties is developed by Jornet and Akyildiz [5]. However, they do not provide a flexible model that can be used for various traffic models. In addition, their energy harvesting process is developed only for the Poisson process. Also, their model does not reveal the effect of each consumption parameter for optimization. So, one of our first challenges is to develop a comprehensive model of energy harvesting and consumption that includes a general stochastic model energy harvesting process with a comprehensive traffic model. This model will be discussed in more detail in Chapter 3.

2.5.5 VIBRATION MODEL FOR NANONETWORKS

We use the energy harvesting model developed by Jornet and Akyildiz [5]. As shown in Figure 13, the stored energy in an ultra-nanocapacitor by piezoelectric nanogenerators has a nonlinear model.

In this model, energy is harvested through vibrations, which produce compress-release cycles of the nanowires on a nanocapacitor. For a specific ultrananocapacitor, the stored energy is specified by the number of cycles. The energy-harvesting rate (Joules/second) is defined as

$$\lambda(E_{cur}, \Delta E) = \frac{1}{t_{cycle}} . \frac{\Delta E}{n_{cycle}(E_{cur} + \Delta E) - n_{cycle}(E_{cur})}, \tag{3}$$

where t_{cycle} is the time between cycles, $n_{cycle}(E)$ is the number of cycles required to

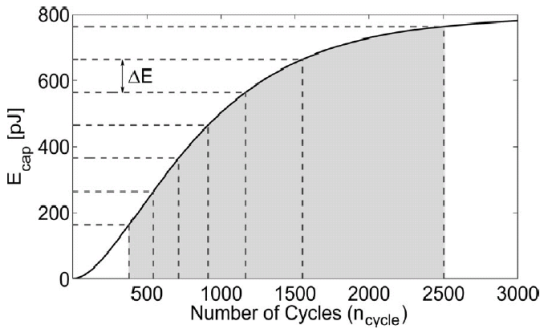


FIG. 13: Energy Harvesting Model [5].

generate E Joules, E_{cur} is the current energy level, and ΔE is the amount of energy increase. We note that if every vibration generates one cycle, then the inverse of t_{cycle} is the vibration rate.

2.6 COMMUNICATION ENERGY CONSUMPTION MODEL

The energy required for transmission and reception of a packet can be computed as follows. For a packet of size N_{packet} bits, the energy consumed when transmitting or receiving a packet with code weight W are respectively given by

$$E_{packet-tx} = N_{packet} \cdot W \cdot E_{pulse-tx} \tag{4}$$

$$E_{packet-rx} = N_{packet} \cdot E_{pulse-rx} \tag{5}$$

where $E_{pulse-tx}$ and $E_{pulse-rx}$ are the energy consumed in the transmission and in the reception of a pulse, respectively. This energy calculation is formulating the energy for transmission and reception of a packet independent of the number of tries for transmission. If re-transmission of packets exists, then these formulas can be extended using the expected value of the packet length.

2.7 NETWORKING - MAC PROTOCOL FOCUS

There has been a large body of research in MAC protocol design for sensor networks and UWB networks. A comprehensive survey was compiled by Akyildiz et al. [96] for sensor networks and by Gupta and Mohapatra [97] for UWB networks. However, these MAC protocols cannot directly be used in nanonetworks because

they do not consider either the limitations of nanodevices or the characteristics of the Terahertz band.

First, the majority of existing MAC protocols for wireless networks have been designed for band-limited channels. Nevertheless, in nanonetworks the Terahertz channel provides nanodevices with an almost 10 THz wide window.

Second, carrier-sensing techniques in classical MAC protocols cannot be used in pulse-based communication systems since there is no carrier for sensing. Only some solutions [97] proposed for Impulse Radio Ultra Wide Band (IR-UWB) networks could be considered, but their complexity limits their usefulness in the nanonetwork scenario. For example, generating and distributing orthogonal time hopping sequences is not a lightweight process for nanodevices.

Third, the main limitation for nanodevices comes from the very limited energy that can be stored in nanobatteries. Therefore, energy harvesting-aware protocols are required.

In the domain of nanosensor networks, Jornet et al. [13] present an initial effort in MAC protocol design, PHysical Layer Aware MAC protocol for Electromagnetic Nanonetworks in the Terahertz Band (PHLAME). This protocol is built on top of the Rate Division Time-Spread On-Off Keying (RD TS-OOK), and it also exploits the benefits of novel low-weight coding. They analytically study the performance of the proposed protocol in terms of energy consumption, delay and achievable throughput, by using models of the Terahertz channel (path-loss and molecular absorption noise) and interference. However, there are open issues (e.g., optimization of parameters, energy efficiency evaluation of the method, limited performance evaluation) in their protocol in addition to the lack of simulation.

2.8 COMPARISON OF DISSERTATION CONTRIBUTION TO RELATED WORK

In the following we describe and compare our solutions with some existing work.

- *Energy Harvesting and Consumption Modeling*: As discussed in Section 2.5, most energy harvesting models developed in other domains such as sensor networks are not applicable for several reasons. First of all, each of the stochastic-based models has limitations such as unlimited energy buffer which cannot be used in our scenario. Second, they mainly assume a linear model for charging their energy storage while our model is based on a non-linear model. Third,

TABLE 6: Comparison of Our Energy Harvesting and Consumption Model with Previous Works - Vib. = Vibration, Y = Yes, N = No.

Ref.	[86]	[80]	[84]	[89]	[5]	Our Model
Stochastic Energy arrival model	N	N	Generic	Generic	Only Poisson	Generic
Energy Source	Solar	Light	General	General	Vib.	Vib.
Nonlinear energy storage	N	Y	N	N	Y	Y
Network Traffic Model	Y	N	N	Y	partially	Y
Pulse based communication	N	Y	Y	N	Y	Y
Optimum packet design	N	N	N	N	partially	Y

we are using ambient vibration as the source of energy, which is a less studied resource. Also, models that are independent of energy resources are not applicable due to their very generic modeling. Next, most models do not include consumption and harvesting at the same time. Even if they do, they are not built on pulse-based communication. Therefore, it is not possible to evaluate the model for different parameters such as packet length, traffic model, etc. Table 6 summarizes the differences between our model and previous work. The Jornet and Akylidiz model [5] is the most similar to our model. However, it has many limitations in the stochastic energy arrival model and network traffic model, among others. Therefore, we propose a more comprehensive model. Moreover, we develop a multi-objective optimization problem which can be used to find the optimum values for packet size, code weight, and repetition to optimize energy consumption in relation to delay and reliability.

- *Maximize Utilization of Harvested Energy Consumption*: The goal is to develop energy-harvesting-aware [92] rather than energy-efficient methods. In energy-efficient methods, the energy budget is limited and the available energy over the total period of problem modeling should be optimized. However, in energy harvesting-aware, the decision about the situation depends on the moment, the amount of available energy at the moment, and the prediction of energy arrival. Therefore, the optimum use of energy needs a different model.

TABLE 7: Comparison of Our Optimization Approach - Y = Yes, N = No.

Ref.	[86]	[95]	[98, 80]	Our MDP model
Stochastic arrival	N	N	limited	Y
Applicable for both Linear and Non-linear Storage	N	N	Y	Y
Applicable in Nanoscale	N	N	Y	Y
Pulse-based	N	N	Y	Y
Objective Function	maximize data collection	reduce consumption rate variance	maximize data rate-no behavior guarantee	maximum utilization with known behavior

Although others [95, 86] argue that the optimization of energy consumption in perpetual networks are different from typical battery-based networks, they do not address the problem in the way that we are considering here. Liu et al. [86] focus on consumption for data collection, not energy consumption for communication. Noh et al. [95] describe the problem of optimization, where energy arrivals are stochastic. However, they develop their solution based on a historical prediction model of energy arrival, not an exact probability distribution function. Gorlatova et al. [98, 80] consider a stochastic model which maximizes the data rate and smoothing consumption for the discrete distribution of energy arrivals. Moreover, the model does not behave based on the stochastic arrival of energy. Therefore, nodes can be without energy for unknown periods of time. Table 7 summarizes the differences of my approach with existing works.

- *Communication Design*: Even though there are many methods for communication among wireless nodes in vicinity of each other, most of them are not designed for pulse-based communication. Table 8 compares our model with current medium access models. Energy is not taken into account in most of the pulse-based models. In addition, we develop a model that is distributed.

TABLE 8: Comparison of Existing Medium Access Solution with My Approach - Y = Yes, N = No, n/a = Not Applicable.

Ref.	**For pulse comm.**	**Req. time syn.**	**Harvest-aware**	**OOK modula-tion**	**Dist.**	**Applicable in nanoscale**
TDMA	N	Y	N	n/a	N	N
FDMA	N	partially	N	n/a	N	N
CSMA/CA	N	partially	N	n/a	partially	N
MACs for UWB	Y	Y	N	N	N	N
Phlame [13]	Y	unknown	N	Y	unknown	Y
Our model (RIH-MAC)	Y	partially	Y	Y	Y	Y

Previous distributed models such as [99] and [100] are evaluated in context of electromagnetic carriers. Here, we are looking to find the appropriate solution in pulse based communication. Also, we include the energy harvesting in design.

2.9 NETWORK ASSUMPTIONS AND CHARACTERISTICS OF APPLICATIONS

In this section, we take a look at how the nanonodes are located next to each other to form a nanonetwork in the presence of an application. Although the formation of a nanonetwork, i.e., topology, could be widely different based on the application, we try to define at least one common expected one. We define our network and topology assumptions based on our defined type of application for nanonetworks.

2.9.1 BLOOD MONITORING APPLICATION

In a blood monitoring application, nanonodes are embedded inside the blood vessels to monitor the blood. They will be able to measure the amount of glucose, mineral ions, hormones, carbon dioxide, etc. This would be the simplest way to help people with diabetes or the possibility of having blood clot. If the amount of any measured element changes significantly based on defined thresholds, the nanonodes

may release some drugs while they also transfer these measurements to outside the human body.

The number of nanonodes that are required to make this measurement effective depends on the fabrication of devices and required measurement accuracy. However, with artery diameters between 0.5-10 mm and nanonodes of 10 μm, there could be 1-10 nanonodes at each point, with the assumption that nanonodes do not occupy more than $0.5 - 1\%$ of artery diameter, to avoid interference with blood flow. If it is assumed these nanonodes have a 1 cm communication range, and the information from these nanonodes is required to be transferred over a 30 cm distance[2] until for example they arrive at a gateway[3] to be sent to outside body domain, then there would be a linear topology of nanonodes which could consist of 30-3,000 nodes. Although in the simplest form, a nanonode would have two neighbors, if several arteries transfer information to a larger artery, then a node at the interconnection of those arteries can have more than one neighbor. Also, in wide arteries, we assume that up to 10 nanonodes can co-locate. Therefore, the number of neighbors for each nanonode can be between 2 to 12.

2.9.2 INFORMATION FLOW

The simplest method to enable the information flow in this network of nanonodes is to use a flooding mechanism. Since flooding is a resource expensive method, we consider a probabilistic flooding method, which will be described in Section 3.1.1. Even though more efficient routing methodologies could be developed, we just use this simple method since the focus of this dissertation is on the data link layer and energy harvesting issues. Customized routing schemes could be developed in each application domain. For example, for intra-body health monitoring applications, where the main functionalities are monitoring and control, a neuron inspired information flow looks promising.

2.9.3 ASSUMPTIONS

We assume the techniques and algorithms in this dissertation are developed for an application of this described category. The nanonodes will communicate with the

[2]Note that this distance can be for a single artery, or from a path over a tree of connected arteries, where smaller arteries are connected to larger ones.

[3]There could be more than one gateway, which can be located at several parts of body such as waist, wrist, chest, knee.

microscale domain through one or several gateways. These applications can tolerate delay in the reception of information on the order of minutes. The amount of information to be transferred can be handled with the limited available resources of nanonodes, i.e., nanomemory, ultra-nanocapacitor. For example, nanonodes with a limited energy budget cannot transfer the measured blood quantities every millisecond due to the limitation in available energy. Models in this dissertation will reveal the information flow capacity that a nanonetwork can support based on different parameters.

We assume that the topology of network is fixed and nanonodes do not have mobility. This work is a first step toward the development of protocols for nanonetworks, and we consider a network with a fixed topology. Future work would focus on extending this to a mobile environment. Nanonodes may run out of energy which means the neighbors of a nanonode would not be available all the times. This will introduce a transient dynamic topology. However, when the nanonode harvest enough energy, it will be available for communication with its neighbors. Therefore, the topology of the nanonetwork in terms of the location of nanonodes is assumed to be fixed.

2.10 SUMMARY

In this chapter, we studied nanonetworks from these aspects: communication, applications, energy harvesting, and energy consumption for communication. We also reviewed works related to energy harvesting in other networks such as sensor networks and RFID networks. We studied how the energy harvesting process is modeled and optimized. The studied methods are not applicable in nanonetworks for several reasons: the assumption of unlimited energy storage, the long duration of optimization (e.g., day), limited analytical results, and non-stochastic energy arrival models. Therefore, we will develop energy harvesting and consumption models that are compatible with nanonetwork characteristics. The goal would be to find models with which application requirements can also be evaluated. Next, methods that maximize consumption utilization of harvested energy are investigated. In addition to the energy harvesting issues, many other open issues such as communication between nodes for creation of nanonetworks are in the early stages of research, as shown in

Section 2.7. Thus, in other parts of our work, we will evaluate energy-harvesting-aware communication models between nanonodes.

CHAPTER 3

PRELIMINARY MODEL OF ENERGY HARVESTING AND CONSUMPTION

3.1 MODEL FOR ENERGY CONSUMPTION AND HARVESTING

In this section, we present the system model which covers network traffic, energy harvesting, and energy consumption. This simple model is developed to understand the role of parameters in combined energy harvesting and consumption processes. This model, of course, could be improved in many directions. However, we are concentrating on a feasible and simple model to identify the role of parameters, i.e., packet size, code weight, and traffic rate.

3.1.1 NETWORK TRAFFIC MODEL

In a nanonetwork, each nanonode will periodically sense and broadcast data. Because of the short communication range, nanonodes will also need to function as routers, re-broadcasting data that they hear from neighbors. The protocol for forwarding neighbors' data is a critical element of the traffic model. Forwarding every packet from all neighbors leads to flooding, but not sending neighbors' traffic severely restricts the communication range of the network. Several methods such as probabilistic-based, counter-based, location-based, distance-based, etc. have been proposed in the literature, e.g., [101] and [102]. For example, in [102], a node decides about forwarding a packet based on various parameters such as a random process, its relative location, or the signal strength of received packet. These methods show that the network flooding problem can be avoided efficiently.

In this dissertation, we develop a model that provides the flexibility of increasing or decreasing the forwarding rate while enabling us to model the traffic rate with energy consumption. The model has been inspired by works such as [101] and [102], which can resolve the flooding (broadcast storm) problem. We assume that each node is sensing, resulting in a rate of μ_{info} bps to be sent. The policy for forwarding

neighbors' traffic is to forward neighbors' traffic with a probability of $\frac{g}{G}$, where G is the number of neighbors and $g \in \Re(0..G]$ is the portion that the scheme decides to forward. Setting $g = G$ will result in flooding. The ratio $\frac{g}{G}$ could be set as a probability ratio p, where $p \in [0..1]$, similar to [101] and [102]; however, without loss of generality, the $\frac{g}{G}$ ratio provides an adaptive probability based on the density of network. Smarter routing schemes could be evaluated. However, since the focus of this model is to understand the process of energy consumption and harvesting, we simply use this routing scheme. We assume that topology would be known and controllable, and therefore the number of neighbors would be a given in design. Otherwise, the number of neighbors can be found by methods such as [103] and [104]. We are assuming that G is the same for all nodes, which is a valid assumption for most network deployments, such as grid/mesh and uniform. There are some nodes on the edge of deployment that may have fewer neighbors, but that does not affect the model, because it is modeling the upper bound, i.e., the maximum possible number of neighbors.

Based on the described policy, the rates that data are sensed, received, and transmitted by each node consist of the rate at which sensed data is sent ($\mu_{send-sense}$), the rate of reception for sensed data from neighbors ($\mu_{recv-neighbors}$), the rate of reception of forwarded data from neighbors ($\mu_{recv-neighbors-forward}$), and the rate of forwarding neighbors' data ($\mu_{send-forward}$). These rates are given in Eqs. 6-9, where μ_{info} is the sensing rate, which is the same for all nodes. Figure 14 illustrates this. Note that $\mu_{recv-neighbors-forward}$ is the sum of all of the traffic forwarded by a node's neighbors and is limited by $MAXHOP$, the maximum number of hops that a node's sensed data will travel. We assume that nodes have the information about the maximum number of nodes that a packet needs to travel to arrive at the destination. In the simplest way, it can be estimated from the time to live (TTL) that is set for the packet. The value of $\frac{TTL}{E[T]}$ would be MAXHOP, where E[T] would be the expected transfer and queueing time between two nodes.

This includes traffic received from the neighbors' neighbors. Traffic received from a 1-hop distance is $G \cdot (\frac{g}{G})^1 \mu_{info}$, from a 2-hop distance is $G^2 \cdot (\frac{g}{G})^2 \mu_{info}$, and so on. It means that for all i-hop distance nodes, received traffic is $G^i \cdot (\frac{g}{G})^i \mu_{info}$. Finally, the rate of information forwarded by a node, $\mu_{send-forward}$, is $\frac{g}{G}$ of all the traffic received from neighbors (Eq. 9).

$$\mu_{send-sense} = \mu_{info}, \tag{6}$$

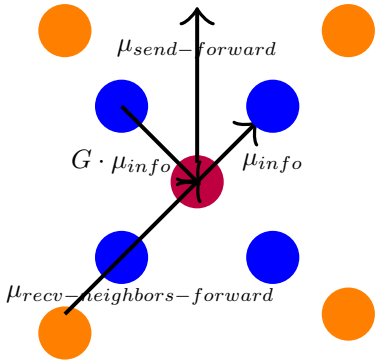


FIG. 14: Traffic model.

$$\mu_{recv-neighbors} = G \cdot \mu_{info}, \tag{7}$$

$$\mu_{recv-neighbors-forward} = G \cdot \sum_{i=1}^{MAXHOP} G^i (\frac{g}{G})^i \mu_{info} = G \cdot \mu_{info} \cdot RecvF, \tag{8}$$

where

$$RecvF = \begin{cases} \frac{g \cdot (g^{MAXHOP} - 1)}{g-1} & g \neq 1 \\ MAXHOP & g = 1 \end{cases},$$

$$\mu_{send-forward} = \frac{g}{G} \cdot [\mu_{recv-neighbors} + \mu_{recv-neighbors-forward}]. \tag{9}$$

In this model, it is assumed that nodes have a FIFO disciplined queue and will not be able to send and receive packets at the same time.

3.1.2 PROCESS MODEL FOR ENERGY HARVESTING AND CONSUMPTION

We model a nanosensor's energy as a continuous-time Markov process, $E(t)$, which describes the transition between energy states. Each state represents a different level of energy. The number of states is $N + 1$, where

$$N = \lfloor \frac{E_{max} - E_{min}}{E_{packet-tx}} \rfloor, \tag{10}$$

and E_{max} and E_{min} are respectively the maximum and minimum energy capacity of nanosensor's energy storage. Recall that $E_{packet-tx}$ represents the energy for one packet transmission. Hence, the energy of the $(n+1)$th state would have a value in $[E_n, E_{n+1})$ where

$$E_n = E_{min} + n \cdot (E_{packet-tx}), \quad n \in [0..N]. \tag{11}$$

For simplicity, the states are named by the lower bound of energy, E_n. Based on this, when the system is in state 1, even though it may have energy for receiving packets, it will not receive since it will not then have energy to transmit. This strict policy is applied to keep the queue of packets stable. In fact, in perpetual nanonetworks with scarce energy, it is better not to transmit or receive in hope of obtaining energy in the future. Otherwise, the reception of many packets when there is no energy for sending them results in many expired packets in the queue of a node or sending information with a very high delay.

Similar to the model in [5], we model the energy harvesting process as a Poisson distribution when ambient vibrations are considered. Yet, we strongly believe that energy harvesting may follow another form. We will discuss the general distribution for energy arrival later in Section 3.1.6. For the energy process, we consider that nanosensors generate new information by the well-known traffic model of Poisson distribution, which has been used in previous work [5].

The Markov process $E(t)$, Figure 15, is defined by its transition rate matrix $Q(t)$, as in (12). For clarity, Figure 15 does not present self transitions, which are the rates of transition to the same state. Each element of the matrix q_{ij} refers to the transition rate from state i to state j and is defined as

$$q_{ij} = \begin{cases} \lambda_i & \forall 1 \leq i \leq N, \quad j = i+1 \\ \mu_i & \forall 2 \leq i \leq N+1, \quad j = i-1 \\ -\sum_{k=1, k\neq i}^{N+1} q_{ik}, & i == j \\ 0 & \text{all other elements} \end{cases},$$

where λ_i and μ_i refer to energy harvesting rates and energy consumption rates, respectively. The rates are defined in the following sections.

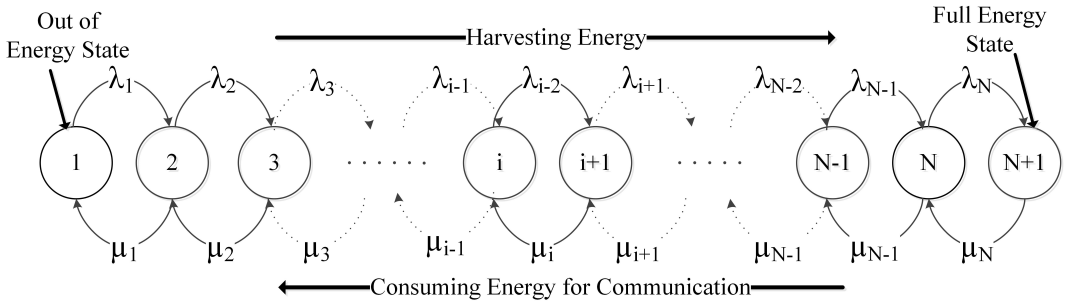


FIG. 15: States and Transitions of Markov Process $E(t)$ for Energy Harvesting and Consumption Process Model.

$$Q(t) = \begin{array}{c} \\ 1 \\ 2 \\ \vdots \\ N \\ N+1 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & \dots & \dots & N & N+1 \end{array} \\ \begin{pmatrix} -\lambda_1 & \lambda_1 & 0 & \cdots & & 0 \\ \mu_1 & -(\lambda_2+\mu_1) & & & \cdots & 0 \\ \vdots & \ddots & \dots & & & \vdots \\ 0 & \cdots & & \cdots & -(\lambda_N+\mu_{N-1}) & \lambda_N \\ 0 & \cdots & & 0 & \mu_N & -\mu_N \end{pmatrix} \end{array}. \quad (12)$$

3.1.3 ENERGY HARVESTING RATE

As introduced in Section 2.5.5, Equation 13 shows the rate of energy harvesting in joules/second that is specified by the vibration rate, current energy level E_{cur} and additional energy ΔE.

$$\lambda(E_{cur}, \Delta E) = \frac{1}{t_{cycle}} . \frac{\Delta E}{n_{cycle}(E_{cur} + \Delta E) - n_{cycle}(E_{cur})}. \quad (13)$$

Based on this equation, the rate for the Poisson harvesting process, that is transitioning from energy state i to state $i+1$, $1 \leq i \leq N$ (gaining enough energy for an additional packet transmission, i.e., $\Delta E = E_{packet-tx}$), is given by

$$\lambda_i = \frac{\lambda(E_i, E_{packet-tx})}{E_{packet-tx}}. \quad (14)$$

Note that division by $E_{packet-tx}$ is required to obtain the rate in $\frac{1}{second}$ units.

3.1.4 ENERGY CONSUMPTION RATE

The consumption rate μ_i is defined as

$$\mu_i = \frac{\sum_x \mu_x}{E_{packet-tx}}, \quad 1 \leq i \leq N, \quad x \in Eqs.\ 6-9, \tag{15}$$

where μ_x is the rate that information is sent or received based on Equations 6-9. Here, the energy consumption rate between two states, i.e., μ_i, is the same for all of the states, although, we will solve our problem for dissimilar values.

After information is sensed, the packet transmission schedule could be easily set to a Poisson process by exponentially randomizing the time between packet transmissions. This way, for any type of traffic such as a burst traffic that is created in the application layer, the queue schedule for packet transmission can still use this randomizing the time between packet transmissions method to generate Poisson traffic[1]. Moreover, the convolution of two or more independent Poisson process, here sending, reception and forwarding traffic processes, is still a Poisson process [105]. So, μ_i would be the rate of a Poisson process. Finally, with assuming $g = 1$ and considering the parameters α and W, which are the portion of $\frac{E_{pulse-rx}}{E_{pulse-tx}}$ (2.6) and the code weight respectively, the rate for energy consumption of all traffic rates is given by

$$\begin{aligned}\mu_i =& \frac{W \cdot E_{pulse-tx}}{E_{packet-tx}}(\mu_{info}+ \\ & \frac{1}{G}[G \cdot \mu_{info} + G \cdot MAXHOP \cdot \mu_{info}])+ \\ & \frac{\alpha \cdot E_{pulse-tx}}{E_{packet-tx}}[G \cdot \mu_{info} + G \cdot MAXHOP \cdot \mu_{info}].\end{aligned}$$

Factorizing will result in

$$\begin{aligned}\mu_i =& \frac{\mu_{info} \cdot E_{pulse-tx}}{E_{packet-tx}} \cdot \\ & [W \cdot (2 + MAXHOP) + \alpha \cdot G \cdot (1 + MAXHOP)].\end{aligned} \tag{16}$$

After replacing the $E_{packet-tx}$,

$$E_{packet-tx} = N_{packet} \cdot W \cdot E_{pulse-tx}, \tag{17}$$

[1]Note that this way of shaping the traffic may not lead to the best utilization of energy harvesting. However, as mentioned before the goal of this model is just to identify the role of various parameters in the combined process of energy harvesting and consumption. Later, in Section 5.4, we remove the Poisson shaping of traffic by controlling the transmission quantity for each time slot, which results in maximizing the utilization of harvested energy.

the final consumption rate would be

$$\mu_i = \frac{\mu_{info}}{N_{packet} \cdot W} \cdot \\ [W \cdot (2 + MAXHOP) + \alpha \cdot G \cdot (1 + MAXHOP)]. \tag{18}$$

This expression shows the main parameters that affect the consumption rate, i.e., N_{packet}, W, G, α. Later, in Chapter 4, we investigate finding optimal values for these parameters to find the highest throughput in combination with other objective functions. The values of μ_{info} and $MAXHOP$ are determined by application requirements. The state probability of the $E(t)$ Markov process, which is the probability of finding the process in any of the states at time t, is defined as $\pi(t) = \{\pi_1(t), \pi_2(t), \ldots, \pi_{N+1}(t)\}$, where

$$\sum_{i=1}^{N+1} \pi_i(t) = 1.$$

3.1.5 STEADY STATE OF NANONODE

Assuming that Markov process rates are stationary, the long term behavior of the nanonode in steady state can be found. Steady states for the defined Markov process can be found based on Kolmogorov forward equation [105]. Solving the system of differential equations provides the probability of being in each of the states. We can write the Kolmogorov equations as follows

$$\begin{aligned} &\lambda_1 \pi_1 = \mu_1 \pi_2 \\ &\lambda_1 \pi_1 + \mu_2 \pi_3 = (\lambda_2 + \mu_1) \pi_2 \\ &\lambda_2 \pi_2 + \mu_3 \pi_4 = (\lambda_3 + \mu_2) \pi_3 \\ &\ldots \\ &\lambda_{N-1} \pi_{N-1} + \mu_N \pi_{N+1} = (\lambda_N + \mu_{N-1}) \pi_N \\ &\lambda_N \pi_N = \mu_N \pi_{N+1} \end{aligned} \tag{19}$$

$$\tag{20}$$

where $\pi_i \in [0, 1], \quad 1 \leq i \leq N + 1$, represents the steady state probability of the

process. From the above equation system, it can be written

$$\pi_i = \prod_{j=1}^{i} \frac{\lambda_1 \cdots \lambda_{j-1}}{\mu_1 \cdots \mu_{j-1}} \cdot \pi_1 \quad 2 \leq i \leq N+1 \tag{21}$$

$$\text{and } \pi_1 = \frac{1}{\sum_{i=2}^{N+1} \prod_{j=1}^{i} \frac{\lambda_1 \cdots \lambda_{j-1}}{\mu_1 \cdots \mu_{j-1}}}, \tag{22}$$

where λ_i is the energy harvesting rate from Eq. 14, and μ_i is the energy consumption rate from Eq. 18. In the current model, all μ_i are the same, but the presented solution is a general form that can be used in the case where the rates need to be adapted based on available energy. The state that we are interested in is the first state where the energy level is not enough to send or receive any packet. Thus, it can be used to calculate the probability of unsuccessful transmission and other network performance metrics. These will be described in Section 3.1.7.

3.1.6 CASE OF GENERAL ENERGY ARRIVAL

In this section, we evaluate the scenario where the energy arrival process does not follow the Poisson distribution introduced in Section 3.1.2. In fact, we evaluate the energy arrival that follows a general (arbitrary) distribution. A deterministic distribution, uniform distribution or normal distribution are samples of a generic distribution that could be found in some circumstances of energy arrival, particularly for vibration sources of energy harvesting [75, 76, 77, 98]. For example, the heartbeat rate in normal conditions of a human or almost constant wind speed provides a constant distribution.

We use the *embedded Markov chain* method [105] for solving the problem as follows. The number of states are as before and are defined based on the energy required for the transmission of N packets. Then, the transition rates between states are given by the following $(N+1) \times (N+1)$ matrix

$$F = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ \vdots \\ N \\ N+1 \end{array} \begin{array}{c} \begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & \cdots & N+1 \end{array} \\ \begin{pmatrix} 1-\beta_0 & \beta_0 & 0 & 0 & 0 & \cdots & 0 \\ 1-\sum_{i=0}^{1}\beta_i & \beta_1 & \beta_0 & 0 & 0 & \cdots & 0 \\ 1-\sum_{i=0}^{2}\beta_i & \beta_2 & \beta_1 & \beta_0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & & \\ 1-\sum_{i=0}^{N-1}\beta_i & \beta_{N-1} & \beta_{N-2} & \beta_{N-3} & \beta_{N-4} & \cdots & \beta_0 \\ 1-\sum_{i=0}^{N-1}\beta_i & \beta_{N-1} & \beta_{N-2} & \beta_{N-3} & \beta_{N-4} & \cdots & \beta_0 \end{pmatrix} \end{array},$$

where $0 < \beta_i < 1$ and $\sum_i \beta_i = 1$.

The β_i rates are defined based on the amount of energy in a nanonode when an energy arrival occurs. For example, if an arrival finds energy available for sending two packets (row 3 of the matrix), the next arrival of energy will find the state with one these conditions: (I) energy for sending three packets (with probability β_0, there was no energy consumption between the two energy arrivals), (II) energy for sending two packets (with probability β_1, there was a single transmission between the two arrivals of energy), (II) energy for sending one packet (with probability β_2, there were two transmissions between the two arrivals of energy) or (IV) no energy for sending packets (with probability $1-\sum_{i=0}^{2}\beta_i$). The last two rows are identical because when an arrival finds the states with energy for transmission of either $N-1$ or N packets, after the arrival, the energy reaches the maximum capacity. Therefore, the next arrival energy can find any number between N and 0 according to probabilities β_0 to $1-\sum_{i=0}^{N}\beta_i$. Due to the Poisson model of consumption, β_i is found as follows:

$$\beta_i = \frac{\mu_*^i}{i!}e^{-\mu_*}, \tag{23}$$

where μ_* is the consumption rate from Equation 18.

Then the probabilities π_i, $i = 1, \ldots, N+1$, which denote the probabilities of an arrival finding an energy level of $i-1$, can be obtained from the system of equations $\pi(F - I) = 0$ by a *recurrence procedure* where I is the identity matrix. The last equation of the system is

$$\pi_{N+1} = \beta_0\pi_N + \beta_0\pi_{N+1}. \tag{24}$$

Next, if a value is assigned to the last component of the solution, such as $\pi_{N+1} = 1$ for simplicity, the next component can be found as

$$\pi_N = \frac{1-\beta_0}{\beta_0}. \tag{25}$$

The next equation would be

$$\pi_{N-1} = \frac{(1-\beta_0)\pi_N}{\beta_0} - \frac{\beta_1}{\beta_0}\,, \tag{26}$$

and so on. After finding all π_is in this manner, a normalization is forced to make the sum of all components be equal to 1, which yields the stationary probability distribution of the energy state at energy arrival epochs of the system.

Therefore, we can find the probability of being in different state of energy, π_is. The most important one is π_1, which is the probability of being in state with no energy to transmit any packet. Recall that we also set a policy not to receive any packet in state π_1.

3.1.7 EVALUATION METRICS

Delivery Rate

Based on π_1, the probability of unsuccessful transmission between two nodes due to lack of energy in the receiver is defined as $p_{drop-trx} = \pi_1$. The transmission can also be unsuccessful due to collision or absorption. The probability of no collision in simultaneous transmission of pulses of neighbors, G, would be

$$p_{no-coll} = (1 - \mu_{trans} e^{-\mu_{trans}})^{G \cdot N_{packet}}, \tag{27}$$

where

$$\mu_{trans} = (\mu_{info} + \mu_{send-forward}) \cdot W \cdot \frac{T_p}{T_s}\,.$$

Recall that T_p and T_s are the symbol duration and the interval between symbols, respectively. The probability of unsuccessful transmission because of absorption is $p_{error} = 1 - (1 - BER)^{N_{packet}}$, where BER is the bit error rate equal to 10^{-4} at 10 mm distance [5, 12]. Next, we define the probability of a successful transmission $p_{success}$ if (I) all of the neighbors have enough energy to receive; (II) no collision or error due to absorption occurs; and (III) neighbors are in the idle state, i.e., are able to receive because we assumed that nodes cannot transmit and receive at the same time

$$p_{success} = (1 - p_{drop-trx})^G \cdot (1 - p_{error}) \cdot (p_{no-coll}) \cdot (p_{idle})^G\,.$$

We show that p_{idle}, the probability of being in the idle state, can be computed by knowing the rate of transition from busy to idle, μ_{idle}, and from idle to busy, μ_{busy}, as follows.

$$\mu_{idle} = \frac{1}{N_{packet} \cdot T_p}, \quad \mu_{busy} = \mu_i \,.$$

Next, by solving a simple two states Markov process

$$p_{busy} = \frac{\mu_{busy}}{\mu_{idle} + \mu_{busy}} \quad \text{and} \quad p_{idle} = \frac{\mu_{idle}}{\mu_{idle} + \mu_{busy}} \,.$$

Evaluation based on $p_{success}$ is important because information in nanonetworks would be delivered through a multi-hop mechanism. In other words, end-to-end successful delivery, $e2e_{delivery}$, for N_{hop} hops would be

$$e2e_{delivery} = {p_{success}}^{N_{hop}} \,, \tag{28}$$

where $1 \leq N_{hop} \leq MAXHOP$, which is determined as a design parameter.

Delay

The other metric for performance evaluation is end-to-end delay. It includes the delay for the propagation of a packet D_{prop}, packet transmission time D_{trans}, the delay imposed by retransmission due to collisions and absorption D_{error}, and finally transfer delay because of lack of energy for sending or receiving $D_{no-energy}$. The values of D_{prop} and D_{trans} are not significant because of the short packet size and short pulse duration in nanonetworks. The value of D_{error} will exist only when there is a transmission error, which has a low probability. The value of D_{error} also depends on the mechanism for handling the retransmission. Because of the low probability of error, we ignore its delay. In this perpetual network, the main delay is posed by lack of energy, $D_{no-energy}$. In fact, the lack of energy implies a requirement of time until the node harvests enough energy to be able to send or receive packets. In this regard, the average delay due to lack of energy, $D_{no-energy}$, can be calculated as the staying time of Markov process, $E(t)$, in its first state, which is the state with no energy. Based on the definition of staying time, it would be the probability of being in the first state $\times$ time it takes to go to the next state. Therefore, it can be calculated as

$$D_{no-energy} = \pi_1 \frac{1}{\lambda_1} \,. \tag{29}$$

Utilization of Nanonode

A lower traffic rate allows the nanonode to avoid low energy states; however, there would be some energy that is not fully utilized. On the other hand, a high traffic rate would result in too many dropped packets due to lack of energy. So, first a definition for utilization is required. Second, an optimum utilization value needs to be specified. We define the energy consumption intensity, ECI, as follows.

$$ECI = \frac{\sum_{i=1}^{N} \mu_i}{\sum_{i=1}^{N} \lambda_i}. \tag{30}$$

Recall that μ_i is the consumption and λ_i is the harvesting rate of Markov process $E(t)$. If ECI is larger than 1, it means that the traffic load is too high and with high probability, the nanonode will be in a low energy state. On the other hand, a very low ECI means that the traffic load is low and the nanonode is underutilized. Utilization ρ is equal to ECI when $ECI \leq 1$.

3.2 EXPLORING THE ENERGY CONSUMPTION AND HARVESTING MODEL

In this section, we compare the analytic energy model versus simulation in MATLAB. The goal of this section is to evaluate and to study the behavior of the energy model. We investigate optimal parameter values in Chapter 4. Nanosensors harvest vibrational energy with a vibration rate of 50 cycles per second (50 Hz). The energy capacity E_{max} is 800 pJ, and the minimum energy E_{min} that a node will have in its first state is 5 pJ. Detailed parameters of the capacitor are similar to [5]. Each nanosensor generates new data that is composed of a Poisson arrival with parameter μ_{info} which are chosen between 1 to 20 kbits per second. The packet length N_{packet} is varied from 96 to 196 kbits. Nanonodes communicate based on the TS-OOK pulse method. The pulse duration is 100 femtoseconds and the time between symbols (pulses or silences) is 100 picoseconds. Based on the numerical results provided in [12] and modeling in [5], the energy consumption for the transmission of a pulse, $E_{pulse-tx}$, is set to 1 fJ and for the reception of a pulse, $E_{pulse-rx}$, is set to 0.1 fJ. The number of neighbors, G, is selected from 2-8. This number of neighbors represents the typical sizes of networks in a mesh topology and has been used in studies such as [103] and [104]. Also, it is a reasonable value even in scenarios where the topology is controlled, e.g., [106] and [107]. The number of neighbor nodes that will forward g and $MAXHOP$ are set to 1 and 3, respectively. Table 9 summarizes the evaluation

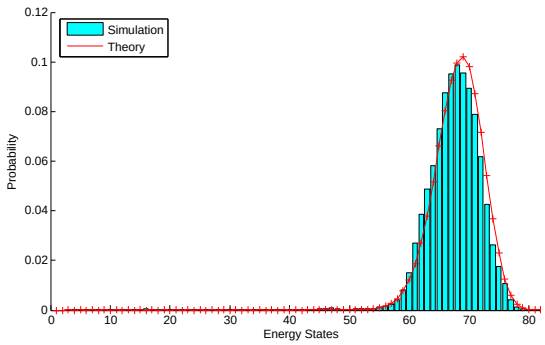
TABLE 9: Evaluation Parameters Values.

Parameter	**Value**
Vibration rate	50 Hz
E_{max}	800 pJ
E_{min}	5 pJ
N_{packet}	96 or 196 kbits
$E_{pulse-tx}$	1 fJ
$E_{pulse-rx}$	0.1 fJ
G	2-8
g	1
$MAXHOP$	3
μ_{info}	1-20 kb/s

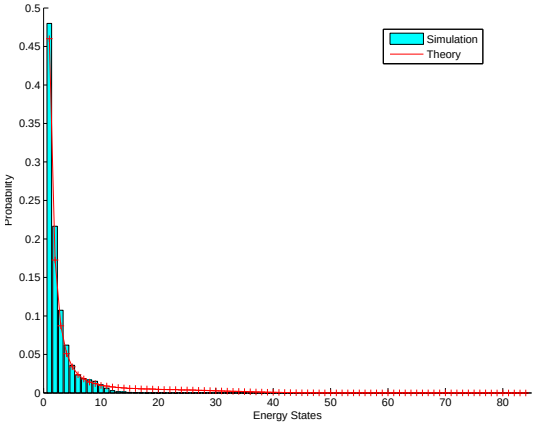
parameters.

The steady state of the model is compared with the normalized histogram of the energy evolution over time in the simulations. Figures 16(a)-16(f) show the histograms of the probability of being in different energy states for different configurations when the energy arrivals follow a Poisson process, as described in Section 3.1.3. Note that the number of states are different for various code weights since the energy for the transmission of packets, which specifies the number of states, depends on the code weight. As a reminder, being in energy state i means that the node has enough energy for $i-1$ packet transmissions. It is clear that with lower data rates, lower number of neighbors and lower code weight (W), the nanonode is more often found in higher levels of energy (Figures 16(a) and 16(f) in comparison to others). In each of two consecutive figures only one parameter is changed. Therefore, the effect of G, code weight, and μ_{info} can be viewed. Also, ECI is shown for each scenario, which indicates if the nanonode is overloaded or not. Indeed, with ECI, it can be specified if a nanonode can accept higher load, i.e., higher μ_{info}, without being in low energy states with a high probability. The other observation is that parameters such as code weight can greatly affect the histogram (Figures 16(a) and 16(e)). So, finding the best combination of traffic rate, code weight, and packet size is a critical evaluation point, which we will address in Chapter 4.

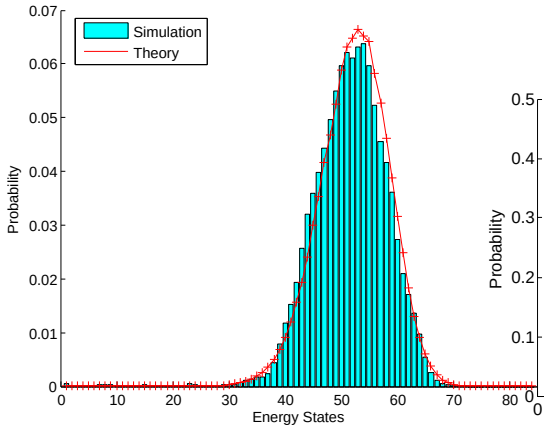
The effect of the number of neighbors, G, on the probability of being in the first state, π_1, is shown in Figure 17. It can be viewed that the worst scenario happens when G is 8, which is because of the high generated traffic load.



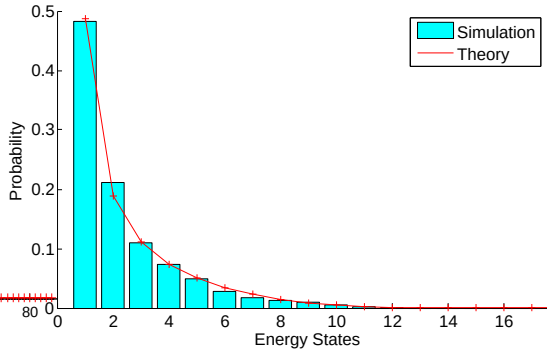
(a) μ_{info} = 8 kbits/s, W = 0.1, G = 2, ECI = 0.49



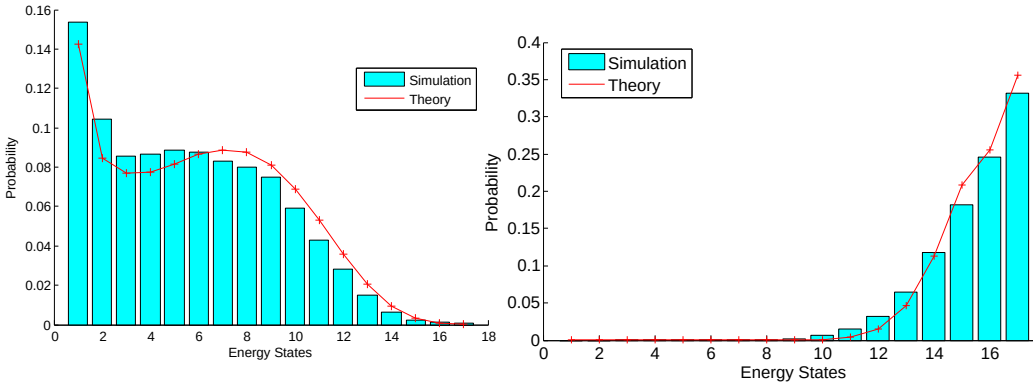
(b) μ_{info} = 8 kbits/s, W = 0.1, G = 8, ECI = 1.57



(c) μ_{info} = 5.5 kbits/s, W = 0.1, G = 8, ECI = 0.97



(d) μ_{info} = 8 kbits/s, W = 0.5, G = 8, ECI = 2.08



(e) μ_{info} = 8 kbits/s, W = 0.5, G = 2, ECI = 1.20

(f) μ_{info} = 1 kbits/s, W = 0.5, G = 8, ECI = 0.26

FIG. 16: Probability of Being in Different Energy States.

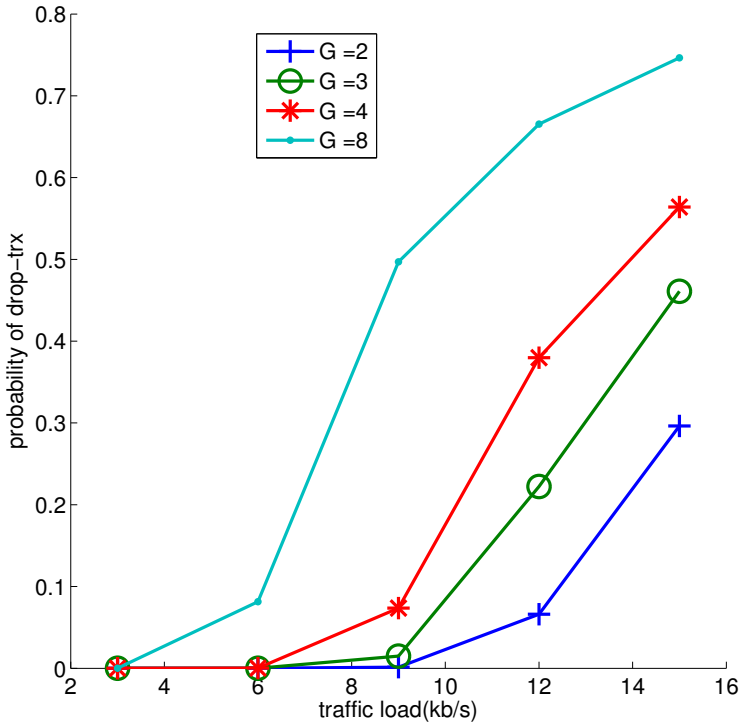


FIG. 17: Effect of Number of Neighbors, G, on the Probability of Drops for Different μ_{info} with Packet Size = 96 kbits, W = 0.3.

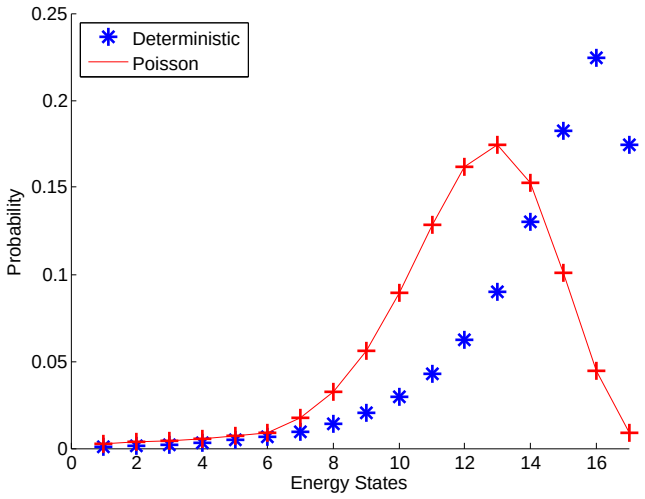


FIG. 18: Comparison of General vs. Poisson Distribution Energy Arrival, with μ_{info} = 4 kbits/s, W = 0.5, G = 8.

Figure 18 compares the Poisson energy arrival with that of the general process. The general distribution is a deterministic distribution that has rate with value λ_i. As can be viewed, the probability of being in first energy state for the general model is less than that for the corresponding Poisson model. In the case of the general distribution, as long as the traffic load rate is not more than the harvest rate, then π_1 is less than or equal to that in the Poisson process. The reason is that in the Poisson process energy inter-arrival (which are exponential) can be very long, resulting in the probability of being in the no-energy state being higher than for a deterministic process. Since the results show that the Poisson process can be an upper bound on the general distribution, from here on we discuss only the Poisson model.

3.2.1 DELIVERY RATE

Figure 19 shows the probability of successful transmission, $p_{succcess}$, for various loads as the code weight is varied. A significant difference between code weights of 0.5 and 0.1 exists. A code weight of 0.1 can have a successful probability close to 1 for 75% higher load than with $W = 0.5$.

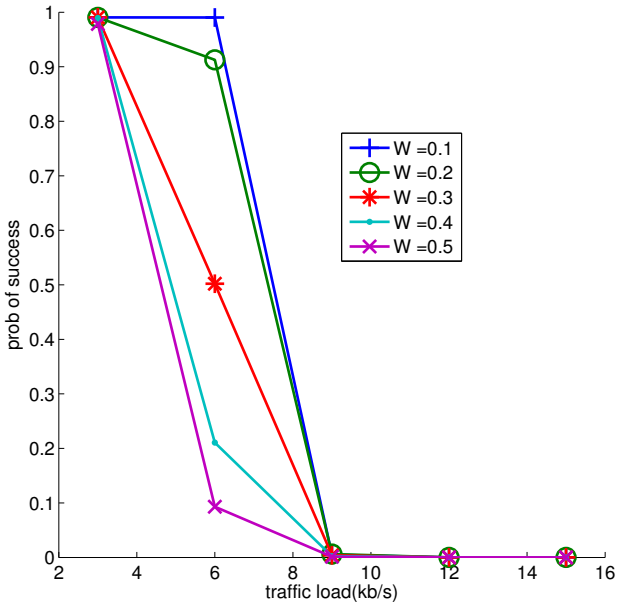


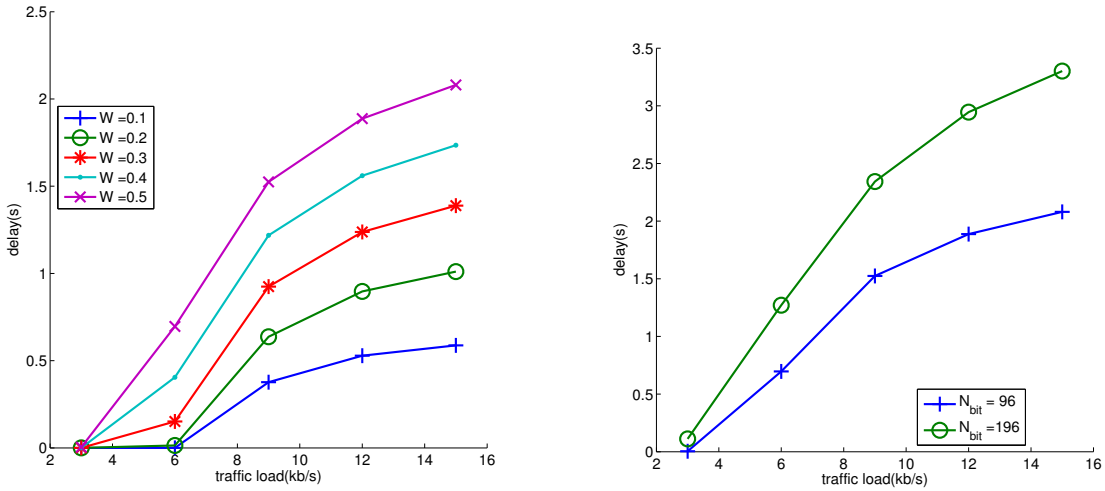
FIG. 19: Probability of Successful Transmission for Different μ_{info} and W, with Packet Size = 96 kbits, G = 8.

3.2.2 DELAY

The average delay due to lack of energy is shown in Figures 20(a)-20(b). In Figure 20(a), delay is limited to 0.5 seconds when $W = 0.1$. Even though generally the lower code weight is more efficient, the bit rate transmitted is reduced with a lower code weight. As shown in Figure 20(b), the difference in delay is almost doubled when a longer packet is used. This is because more time is required for the node to harvest the energy to be able to send a longer packet. Moreover, recall that this delay only includes the average delay for transmission of packets due to lack of energy. In other words, in high traffic loads, most of the time there is not enough energy for transmission, and when the transmission occurs (there is enough harvested energy for transmission), transmission will have a low chance of success due to the high probability of the receiver being in an energy state that cannot receive packets.

3.2.3 ENERGY CONSUMPTION INTENSITY

The *energy consumption intensity* (ECI) is shown in Figure 21. The higher the code weight and μ_{info}, the higher the slope of ECI.



(a) Packet Size = 96, G = 8 (b) W = 0.5, G = 8

FIG. 20: Average Delay.

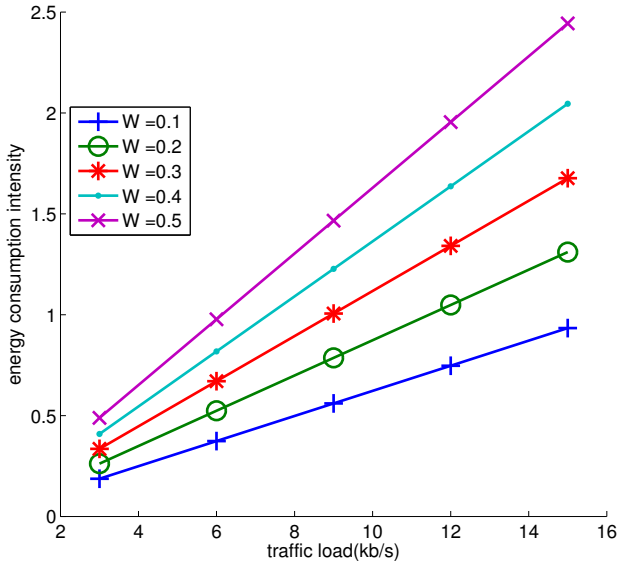


FIG. 21: ECI for Various Code Weight, with Packet Size = 96, G = 8.

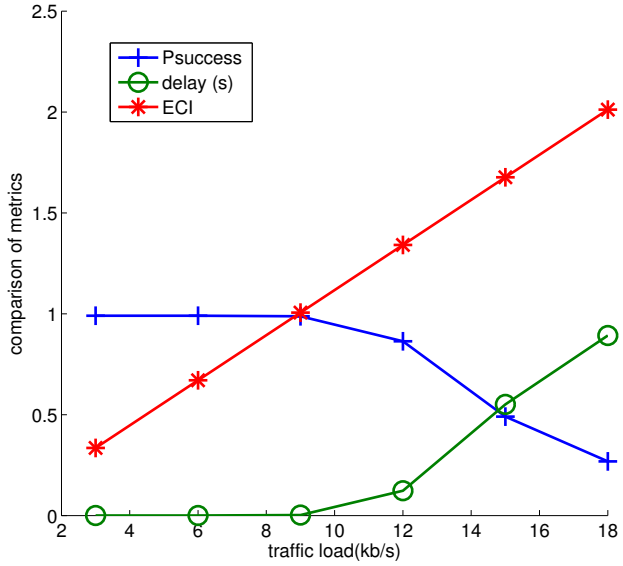


FIG. 22: Comparison of Various Metrics, with Packet Size = 96, W= 0.3, G = 2.

Figure 22 shows that ECI is a well-defined metric. When $ECI \leq 1$, the delay and probability of success are acceptable. After $ECI > 1$, the delay has a jump as well as a drop in the probability of success.

By knowing ECI for each set of configurations (packet size, code weight), it is possible to evaluate the effect of traffic load on network performance. The next step is to find the optimal configuration. We will investigate this issue in Chapter 4.

3.3 SUMMARY

In this chapter, we introduced a model for the evaluation of energy harvesting and consumption processes. Our model could reveal the effect of various parameters. Also, it is general enough to be used for any energy arrival model. This model inspires the optimization of energy consumption. This model shows that two aspects of the process can be optimized: packet formation and packet scheduling. We will investigate how to optimize each of these in Chapters 4 and 5.

CHAPTER 4

OPTIMIZING ENERGY CONSUMPTION IN PACKET FORMATION

4.1 OPTIMIZING THE FACTORS OF ENERGY CONSUMPTION

As described in Chapter 3, various parameters can affect the model of energy harvesting and consumption. Particularly, packet size, code weight, and repetition can affect the amount of energy that is consumed. The introduced performance metrics in Section 3.2, i.e., ECI, delay and $p_{success}$, are helpful to study the behavior of a nanonode. However, it is also required to know the optimal achievable performance. In other words, repetition and code weight should be selected in a way that provides an efficient bit rate. Therefore, finding the optimum design point between energy usage efficiency and bit rate efficiency is the challenge that is addressed in this section. We first describe a model that can find the best combination of these parameters. Then, we show how the best answer could be selected among a list of candidates when traffic load and utilization are taken into account.

4.2 OPTIMIZATION MODEL

We model the problem as a *Multi-Objective Combinatorial Optimization* (MOCO), a special form of Multi-Objective Optimization (MOP) [108], where variables can take discrete values. In a MOP/MOCO problem, several functions need to be optimized at the same time. Then, instead of having a unique solution to the problem, the solution is a possibly infinite set of Pareto points. These points are called *Pareto optimal*. The general form of a MOCO is

$$
\begin{aligned}
& min_x \left[f_1(x), f_2(x), \ldots, f_n(x)\right]^T \\
& \text{s.t.} \\
& g(x) \leq 0, \\
& h(x) = 0, \\
& x_l \leq x \leq x_u,
\end{aligned}
$$

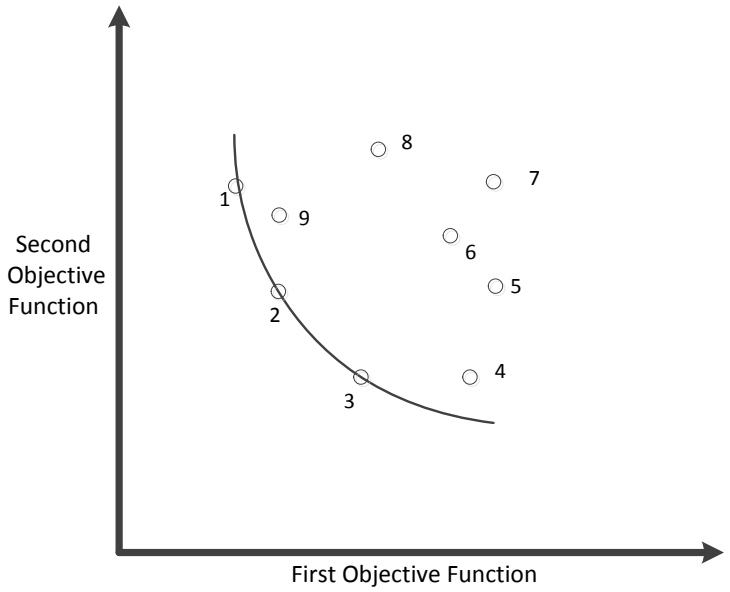


FIG. 23: Pareto Points Among Other Solution Points for a Two Objective Function Problem.

where f_i is the i-th objective function, g and h are the inequality and equality constraints, respectively, and x is the vector of optimization or decision variables. The solution to the above problem is a set of Pareto points. A design point in objective space f is termed *Pareto optimal* if there does not exist another feasible design objective vector f^* such that $f_i \leq f_i^*$ for all $i \in \{1, 2, \ldots, n\}$, and $f_j < f_j^*$ for at least one index of j ,$j \in \{1, 2, \ldots, n\}$. Figure 23 represents the Pareto points among all possible solutions for a two objective function problem. The problem requires the minimization of both objective functions. The curved line represents the Pareto-front, identified by non-dominated solutions that have labels 1, 2, and 3. The other points are not optimal Pareto points because they have a higher value in at least one of the objective functions. For example, point 4 has higher value than point 3 for the first objective function and point 9 has a higher value than point 2 for the second objective function. Clearly, representing and finding the solution of a MOCO problem, when the number of variables, number of objective functions, and search space is larger, is more challenging.

In our problem, the functions to be optimized at the same time for the N_{packet}, *repetition*, and W variables are defined as follows:

$$
\begin{aligned}
&Min_v \left[f_1(v), f_2(v), f_3(v), f_4(v), f_5(v)\right]_T \\
&\quad s.t. \\
&\quad g_1(v) \leq 0, \\
&\quad g_2(v) \leq 0, \\
&\quad v = [N_{packet}, repetition, W], \\
&\quad W \in (0.15 : 0.05 : 0.5), \\
&\quad repetition \in (1, 3, 5), \\
&\quad 1 \leq N_{packet} \leq 2500\,.
\end{aligned}
$$

The first function is energy consumption, that is, the energy consumed for transmission (Eq. 4) plus reception (Eq. 5) of a packet by all the neighbors with N_{packet} data.

$$
\begin{aligned}
f_1 &= \frac{E_{packet-tx} + G \cdot E_{packet-rx}}{N_{packet}} = \frac{m' \cdot W \cdot E_{pulse-tx} + G \cdot m' \cdot \alpha \cdot E_{pulse-tx}}{N_{packet}} \\
&= \frac{m' \cdot E_{pulse-tx}}{N_{packet}} \cdot (W + G \cdot \alpha),
\end{aligned}
$$

where α = ratio of energy for pulse reception to transmission, G = the number of neighbors, W = code weight, and $m' = N_{packet}+a$, where a is the number of additional bits added to N_{packet} that enables coding with code weight W. We developed the model in the general form that there are G neighbors. Therefore, it would cover most unicast or broadcast scenarios where the packet will be received by one, some, or all of the neighbors. Moreover, a preamble or handshake method could be deployed to avoid reception of a packet by all neighbors when it is not targeted for them. This objective function is set to be minimized, which means that the total energy that is consumed for transmission and reception per bit of information should be minimized.

The next objective function concerns delay. Since N_{packet} is larger than the information generation rate, the packet would contain several pieces of information together to avoid the overhead of packet transmission. However, this increases the delay in transmission of information. For example, if information is generated at 10 bits per seconds and the packet size is 1000 bits, it means that it will take 100 seconds to prepare a packet. This may be acceptable for non-real time applications, or when the rest of packet can be filled with neighbors' forwarding data, or can just

be left empty. However, in our model, we are assuming that there is a limit on the delay in packet. The simplest way to define the delay function is to model it in a linear relation with packet length, N_{packet}. However, if delay has higher importance, the function could be modeled as a higher degree polynomial function of N_{packet}.

$$f_2 = N_{packet}\,. \tag{31}$$

This function is set to be minimized.

The next objective function associates the chance of bit error with code weight code. A lower code weight means transmission of fewer 1s, which results in a lower probability of absorption as well as collision between 1s.

$$f_3 = W\,. \tag{32}$$

This function is set to be minimized.

The optimization problem can be formulated with only the $f1$, $f2$ and $f3$ functions, if repetition is not required to be considered as a variable. This could be the case if it is known that the environment would not affect the pulses significantly and it is better to repeat the whole packet in case of error rather than consume energy with the repetition of symbols. However, we define the functions for repetition to have a comprehensive model.

The following function shows the effect of repetition. The higher the repetition, the higher the chance of error detection and recovery.

$$f_4 = \lfloor \frac{\frac{repetition-1}{2}}{repetition} \rfloor\,. \tag{33}$$

On the other hand, lower repetition means fewer bits and less energy consumption.

$$f_5 = \frac{N_{packet}}{repetition}\,. \tag{34}$$

This function actually shows the efficient bit rate when repetition is used, and it should be maximized.

The constraint functions would be

$$g_1 = m' \cdot W \cdot E_{pulse-tx} - E_{max} + E_{min} \leq 0\,,$$
$$g_2 = m' \cdot E_{pulse-rx} - E_{max} + E_{min} \leq 0$$

This means that the energy for transmission or reception of one packet cannot exceed the maximum energy capacity of the node.

The bounds on the variables of the problem are defined as follows:

$$W \in (0.15 : 0.05 : 0.5)\,,$$
$$repetition \in (1, 3, 5) \text{ and}$$
$$1 \leq N_{packet} \leq 2500\,.$$

Note that because the problem is a combinatorial problem, the bounds are actually the set of valid values that can be assigned to variables, i.e., W and *repetition*. For N_{packet}, in addition to the bounds, the values should be discrete.

4.3 OPTIMIZATION PROBLEM SOLUTION

Various methods are used to solve MOCO problems, such as the method of objective weighting or min/max formulation [109, 108]. In some specific problems, it is possible to merge multiple objectives into one objective so that the resulting solution depends mainly on the weight vector assigned to each objective [109]. As a result, the same problem must be solved several times for different weight vectors.

Another way to solve multiobjective optimization problems is to use Genetic Algorithms (GA). Since GAs search for the optimal solutions based on a population of points instead of a single point, they can find multiple Pareto optimal solutions in a single run. It helps decision makers to choose the best solution from set of Pareto optimal points based on the situation. In fact, it removes the burden and common difficulty with multi-objective optimization in balancing different objective needs.

Finding the Pareto optimal set is computationally intensive and requires efficient methods. GA-based multi-objective optimization tools such as Non-dominated Sorting Genetic Algorithms (NSGA), Strength Pareto Evolutionary Algorithm (SPEA), and Non-dominated Sorting Genetic Algorithms II (NSGA-II) [109] have been developed to solve MOP problems efficiently. Among all, NSGA-II has the most promising results. NSGA-II is categorized as an elitist genetic algorithm. An elitist GA always favors individuals with better fitness value (rank). A controlled NSGA-II is a variation of NSGA-II that also favors individuals while helping to increase the diversity of the population even if they have a lower fitness value. It is important to maintain the diversity of the population for convergence to an optimal Pareto front. Diversity is maintained by controlling the elite members of the population as the algorithm progresses. Two parameters, *Pareto Fraction* and *Distance Function*, control the elitism. The Pareto Fraction limits the number of individuals on the Pareto front

(elite members). The Distance Function maintains diversity on a front by favoring individuals that have relatively far away distance.

We use the controlled NSGA-II to solve our MOCO problem. As mentioned before, the output of MOCO would be a set of Pareto optimal points. Typically, the selection of one point depends on the application and context that a decision maker is facing.

4.4 SIMULATION

We solved our defined MOCO problem with the optimization toolbox of MATLAB. We customized the creation, mutation, and crossover functions of model. The simulation parameters for our MOCO solution with the NSGA-II method are shown in Table 10.

TABLE 10: MOCO Problem Parameters.

Parameter	**Value(s)**
Population size	100
Pareto fraction	0.2
Generations	150
Selection	Uniform
Crossover fraction	0.8
Mutation function	Uniform
Crossover function	Three parents

We ran the optimization with different values for α, G, and *repetition* to show the effect of these parameters on the points that are selected as optimum. The result for each of the configuration scenarios, listed in Table 11, are presented in the following subsections. Note that Pareto optimal points are not unique and even can be different in several runs. However, the results that are presented here have a similar pattern for all runs and different runs give only non-significant bit differences in packet size.

4.4.1 SCENARIO 1 (G=1, α =0.15, REPETITION =1)

In this scenario, we set G to 1 and α to 0.15. This scenario will evaluate the case of transmission between two adjacent nodes when broadcast will result only in reception by one neighbor. The α value is set to 0.1, based on the numerical values in [12] and modeling in [5].

Figure 24 shows the Pareto optimal points that are selected. This scatterplot

TABLE 11: Scenario Parameters.

Scenario	G	α	**max repetition**
1	1	0.1	1
2	1	0.5	1
3	4	0.1	1
4	1	0.1	5
5	4	0.1	5

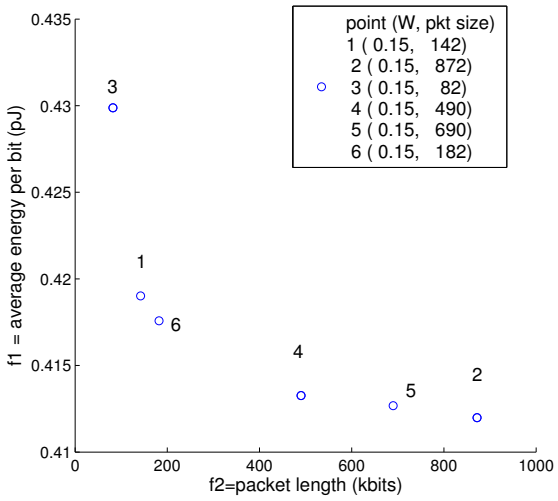


FIG. 24: Pareto Point and Function Values for Scenario 1.

represents the value of first and second objective functions for each of Pareto points. The code weight and the packet length for each of the points are presented in the legend. Recall that the first objective function tries to minimize the amount of consumed energy per bit. On the other hand, the second function, minimizing delay, is related to packet length. Each of these points dominates the other one in one of the two objective functions. Therefore, depending on design priority, any of these points can be selected as the optimal solution. For example, if the priority is energy consumption, one of the points in the lower-right of the chart could be selected. If delay has priority, one of points in the left side of chart would be the choice.

Figure 24 also illustrates that various packet lengths are selected. A deeper look at the selected code weight for these points shows that all of them are equal to 0.15, which is the minimum code weight. It means that with this setting for G and α, it is better to choose the minimum code weight that is available.

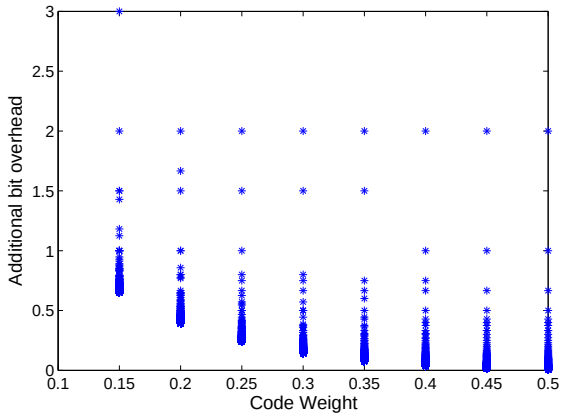


FIG. 25: Additional Bit Overhead for Various Code Weights.

Figure 24 also shows that the difference in terms of efficient energy per actual information bit, f_1, is not significantly different among all the optimal points. This observation can be confirmed by the fact that for a selected code weight, usually the ratio of additional bit rate to actual bits, i.e., $\frac{a}{N_{packet}}$, illustrated in Figure 25, is almost the same for each code weight independent of N_{packet}.

Figure 25 also shows that overhead from code weight generally does not depend on the length of data. The figure illustrates data lengths that are in [1..1000] range. Outliers occur when the number of original bits is very small, i.e., less than 10 bits. These short data length is not applicable in packet transmission.

4.4.2 SCENARIO 2 - EFFECT OF α

In this scenario, we increase the value of α to 0.5. Recall that α is the $\frac{E_{pulse-rx}}{E_{pulse-tx}}$. Note that in reality, α is fixed. Here we are only evaluating scenarios for different α to show its effect on the optimization problem. The selected optimum points now cover a wide range of various code weights and packet sizes, as shown in Figure 26.

This easily shows that code weight is more effective for smaller values of α. In fact, when α becomes larger, the effect of code weight is reduced. This happens because the overhead bits from the lower code weight increase the reception energy, which eventually increases the average energy per bit. It is worthwhile to mention that because there are more than two objective functions, the optimal points cannot be chosen from the figure. Otherwise, if there were only these two functions, the optimum point would be one of the points in the bottom-left of Figure 26.

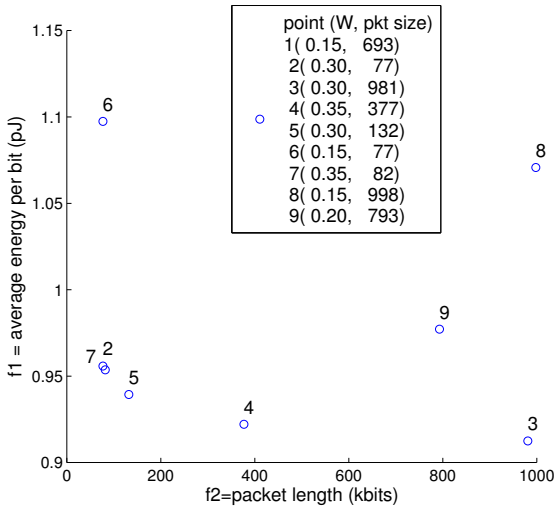


FIG. 26: Pareto Point and Function Values for Scenario 2.

4.4.3 SCENARIO 3 - EFFECT OF G

In this scenario, the effect of G on the selection of optimal points can be viewed in Figure 27, when G is changed from 1 to 4. α is set to 0.1 as described in Table 11. The main observation is that more points with lower code weights are selected since there are more recipients, which makes it efficient to use a lower code weight. This effect can be viewed also in the average energy per bit function, f_1, where it is almost twice Scenario 1 (with one neighbor) while the number of neighbors has increased four times.

4.4.4 SCENARIO 4 - EFFECT OF REPETITION

This scenario takes into account the effect of repetition as another variable. A maximum of 5-repetition is allowed. As shown in Table 11, this scenario is similar to Scenario 1, as G and α are set to 1 and 0.1 respectively. Figure 28 shows the selection of optimum points. Points with various ranges of values for repetition, code weight and packet length are selected. This behavior is mainly due to the dominance of one of the objective functions. For example, point number 3 is selected because it provides a low average energy per bit. On the other hand, point 7 is selected because it provides high reliability with 5-repetition even though it has higher delay, i.e., packet length, and higher average energy per bit in comparison to other points.

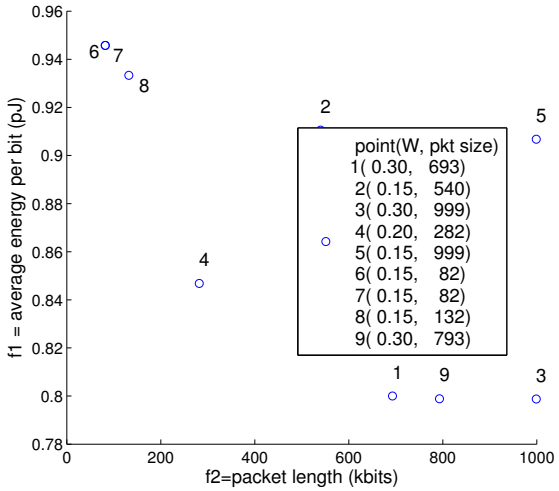


FIG. 27: Pareto Point and Function Values for Scenario 3.

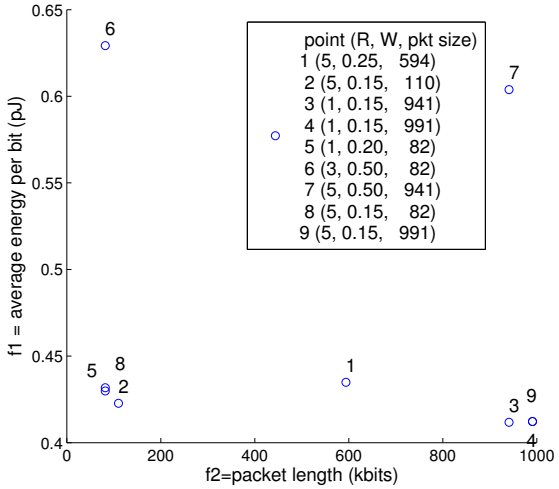


FIG. 28: Pareto Point and Function Values for Scenario 4.

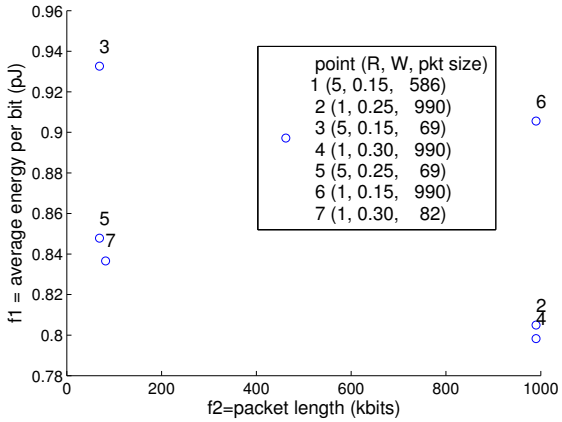


FIG. 29: Pareto Point and Function Values for Scenario 5.

4.4.5 SCENARIO 5 - REPETITION FOR HIGHER G

The fifth scenario evaluates the effect of repetition in combination with a higher number of neighbors (from one to four). As indicated in Table 11, the maximum repetition and α are set to 5 and 0.1, respectively.

In this scenario, the optimal points, as illustrated in Figure 29, are selected from almost all ranges of code weight and repetition. However, packet length values are mainly chosen from very short or very large packet sizes. The reason is that when a short packet size is selected, the energy bit efficiency and delay will be dominant functions. On the other hand, for large packet sizes, code weight will be the dominant factor that leads to lower average energy per bit.

4.4.6 SELECTION BASED ON PERFORMANCE

After finding the Pareto optimal points (illustrated in Figure 29) of our MOCO problem, we compute $p_{success}$ for each one with a high traffic load, i.e., $\mu_{info} = 10$ kbit/s. As mentioned before, Pareto optimal points are not unique and can even be different in several runs. However, the results that are presented here have a similar pattern for all runs and different runs give only non-significant bit differences in packet size.

The last column of Table 12 shows the scenarios that will have $p_{success} \geq 0.98$. It can be inferred that while the selected points belongs to various packet sizes, the selected code weights are mainly 0.15 and 0.3, and repetition with 1 and 5 could be

TABLE 12: Pareto Optimal Points.

Repetition	**Weight Code**	**Packet Size(kb)**	$p_{success} \geq$ **0.98**
1	0.15	990	Y
1	0.3	82	Y
5	0.15	69	Y
5	0.15	586	Y
5	0.25	69	Y
1	0.25	990	N
1	0.3	990	N

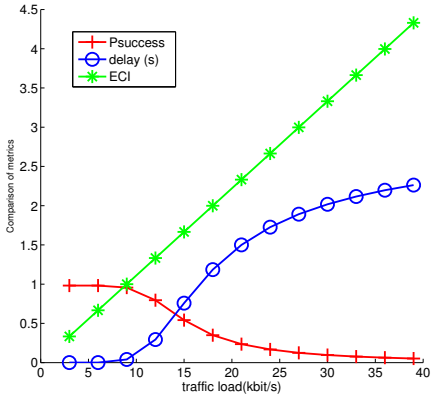


FIG. 30: Comparison of Success, Delay and Intensity versus Different Loads.

found.

Similar to selection based on $p_{success}$, $D_{no-energy}$ or ρ could be used as a performance selection metric. Also, for each optimum point, we can specify the maximum traffic load that the network could handle while keeping the utilization close to 1. Figure 30 shows the ECI, $p_{success}$, and delay for various traffic loads. In this scenario, the packet size is 82 kb, code weight is 0.3, and repetition is 1. When load is about 10 kb/s, both ECI (intensity) and $p_{success}$ are close to 1.

4.5 SUMMARY

This chapter investigated simultaneous optimization of energy consumption and the some requirements of perpetual wireless nanosensor networks. Code weight and repetition as parameters to reduce energy consumption and increase transmission

reliability were studied in combination with packet size. The goal is to provide optimum energy consumption while the delay and transmission reliability requirements are considered. The effect of energy for reception/transmission of a pulse and network topology is shown in the model. The optimized model provides a guideline for optimal design of energy harvesting wireless nanonetworks. In the next chapter, the optimum energy consumption schedule for transmission and reception of packets is investigated.

CHAPTER 5

ENERGY CONSUMPTION SCHEDULING OPTIMIZATION

In this chapter, we investigate how to optimize the amount of energy consumed at each time slot. The goal is to maximize the utilization of energy while providing the balance between energy consumption for reception and random behavior of energy arrivals. Finding the optimum energy consumption is difficult because of very limited energy storage as well as limited amount of energy harvested at each step in time.

Moreover, maximizing the utilization of harvested energy is challenging because the intensity of available energy has a stochastic behavior. The utilization is achieved through optimizing communication energy inasmuch as the energy for communication comprises the major portion of energy consumption for a nanonode. Thus, we must design the energy consumption rate (i.e., transmission and reception rate of data) in such a way that the probability that the nanonode does not have energy to communicate in the future is minimized while the data rate is maximized.

In the remainder of this chapter, we first overview literature on energy optimization in Section 5.1. Next, we introduce our problem formally in Section 5.2. Section 5.3 follows with the introduction of some basic schemes for energy consumption. In Section 5.4, we formulate the problem of consumption rate allocation as a Markov decision process (MDP), which can find the optimal scheme to maximize the data rate. We follow this with an analysis of the energy storage capacity, harvesting process, and data rate utilization. This analysis with the basic schemes inspires the development of light-weight heuristic schemes in Section 5.5. The performance of the designed schemes is evaluated in Section 5.6. Finally, the chapter is concluded in Section 5.7.

5.1 RELATED WORK

Optimizing the consumption of harvested energy for communication has been a popular topic of research in recent years [93, 110, 111, 112, 95]. Kansal et al. [93], as one of the initial efforts in this domain, developed a model to evaluate the

process of energy harvesting and consumption. They mainly focus on modeling the energy consumption and harvesting process for sensor nodes without optimizing the consumption. The next step, which has been considered in later work (e.g., [110, 111, 112, 95]), focuses on the optimization of the problem. In some work (e.g., [113], [84]), data transmission with a limited data buffer has been considered. Then, optimal online policies for stabilizing or controlling admissions into the data buffer are proposed. Later on, finding the optimal throughput or minimizing transmission delay has been investigated [114, 111, 115, 116]. Another aspect of optimization occurs when the capacity of energy storage is considered finite or infinite [84, 111]. The general solution approach is to find the trade-off between the loss of energy and one of several quality of service metrics (e.g., packet loss, delay). Since energy arrival is a random process, research has mainly taken two approaches to address this problem.

In the first approach, most of the work, such as [95, 115], does not explicitly include the stochastic behavior of energy harvesting in their modeling. They assume that they can develop a prediction method such as exponentially weighted moving average (EWMA) to predict the amount of available energy in upcoming slots. Noh and Abdelzaher [95] consider the optimal scheme for finding the optimal data flow. They assume the energy arrival is known for a defined period of time, for example, a day. Then, they model it as a linear programming problem, in which they find the optimal rate of transmission during the defined period. These methods could be useful in long-term scenarios. However, these methods do not fully utilize the energy. Ho and Zhang [115] evaluate optimal energy allocation when there are variable channel conditions and energy sources. The throughput as the maximization objective is investigated in two settings: partial information (status of past and present slots) and full information (status of past, present, and future slots). They use dynamic programming to find the optimal rate in a defined duration.

The second approach is to explicitly include the random properties of energy harvesting in modeling and optimization. Yang and Ulukus [116] investigate the optimal packet scheduling problem for a single-user energy harvesting system, where both the data packets and the harvested energy follow stochastic arrival. They develop a scheme to adaptively change the transmission rate based on the traffic load and available energy required to minimize average packet delivery time. They assume that the energy harvesting times and harvested energy amounts are known.

Therefore, they could develop optimal off-line scheduling policies. Tutuncuoglu et al. [111] investigate optimal schemes for wireless transmission when channel fading exists. They evaluate two objective functions: maximizing the throughput and minimizing the transmission completion time. They solve the problem in both deterministic and stochastic settings. The deterministic case is solved in an offline fashion, where the energy arrival and channel fading properties are known. Next, they solve the stochastic problem, which involves random processes of energy arrival and channel fading. The common approach is to evaluate various scenarios with full or casual information about various stochastic processes such as harvesting or channel conditions. Furthermore, they analyze the processes of energy harvesting and consumption to find heuristic methods which can perform near-optimal.

Huang and Neely [117] investigate the problem of finding the best scheduling and flow control at the network level. They include stochastic models of energy harvesting. However, they do not use the stochastic properties directly, and they only need to know the amount of available energy at each instant of decision making. They use the Lyapunov optimization technique for modeling, since linear or dynamic programming does not apply in their problem conditions. However, the complexity of the scheme does not allow it to be run on resource-limited nodes. A common approach for utilizing the harvested energy, which we also use, is to model the optimum energy consumption problem as Markov decision processes to maximize a utility function, e.g., [110, 80, 113]. Seydi and Sikdar [110] develop a model in which they optimize a utility function. The goal is to maximize the possibility of reporting different events when packets may be dropped due to lack of energy or error in the wireless channel. Gorlatova et al. [80] model energy harvesting and consumption to maximize the data rate through a utility function. They consider an extremely large energy storage. Their optimization model computes the optimum rate of transmission. Lei et al. [113] define a Markov chain model to find the optimal transmission policy for sensors. Upon successful transmission, a reward is given to the node. The goal is to maximize the average reward rate where the different energy budgets and energy renewal modes (recharging and replacement) are considered.

In addition to various approaches to optimize the energy consumption, the problem can be solved at different scales, e.g., node, link, or network. Most of the previous work has been developed at the node or link level [93, 110, 111, 112, 114, 80]. A solution at the network level (more than two hops) would seem to be more useful.

However, these solutions [117, 95] mainly have communication overhead for synchronization. Since the goal in nanonetworks is to have independent and distributed solutions as much as possible, it is better to develop a distributed mechanism when it comes to optimizing the energy consumption at the network level. We will investigate this issue in Chapter 6. In this chapter, we will focus on optimizing the energy consumption for a single node.

Previous work for optimizing energy consumption is not applicable to nanoscale networks. First of all, models assume that energy storage capacity is infinite or extremely large. In nanonetworks, it is envisioned that the energy storage capacity will be very limited [5] where nanonodes will have only enough energy storage for the communication of several hundred bits. Therefore, a new model is required to take into account this limitation. Second, in the previous models, the energy for reception is not considered. This assumption is valid when the reception energy is much lower than the transmission energy or in RFID networks, where a node exploits the energy of received packets for transmission. Another example could be single hop communication, where for each reception, a node either does not transmit or sends only one transmission (request-response model). However, in nanonodes, which will most likely operate in a multi-hop fashion with several neighbors, reception can be significant, especially when the energy budget is very limited. Third, in previous work, the harvesting rate is assumed to be very close to the consumption rate. This is not valid for nanonetworks, where for example it can take 10 seconds to have energy for the transmission of only a couple hundred bits [5]. This assumption affects performance, as we will see in this work. Finally, most optimal models are either valid for very limited scenarios, which are not useful if any of the parameters are changed, or they are too compute-intensive to be run on nanonodes. Therefore, new schemes such as heuristic light-weight methods similar to ones which we develop in this work are required.

Due to differences between the nanoscale and microscale paradigms, previous optimization models of harvested energy are not applicable to the nanoscale problem. Most previous work at the microscale does not include the characteristics of the energy harvesting process, energy storage, and processing capability of nanoscale devices in their models. In the domain of nanonetworks, an initial model of energy consumption and harvesting has been proposed by Jornet and Akyildiz [5]. They show that communication would be the main consumer of energy, especially

for nanonodes that communicate through electromagnetic wireless channels [5, 2]. However, they only model the joint process of harvesting and consuming energy, not the optimization of energy consumption.

5.2 SYSTEM MODEL

In this section, we introduce our notation for the combined process of energy harvesting and energy consumption at a nanonode that is part of an ad hoc nanonetwork. Each nanonode transmits its own data as well as receives and forwards its neighbors' data. The particular reception and forwarding schemes are described in Chapter 6.

Energy harvesting follows a random variable, while energy consumption is defined based on a set of available actions on how much energy is to be consumed. Later, several schemes are developed to control the process of energy consumption, i.e., select the action for each state of energy based on various objectives. Various schemes are described in Sections 5.3-5.5.

We consider a discrete time model, in which time is slotted into intervals of unit length. In each slot, some energy is harvested and added to the energy storage, and similarly some energy is consumed and deducted from the energy storage based on the consumption scheme. We assume that the energy storage is ideal and there is no significant leakage. The amount of harvested energy follows a random process.

We denote the system states by $S = S_1, S_2, \cdots, S_s$, where $s = C + 1$ for energy storage of capacity $C \cdot E_{min}$ units of energy. The value of E_{min} denotes the unit of energy, e.g., 1 pJ. The first state (S_1) is called the *out of energy* state, where there is no energy for communication. The last state (S_s) is called the *full energy* state, where there is no capacity to store new energy arrivals. Being in either *out of energy* state or *full energy* state is not desirable because it means the loss of packet receptions (due to lack of energy) or loss of harvested energy (due to lack of storage), respectively.

The energy generation process of the nanonode is modeled by a random process, denoted as an i.i.d. random variable H. We discretize H to take one of the discrete values $[h_0, h_1, \cdots, h_D]$ with probability $p = [p_0, p_1, \cdots, p_D]$. The h_i indicates the amount of energy harvested and p_i is defined as

$$p_i = F_H(h_i) - F_H(h_{i-1}),\ h_{-1} = 0\,, h_i > h_{i-1}\,. \tag{35}$$

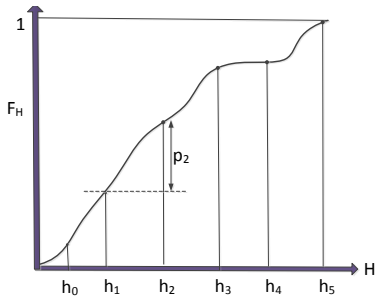


FIG. 31: Discretization of the Energy Harvesting CDF.

Figure 31 shows a sample discretization of a harvesting cumulative distribution function (CDF). The p_2 corresponding to h_2 is shown ($p_2 = F_H(h_2) - F_H(h_1)$). The value of D is determined based on the requirement that p_i is always greater than the threshold δ and $F_H(h_i) \leq \delta_2$, where $\delta \in \mathbb{R}(0, 10^{-pp}]$, $\delta_2 \in \mathbb{R}[1 - 10^{-pp}, 1)$ for pp digits of accuracy. The value of h_0 can take both zero and nonzero values. If $h_0 = 0$, then $p_0 = 0$. This means that always some energy will be harvested. In some scenarios, the amount of harvested energy may be lower than one unit of energy (E_{min}). In this case, $h_0 > 0$ and $p_0 > 0$.

Since the differences between the h_is need not necessarily to be equal, this mapping applies for both linear and nonlinear storage. In the simplest form, for linear storage, harvested amounts in $[h_{i-1}, h_i)$ represent that $i \cdot E_{min}$ units of energy are harvested, as shown in Figure 31. The value of p_2 represents the probability that two units were harvested. This means that the system will move from arbitrary state S_m to state $max(S_{2+m}, S_s)$ with the assumption of no energy consumption during the same slot.

The unequal differences between value of the h_is can also represent the units of energy for nonlinear storage. This can be done by applying a nonlinear function to the random variable H, which still produces a random variable [105]. Nonlinear storage is often found in capacitor storage, e.g., [5, 80].

It is assumed that there are always packets ready for transmission. The transmission and reception of each packet will consume E_{tx} and E_{rx} units of energy, respectively. We assume the energy consumed for listening and idle modes is negligible, based on previous studies [5, 11]. The consumption strategy of a nanonode, i.e., the number of transmissions and receptions per slot, is denoted as scheme π. The action taken by a node in a time slot is denoted as $a_{(i,j)}$, $i, j \geq 0$, which is selected

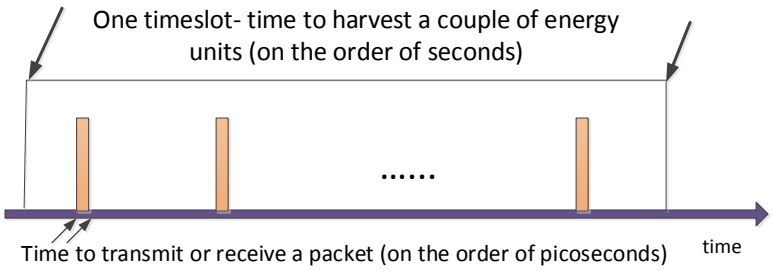


FIG. 32: Comparison of Timescales Between Harvesting and Consumption of Energy.

from $A = \{a_{(0,0)}, a_{(1,0)}, a_{(0,1)}, a_{(1,1)}, \cdots, a_{(m,n)}, a_{(m+1,n)}, a_{(m,n+1)}, a_{(m+1,n+1)}, \cdots\}$. The action $a_{(i,j)}$ corresponds to the reception of i and transmission of j packets in the time slot, where the sum of the energy consumption for them, denoted as E_k, cannot exceed the maximum consumption per slot, E_c, $0 \leq E_c \leq C$, i.e., $i \cdot E_{rx} + j \cdot E_{tx} \leq E_c \quad \forall\, i, j$. We denote S_A as the number of members of A. The minimum S_A is 3, which corresponds to $A = \{a_{(0,0)}, a_{(1,0)}, a_{(0,1)}\}$, and consequently $E_c = \max(E_{tx}, E_{rx})$. For the simplicity of presentation, we consider the total energy consumption for both transmissions and receptions as E_k with the corresponding $a_k\,, 1 \leq k \leq S_A$. Without loss of generality, we assume that the actions of A are ordered ascending based on their E_k values. We assume that there is a mechanism in which a nanonode can enforce the number of receptions. This can be an independent mechanism by each node or can be a synchronized mechanism between transmitters and receivers. The simplest mechanism is just to disable the communication module for some period of time, during which the nanonode does not want to receive packets. Details of how to decide the times for disabling the communication module are described in Chapter 6.

Although the model is general, the focus of this work is for scenarios where the consumption rate is faster than harvesting rate, as illustrated in Figure 32. Therefore, it means that several units of energy are consumed per packet transmission or reception. Likewise, several packets can be exchanged in one time slot. Note that this action set definition can cover multiple packet communication situations. For example, $a_{(0,2)}$ can represent the transmission of one packet with twice the amount of energy, as well as the transmission of two packets. Nevertheless, for simplicity we assume that $a_{(i,j)}$ maps to i receptions and j transmissions. Before describing our optimal scheme, we first introduce some basic schemes.

5.3 BASIC SCHEMES

In this section, we describe some basic consumption schemes that are intuitive and common in the literature ([110, 118, 119, 120]). They will be used later to compare with our optimal and heuristic schemes. Also, they will help in designing the heuristic schemes.

- **Aggressive (Agg)**: In this scheme, the highest possible consumption action, based on the amount of available energy, is always selected. This method tries to achieve the highest data rate. However, it will result in the *out of energy* situations most of the time.

- **Conservative (Con)**: In this scheme, one of the lowest consumption rates, i.e., $a_{(1,1)}$, $a_{(0,1)}$, $a_{(0,1)}$, or $a_{(0,0)}$, is selected based on the availability of energy. With this scheme, there is always some energy left, but the data rate as well as the utilization of energy is very low.

- **Consume-Harvest (C-H)**: In this scheme, consumption is selected based on the amount of energy which has just been harvested in the previous time slot. More specifically, it will choose the action with the amount of energy closest to the amount of just harvested energy. This scheme is expected to behave better than the conservative scheme in terms of data rate. However, there is the chance of falling to the *full energy* state because in many time slots the amount of harvesting may not be enough to transmit or receive any packet, but it will result in the accumulation of energy units.

- **Mean**: In this scheme, the average action, which is $a_k, k = \lfloor \frac{S_A}{2} \rfloor$, is selected. If there is not enough energy to select the average action, then the closest action is chosen. The performance of this scheme would be between the conservative and aggressive schemes.

- **Random (Rand)**: This scheme selects an action randomly from the set of actions. If the energy for the chosen action is above the current energy level of storage, the random selection process is repeated. The behavior of this scheme cannot be predicted exactly. In general, it is expected that it will have an average performance in the long-term.

TABLE 13: Performance of Basic Schemes - (H)igh, (M)ean, (L)ow.

Evaluation Metric	**Agg**	**Con**	**C-H**	**Mean**	**Rand**
Chance of Being in Out of Energy State	H	L	L	M	M
Chance of Being in Full Energy State	L	H	H	M	M
Energy Utilization	H	L	M	M	M

The evaluation of these basic schemes reveals that they cannot achieve the maximum utilization of energy while avoiding going to the *full* or *out of energy* states. Table 13 compares these basic schemes in a general view without going into details of evaluation and results, which will be described later in Section 5.6. Table 13 reveals that none of the schemes perform well in all metrics. In fact, they cannot satisfy and balance these metrics at the same time. Therefore, there is a need to develop an optimal model. In fact, a model with a low chance of being in the *full* or *out of energy* state while having high utilization of energy is required. We will develop such a model in the next section.

5.4 OPTIMAL MODEL

The problem of assigning the optimal action (i.e., number of transmissions and receptions) per slot can be described as a Markov decision process as follows.

The system model is as defined in Section 5.2. The probabilities of transferring between states depend on the current state, the amount of energy harvested, and the action taken. Actions are selected from the set A. Formally, state transitions can be written as

$$P(S_i, S_j, a_k) = p_u \,, \quad \sum_{x=1}^{s} P(S_i, S_x, a_k) = 1 \,, \; 1 \leq i, j \leq s \,,$$

and j is specified as $j = i + h_u - E_k$, $E_k < i$, $i + h_u \leq s$, $0 \leq u \leq D$, $1 \leq k \leq S_A$. The value of j represents the energy state after the harvesting of h_u units and consumption of E_k units of energy for action a_k taken. The condition $E_k < i$ limits the actions which can be taken to avoid consuming more energy than is harvested and stored. The condition $i + h_u \leq s$ limits the harvested energy to the available capacity of energy storage. When $j = 1$, the system falls into the *out of energy* state, i.e., the node has consumed all of energy that it has stored and harvested. When $j = s$, the system falls into the *full energy* state, i.e., even after consumption, the system has

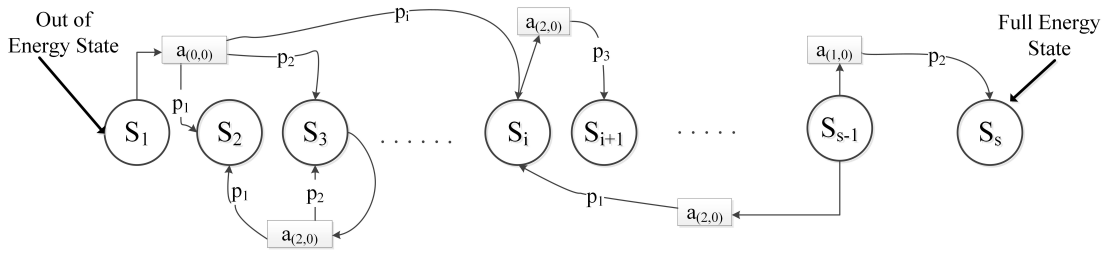


FIG. 33: A partial MDP (states and some actions are represented) - E_{min} is set to E_{rx}.

stored and harvested up to the capacity C.

Figure 33 illustrates some states, actions, and transitions between states. We note that this is not a full MDP diagram, but just serves as an illustrative example. Assume the node begins in state S_1 (*out of energy*). It will take action $a_{(0,0)}$ because there is no energy for either transmission or reception. Thus, the state it transits to is only dependent upon the amount of energy harvested. Assuming linear storage and that each state represents the energy for one additional reception, with probability p_1 the system will harvest one unit of energy and move to state S_2. With probability p_2 it will harvest two units of energy and move to S_3. If the system is in state S_3 and takes action $a_{(2,0)}$ (two receptions which consume two units of energy), with probability p_1 it will harvest one unit of energy and move to state S_2, and with probability p_2 it will harvest two units of energy and remain in state S_3.

The reward function is defined in a way to maximize the utilization of energy, i.e., higher data rate, while satisfying the *packet balance* between reception and transmission. Maximizing the utilization of energy is a well-known metric [80, 114]. It also can be directly used as a metric for delay performance [114]. In addition to maximizing the utilization of energy, our model includes the packet balance between the number of packet receptions to the number of packet transmissions. As mentioned before, the energy for reception can be significant when a limited amount of energy exists, as well as when the number of transmitting neighbors becomes large. Therefore, we define our reward function to include the packet balance as well as energy utilization. Each of these two objectives is defined as follows.

5.4.1 MAXIMIZING THE ENERGY UTILIZATION

Maximizing the utilization of energy is directly related to moving between states. If the action is taking the system to a state with a higher level of energy, this poses negative rewards, i.e., energy has not been utilized. Recall that we assume that there are always packets ready to transmit. Therefore, not utilizing available energy means that packets that could be sent are not transmitted due to not taking the appropriate action for the current state. On the other hand, if the action is taking the system to a state with a lower level of energy, a positive reward will be given. Also, the reward for going to the first and last states should be significantly lower than other rewards. This is defined to avoid letting the energy level become zero or full. Being in the *out of energy* state causes the loss of packet receptions due to lack of energy. Likewise, being in the *full energy* state results in the loss of energy reception due to lack of empty space in storage.

The transition function between states, which is directly related to the utilization of available energy, is defined as the following function

$$J(S_i, S_j, a_k) = \begin{cases} i - j & j \neq 1, s \\ -(s+1) & j = 1, s \end{cases}.$$

The function J simply defines positive values on more consumption of energy (utilizing energy) and negative values on the more harvesting of energy (not utilizing energy). Moreover, any transition to the first or last states receives a negative value. Although this function does not measure the utilization of energy directly, it satisfies as a function for our MDP model. Energy utilization EU for any point T along time can be formally defined as

$$EU = \frac{\sum_{t=0}^{T} E_t}{\sum_{t=0}^{T} H_t},$$

where E_t and H_t are the amounts consumed and harvested at timeslot t, respectively. EU has values in $[0, 1]$.

5.4.2 BALANCING PACKETS

Balancing between the number of receptions and transmissions is required for several reasons. First, if a packet balance is not defined, the MDP solution may lead to only transmissions or only receptions, which is not desirable. Second, always

having a fair distribution of energy between transmission and reception in an ad hoc network is required to balance between the transmission of a node's data versus forwarding neighbors' data. The target packet balance B_D is defined as the ratio between the number of packet receptions to the number of packet transmissions. The target *packet balance* is an application-dependent parameter, which may vary based on the number of neighbors, routing scheme, etc.

An *action balance* for action a_k is defined as $B_{a_k},\ 1 \leq k \leq S_A$,

$$B_{a_k} = \begin{cases} \frac{a_{k_r}}{a_{k_t}} & a_{k_t} \neq 0 \text{ and } a_{k_r} \neq 0 \\ \epsilon & a_{k_t} = 0 \text{ and } a_{k_r} = 0 \\ \frac{1}{E_k} & a_{k_t} = 0 \text{ or } a_{k_r} = 0 \end{cases},$$

where a_{k_r} and a_{k_t} represent the number of receptions and the number of transmissions for action a_k, respectively. The maximum value of B_{a_k} is denoted as B_{max}. Recall E_k is the amount of energy consumption for action a_k. For an action with no consumption, ϵ is selected as a very small value, e.g., $0 < \epsilon \leq 0.1$, which shows that this action does not affect the packet balance.

Then, the similarity function L of the action's balance B_{a_k} to the target balance B_D is defined as

$$L(B_D, B_{a_k}) = \begin{cases} |B_D - B_{a_k}|^{-1} & B_D \neq B_{a_k} \\ \frac{B_{max}}{\epsilon} & B_D = B_{a_k} \end{cases}.$$

When the action has the maximum similarity with the target balance B_D, it will take a large value. Otherwise, the similarity is related to the proportional ratio of action receptions and transmissions to the target balance.

5.4.3 REWARD FUNCTION

Finally, the reward function is formally defined as

$$R(S_i, S_j, a_k) = J(S_i, S_j, a_k) \cdot L(B_D, B_{a_k})$$

This reward function implies that the total average reward is given to the highest average data rate, which is achieved via maximizing the utilization of available energy for harvesting. Moreover, it favors the actions which try to achieve the target packet balance.

5.4.4 SOLUTION FOR NON-STOCHASTIC SCENARIO

In this section, we look at a scenario, where we have *a priori* knowledge of the amount of harvested energy. Assume that all the harvesting values in the timeslot between $[0, T]$ are known at time 0. We also relax the conditions of avoiding going to the *out of energy* or *full energy* states. Then, the problem of maximizing energy utilization can be written as follows.

$$U_k(S_i) = \max \sum_{t=0}^{k} E(t)\,, \tag{36}$$

where $E(t)$ represents the amount of energy consumption in each slot.

The behavior of this function has been presented as the curve under the stair case of harvested energy in [116]. It is shown that the closer $E(t)$ is to the amount of energy harvested, the better policy will be selected. Finding the solution for continuous time and continuous power consumption function has been studied in previous work, e.g., [115, 114, 111, 116].

In our scenario, we are dealing with discrete time units. Moreover, because the actions are selected from a set of actions, power values do not take continuous values. In addition, in an extended scenario, we include a balance factor, which limits the actions that can be selected. The balance function can be defined as

$$Bal_k = \min B_D - \sum_{t=0}^{k} \frac{N_{rx}(t)}{N_{tx}(t)}\,, \tag{37}$$

where N_{rx} and N_{tx} correspond to the number of receptions and the number of transmissions for E_t energy consumption at one slot, respectively.

Since the problem is discrete, tracing the solution is not trivial and will not result into useful insights for the stochastic scenario. Therefore, we limit our discussion and move to a solution for the stochastic scenario.

5.4.5 SOLUTION FOR MDP

We solve this MDP through the value iteration method [121]. Let $V(S_i)$ be the value for each state, $1 \leq i \leq s$. At the end of the solution by the optimal policy Π, $V(S_i)$ will represent the discounted sum of the rewards to be earned (on average) by using that solution for state i. The iterative steps are calculated based on

$$\pi(S_i) := \arg\max_{a_k} \left\{ \sum_{S_j} P(S_i, S_j, a_k) \left(R(S_i, S_j, a_k) + \gamma V(S_j) \right) \right\},$$

$$V(S_i) := \sum_{S_j} P(S_i, S_j, \pi(S_i)) \left(R(S_i, S_j, \pi(S_i)) + \gamma V(S_j)\right) ,$$

where γ is the discount factor. After substituting the calculation of $\pi(S_i)$ into the calculation of $V(S_i)$, the combined step would be

$$V(S_i) := \max_{a_k} \left\{ \sum_{S_j} P(S_i, S_j, a_k) \left(R(S_i, S_j, a_k) + \gamma V(S_j)\right) \right\} .$$

This is repeated until the results converge.

5.4.6 ANALYSIS OF MARKOV DECISION PROCESS

The behavior of the MDP solution can be roughly described as follows. For a specific harvesting rate, the states of the system could be categorized into three main categories: (I) states close to the *out of energy* state; (II) states close to the *full energy* state; and (III) states in between. In the first category of states, the actions will try to stabilize the system to avoid going to the *out of energy* state. Similarly, in the second category of states, the actions will try to avoid going to the *full energy* state. They can safely go to one of the states in third category. The optimal actions for the third category of states would be to stay in their own category or at most move to one of the first category states. It is better to move towards the *out of energy* state than to be conservative and go to the *full energy* state because this way the energy utilization will be maximized.

The energy storage capacity of the system is the main parameter that determines the number of states. Therefore, we evaluate it here in more detail. This analysis will also provide a better understanding of the MDP in order to develop our heuristic methods in Section 5.5.

First, we evaluate the lower bound of the amount of energy storage, C, required to avoid going to the *full energy* state. In this analysis, we consider linear storage. Recall that H was the random variable for the harvesting with its distribution F_H. For simplicity, it is being discretized into D parts (Figure 31) based on the unit of energy for linear storage. Then the MDP is defined, where the number of states were defined as $s = C + 1$. Assume C is less than D. Thus, s is equal to D at most. Now, consider the following scenarios:

I) The system is in the first state, i.e., *out of energy* state. Clearly, the optimal action for first state is no consumption, $a_{(0,0)}$. Then assume energy arrives with

probability p_D, i.e., D units of energy have been harvested. It means that there is a jump from state 1 to state $D+1$. However, there is no such state because we assume that there are only D states. This implies that C should be larger or equal to D to avoid going to the *full energy* state.

II) The system is in any other state except the first state. Let us assume that the system is in arbitrary state i and the associated optimal action is a_k. The maximum amount of energy that action a_k could consume is E_c. This means that the minimum jump with an arrival of D units of energy from state i towards the *full energy* state would be $D - E_c$, and that the minimum capacity should be $i + D - E_c$. There are two cases here:

- $E_c < D$ which does not provide a bound.
- $E_c \geq D$ which is more likely to happen. This way, there would not be any chance of moving to the *full energy* state.

As a result, the minimum value of D would be a lower bound for capacity. We call this the minimum lower bound for energy storage capacity to avoid going to the *full energy* state.

A similar reasoning applies for the *out of energy* state. In fact, for an arbitrary state i, the maximum jump toward the *out of energy* state occurs when energy arrives with p_0, i.e., no energy arrival, and the consumption is the maximum amount, i.e., E_c. Therefore, the next state would be $i - E_c + 0$, which should be greater than or equal to 2 to avoid going to the *out of energy* state. Then we can write $i \geq E_c + 2$. The minimum jump occurs when $E_c = 1$ since $E_c \geq 0$. This means the minimum number of states is 3. Comparing this with $s = D + 1$, the minimum energy storage capacity to avoid going to *out of energy* state would be $\min(3, D+1)$. Typically $D + 1$ is larger than 3; therefore, the minimum storage to avoid going to either the *full* or *out of energy* states is $C = D$.

5.5 HEURISTIC SCHEMES

Running an MDP solver, especially when the number of states grows, is too compute-intensive for nanonodes. Although the solution for MDP is a stationary solution, which means it can be solved once and used afterwards, in many situations it is better to use lower complexity schemes, e.g., a light-weight heuristic scheme. First,

many parameters such as the capacity of nodes and harvesting models can be different even among neighbor nodes. For example, one node may receive more vibration from human movement when it is mounted on a leg than the chest. So, having a stationary solution may not be the best approach. Second, it is a compute-intensive task for limited resource nanonodes to compute the optimal scheme based on their specific parameters such as capacity of energy storage, action set, etc. Therefore, it is useful to develop heuristic methods with performance close to the optimal solution. In the following, we describe our heuristic schemes.

5.5.1 SLOW BEGINNING FAST ENDING (SBFE)

The slow beginning fast ending (SBFE) method was inspired by the basic schemes. This heuristic method acts conservatively with a low energy level and aggressively with a high energy level. As shown in Algorithm 1, at the first step, the lowest consumption action from set A, i.e., $a_{(0,0)}$, is assigned to the first state, and the highest consumption action, i.e., a_{S_A}, is assigned to the last state. Next, if there are more states remaining than the number of actions, we assign actions to states in an ascending order and then assign the highest consumption for the remaining states. Otherwise, we use the highest consumption rate for all states. This heuristic scheme enables adapting a slow increase in consumption (conservative view) to avoid falling to the *out of energy* state while it uses the highest consumption (aggressive view) to utilize the energy as much as possible when it is available.

Algorithm 1: SBFE Heuristic Method

```
SBFE()
    Input  : s (Number of States), A (Set of Actions)
    Output: Action for each state

    Assign Action a_(0,0) for the first state ;
    Assign Action a_S_A for the last state ;
    if s-2 ≥ S_A
        Assign the rest of Actions a_2 to a_S_A-1 to states 2 through S_A − 1 ;
        /* a_k is the ascending list of actions based on their
        consumption value                                          */
        Assign a_S_A to states S_A through s − 1 ;
    else
        Assign a_S_A to state 2 through s − 1 ;
```

5.5.2 ADAPTIVE

The *adaptive* method tries to select the actions proportional to the state of energy. The higher the level of current energy, an action with higher consumption is selected. If the level of available energy is below the requested energy action, the next lower consumption action is selected. Algorithm 2 represents this adaptive heuristic method. Indeed, the *adaptive* scheme tries to stabilize the state in one of its close states and also not to move to the first or last states. This approach corresponds with the optimal policy solution as described in Section 5.4.6.

For scenarios with *packet balance* B_D, the list of actions that do not provide the requested B_D are filtered out. Note that SBFE does not support the *packet balance* factor because, as it will be shown in simulation results, even the simple form does not perform well.

Algorithm 2: Adaptive Heuristic Method

```
Adaptive()
    Input  : s (Number of States), A (Set of Actions)
    Output: Action for each state

    if B_D
        A = Only actions from A, which meet the B_D
    for i = 1; i ≤ s; i++
        index = ⌈(i/s) · S_A⌉ ;
        while a_index > E_i
            /* E_i:  energy at state i                    */
            a_index = a_index -1;
        Assign a_index to state i;
```

5.6 SIMULATION

In this section, we evaluate the introduced schemes in terms of several metrics. The goal is to show how each scheme performs in maximizing the utilization of harvested energy. The values of the parameters are listed in Table 14. In the first scenario, the performance of various schemes in the utilization of energy is evaluated. In the second scenario, the effect of energy storage on the performance of schemes is presented. The third scenario focuses on how efficiently each scheme can satisfy the requested packet balance. Finally, the fourth scenario illustrates the performance of each scheme when nonlinear linear storage is considered. The harvesting rate

TABLE 14: Simulation Parameters.

Scenario	**Harvest Rate** (pJ/s)	C (pJ)	E_c **(% of** C**)**	B_D
1-Energy Usage	variable	20	50	1
2-Energy Storage	0.5	4-20	50	1
3-Effect of Balance	0.5	6-12	100	3
4-Nonlinear Storage	variable	20	50	1

follows an exponential distribution, except for the last scenario where the lognormal distribution is also evaluated. Nanonodes communicate based on the Rate Division Time-Spread On-Off Keying pulse method [12], where pulses correspond to logical `1`s and silence correspond to logical `0`s. In all scenarios, E_{rx} is equal to 1 pJ and E_{tx} is set to 2 pJ. Similar results were found for setting E_{tx} to 3, 5 and 10 pJ, with corresponding increases in C. The value of C is determined based on the analysis in Section 5.4.6. The results of simulations are for the long-term behavior of the system where no change in performance metrics 1-digit after the decimal exists.

5.6.1 ENERGY USAGE

We first show how energy is used based on the various schemes. In other words, we want to make sure that we do not consume too aggressively or too conservatively, which will lead to the *out of energy* state or *full energy* state, respectively.

Figure 34 illustrates the probability of finding a node in the *out of energy* state for each scheme. Clearly, as the harvesting rate is increased with the same consumption rate and energy storage capacity, there is always some energy available. Therefore, the probability of being in the *out of energy* state goes to zero for all schemes. Not surprisingly, the *optimal* scheme never lets the system be in the *out of energy* state, while the *aggressive* scheme has the highest probability to be in that state. The *optimal*, *adaptive* and *conservative* schemes all have similar performance, almost zero always. The close performance of the *adaptive* scheme to the optimal scheme indicates that our light-weight heuristic scheme, *adaptive*, has a near-optimal performance for the probability of being *out of energy* metric. The conservative behavior of the *conservative* scheme results in a situation where there is always some energy left, so the chance of being *out of energy* is zero. *SBFE* performs better than most of other schemes, except *adaptive* and *optimal*. The *random* and *mean* schemes perform near average in comparison to the other schemes.

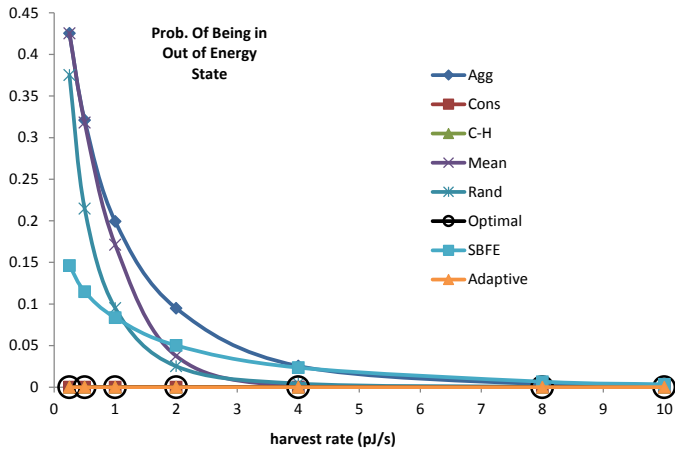


FIG. 34: Probability of Being in Out of Energy State.

Figure 35 illustrates the probability of being in the *full energy* state for various schemes with an increase in harvesting rate. As viewed, the behavior is the reverse of the probability of being *out of energy*. As shown in Figure 35, the probability of being in the *full energy* state is increased with an increase in harvesting rate. The *optimal*, *adaptive* and *SBFE* schemes perform better than the basic schemes. Also, note that for the *conservative* and *C-H* schemes, the probability of being in the *full energy* state is almost one, even with a low harvest rate. This occurs because in the long-term, the consumption of energy is low and storage becomes full. After this, since the consumption is very low, the system still stays in the *full energy* state. One may note that the *optimal* and heuristic (*SBFE* and *adaptive*) schemes perform almost similar to the *aggressive* scheme when the harvest rate is increased. First, this happens to avoid going to the *out of energy* state. Second, this phenomena will occur, independent of scheme, due to the high energy harvest rate. In fact, the harvest rate is faster than the consumption rate in this situation, while the energy storage capacity is the same. In practice, the energy storage capacity should be designed in relation to the harvest rate and the maximum usage of energy, as was discussed in Section 5.4.6. In other words, if it is known that the harvest rate would be much higher than consumption, then the storage capacity should be increased to avoid going to the *full energy* state. Again, the *random* and *mean* schemes perform close to average in comparison to the other schemes.

Figure 36 represents the performance of the schemes in terms of utilizing the harvested energy. Similar to the convergence of all schemes with the increase of

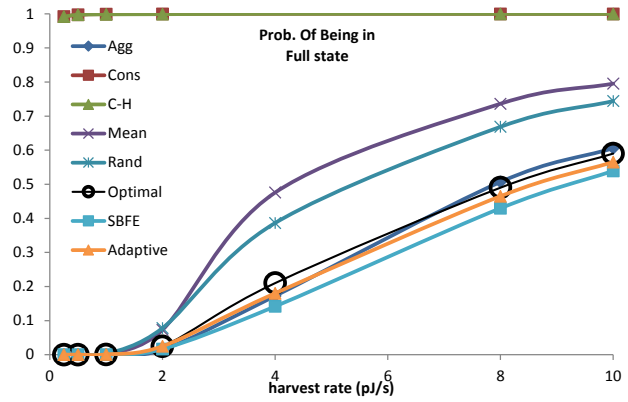


FIG. 35: Probability of Being in Full Energy State.

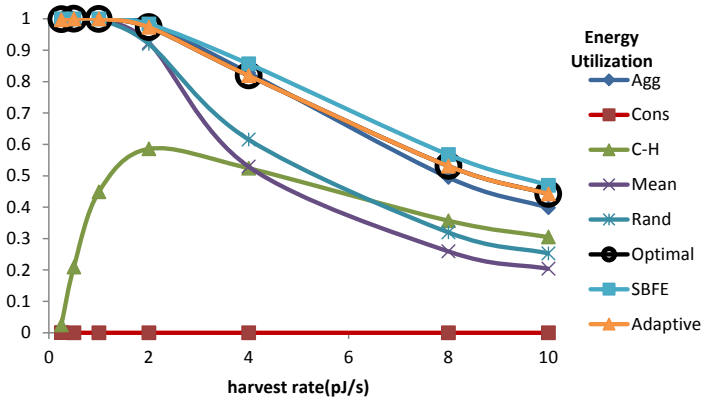


FIG. 36: Energy Utilization for Various Schemes.

harvest rate for the two previous metrics, here also the *energy utilization* for all schemes comes close to each other. Again as shown in Figure 36, the very close performance of *optimal*, *adaptive* and *SBFE* surpasses the basic schemes.

A scheme such as *aggressive* has a very similar energy utilization to the *optimal* and *heuristic* schemes when the harvest rate is increased. However, on the other hand, as shown in Figure 34, this will lead to the *out of energy* state with a higher probability. Note that for smaller harvest rate values, the *optimal* and *heuristic* schemes have almost 100% utilization. Also, as the energy harvest rate is increased, *EU* values converge. Since the energy storage will be full with a high energy harvesting rate and the energy consumption limit is set to the half of energy storage capacity, *EU* values merge towards 0.5.

The different behavior of *C-H* (Figure 36) in comparison to the other schemes is due to the fact that utilization is increased with a higher harvest rate since there is a higher chance of energy arrival. However, after a point, harvest rate = 2 pJ/s, even with the higher arrival of energy, the amount of consumption is limited. Therefore, the utilization falls.

To represent the effect of these three metrics together, the *energy efficiency* (EE) metric is defined as follows

$$EE = \log(\frac{EU}{p(o) \cdot p(f)}), \quad 0 < p(o), p(f) \leq 1, \tag{38}$$

where $p(o)$ represents the probability of being in the *out of energy* state and $p(f)$ represents the probability of being in the *full energy* state. This shows the efficiency of the schemes for these probabilities and energy utilization. The higher the utilization and the lower the probability of being in the *full* or *out of energy* states, the better. A scheme would not be energy efficient if the utilization EU is low and/or the values of $p(o)$ or $p(f)$ are close to 1. The values of $p(o)$ and $p(f)$ are initially set to a finite small value to avoid division by zero.

Figure 37 shows the energy efficiency for the various schemes. Now it is clear that *optimal* has the highest efficiency for lower harvest rates and outperforms other schemes. The *adaptive* scheme performs similarly to the *optimal* scheme. Of course, as the harvest rate increases, there is always energy, which means that $p(o) \to 0$ and $p(f) \to 1$. Similarly, the *energy utilization* goes down because there is not enough storage to store the energy and utilize it. Therefore, the *energy efficiency* metric goes down, independent of the scheme. The *energy utilization* of the *conservative* scheme is low, therefore an increase in the harvest rate, and correspondingly decrease in EU, will result in the taking the log of a small value in (38), which is a negative number. Since the *optimal* and *adaptive* schemes outperform other schemes, for the sake of simplicity, the remaining results are shown only for them.

5.6.2 EFFECT OF ENERGY STORAGE

As shown in Section 5.4.6, energy storage capacity is one of the main design parameters in relation to the harvesting rate. Figure 38 illustrates the probability of being in the *full energy* state with the increase of storage capacity. Clearly, as the storage capacity is increased, the chance of being in the *full energy* state is reduced

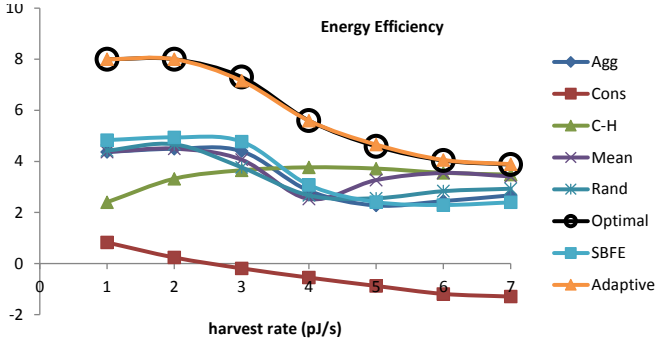


FIG. 37: Energy Efficiency for Various Schemes.

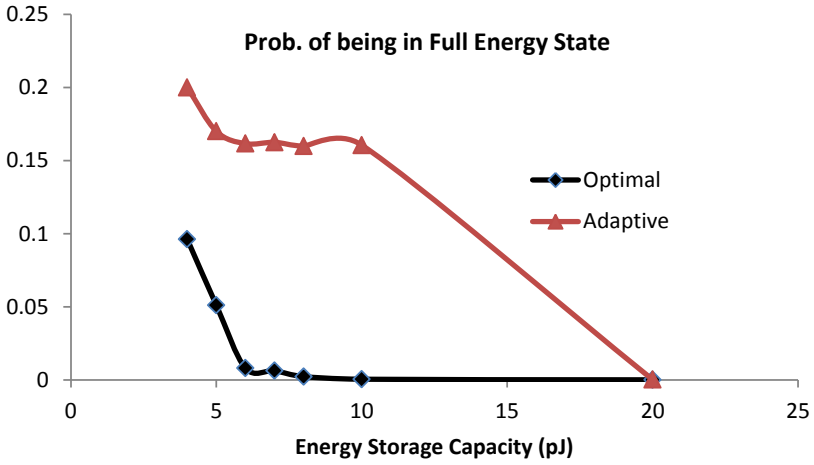


FIG. 38: Probability of Being in Full Energy State with Change of Capacity.

due to there being more capacity available. The probability of being in the *full energy* for the *optimal* scheme goes close to zero at storage capacity 6 pJ. This confirms the analysis in Section 5.4.6, with setting $C = D$ and $\delta = 10^{-6}$ in (35).

Figure 39 shows the energy efficiency of the schemes with the change of storage capacity. As can be seen, the *energy efficiency* between the *optimal* and *adaptive* schemes becomes closer as the energy capacity is increased. This behavior shows that as the energy storage becomes larger, providing the energy utilization is simpler. However, limited energy storage plays an important role for limited energy storage for nanonodes.

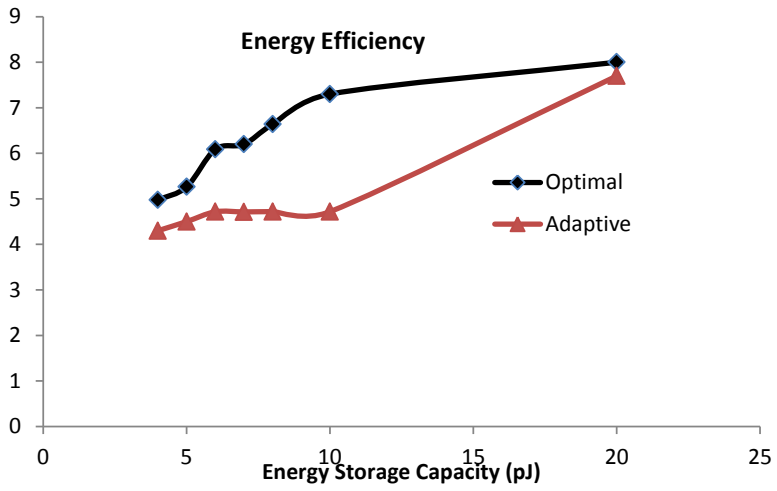


FIG. 39: Energy Efficiency with Change of Capacity.

5.6.3 EFFECT OF BALANCE

Here, we evaluate the performance of the schemes in satisfying the *packet balance* factor. We set B_D to 3, which means the number of receptions has to be 3 times the number of transmissions. In general, as shown in Figure 40, the *adaptive* scheme performs close to the *optimal* scheme for balancing receptions and transmissions. The performance of *optimal* in meeting the balance factor degrades only when the storage capacity becomes smaller. In this case, as illustrated in Figure 40, the *optimal* scheme would have a higher energy efficiency. Similar results were found for packet balances 5 and 7.

5.6.4 NONLINEAR STORAGE

In this experiment, we evaluate the effect of nonlinear energy storage on *energy efficiency*. We assume that nonlinear storage will follow a polynomial of degree d in form of $y = x^d$. Figure 42 represents the effect of nonlinear storage on energy efficiency when the storage has nonlinear structure with degrees $\frac{1}{4}$, $\frac{1}{2}$, 2, and 4. In general, for the *optimal* scheme the lower the degree, the higher the energy efficiency. This shows that the *optimal* scheme takes into account the storage effect, especially when the energy harvesting has a lower rate, here the lower degree. The *adaptive* scheme has the same performance independent of the storage model since storage

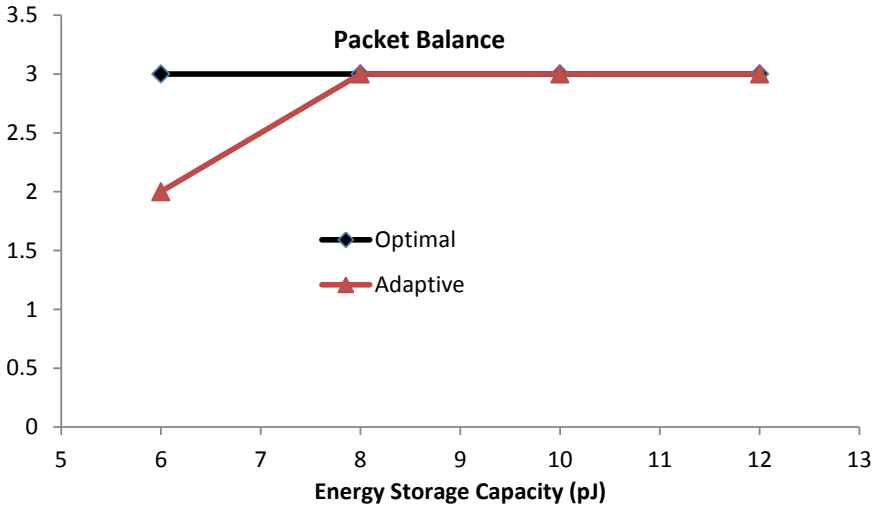


FIG. 40: Packet Balance with Change of Storage Capacity. The target packet balance B_D was 3.

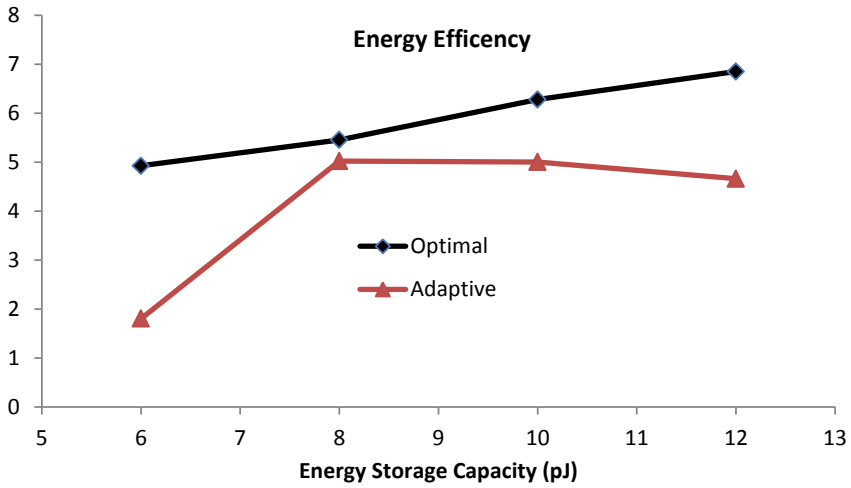


FIG. 41: Energy Efficiency with Packet Balance of 3.

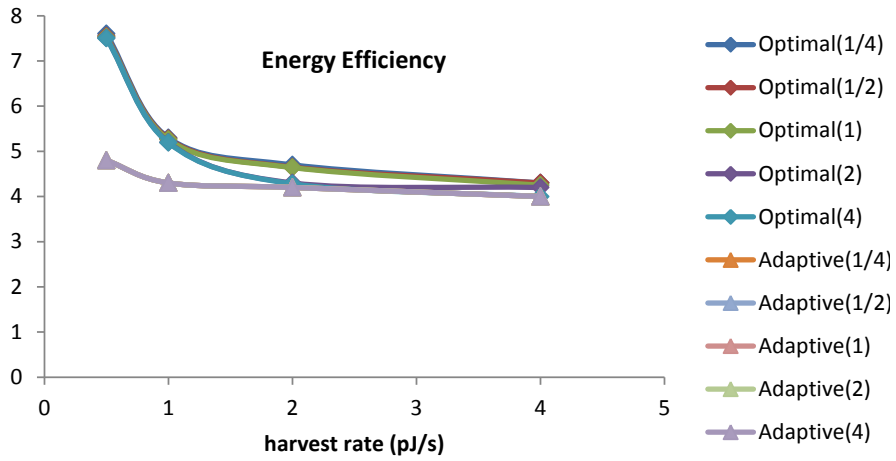


FIG. 42: Energy Efficiency for Linear and Nonlinear Storage - Exponential Harvesting.

model is not included in the scheme.

Figure 43 shows the same behavior for harvesting with a lognormal distribution. The *adaptive* scheme merges with the *optimal* scheme for the lognormal distribution faster than with the exponential distribution. The comparison of these two harvesting distribution models illustrates that the effect of the harvesting distribution is less than the effect of the energy storage model.

5.7 SUMMARY

In this chapter, we introduced the problem of optimum energy consumption for nanoscale nodes that harvest energy from stochastic resources. Nanoscale properties affect the harvesting and storing of energy. Particularly, the low rate of energy harvesting and limited energy storage capacity makes the problem of energy consumption optimization difficult. We analyzed the problem of finding the optimum consumption of harvested energy for nanonodes and proposed an optimal solution that not only maximizes the utilization of energy but also satisfies the ratio of packet reception to transmissions.

We designed a light-weight heuristic approach, the *adaptive* scheme with near optimal performance, that attempts to match consumption with the current energy state. has near optimal performance. This heuristic scheme also tries to satisfy the target packet balance between transmissions and receptions, while striving to avoid

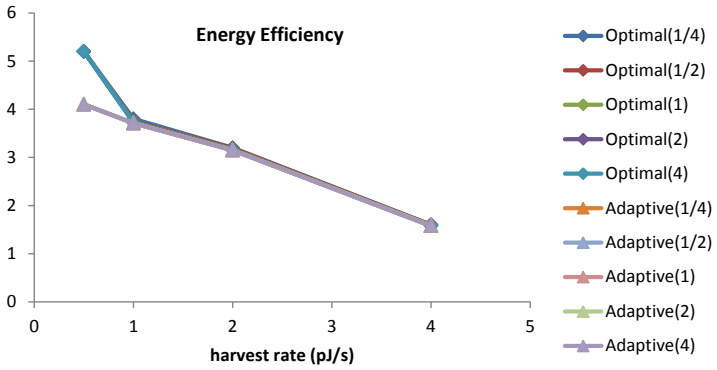


FIG. 43: Energy Efficiency for Linear and Nonlinear Storage - Lognormal Harvesting $(\mu, \sigma^2 = 0.5 \cdot \mu)$.

the *out of energy* and *full energy* states. The *adaptive* scheme with its simplicity shows an acceptable level of performance, especially in avoiding going to *full* or *out of energy* states.

CHAPTER 6

COMMUNICATION BETWEEN NANONODES

Having solutions for optimal packet design and an optimum energy consumption model, a medium access control (MAC) protocol is required to enable communication between nanonodes. The energy harvesting property of the nanonodes is the main parameter that should be included in the design of the protocol. In the MAC layer, coordination between nanonodes is required to make sure that a nanonode receiver will have enough energy to receive packets from a nanonode transmitter at the moment of communication. The design of energy harvesting-aware solutions differs from traditional energy-aware protocols. Energy-aware protocols aim to minimize the consumption of energy while the energy harvesting-aware protocols aim to maximize the utilization of available energy. In nanonetworks, energy is renewed, but the amount of available energy at each moment is limited. Thus, tailored energy harvesting-aware protocols for nano-networks are required.

Designing protocols for accessing the medium is difficult not only because of energy availability, but also because of special properties of nanonetworks. First, in most applications of nanonetworks, coordination among hundreds of nanonodes is required. The tiny nanonodes are also limited in their processing capabilities, implying that complex protocols cannot be considered. Moreover, traditional MAC mechanisms such as message exchange or handshake for synchronization prior to data transfer should be minimized to reduce the consumption of energy as well as to enable the scalability of any solution. Due to these challenges, novel MAC protocols for nanonodes are required [13, 122].

This chapter investigates the issue of MAC protocol design for nanonetworks and develops a scalable, lightweight, distributed, and energy harvesting-aware solution, called RIH-MAC (Receiver-Initiated Harvesting-aware Medium Access Control). Unlike traditional MAC protocols, which mainly focus on minimizing collisions and bandwidth efficiency, our solution relies on a receiver-initiated communication model which addresses the matter of energy harvesting directly. By coordinating the communication through the receiver in RIH-MAC, a transmitter adaptively selects

its participation in the network load, thus allowing RIH-MAC to achieve a low collision probability, a high packet delivery ratio, and high energy utilization. In fact, a transmission occurs only if there is a high probability that the receiver will have enough energy for the reception. RIH-MAC can operate in both centralized and distributed topologies of nanonetworks. The centralized solution deals with topologies in which nanonodes are in direct communication with a more powerful device, called a nanocontroller [9, 122], which will be responsible for scheduling the communication with nanonodes. In the distributed RIH-MAC (DRIH-MAC), we develop a solution for an ad hoc formation of nanonodes. Each nanonode can directly communicate with other nanonodes in the neighborhood, and these neighbors provide connections to other nanonodes in the network. DRIH-MAC is more challenging since there is no central point for scheduling communication. In both solutions, we include the properties of energy harvesting.

This is the first attempt to apply the idea of receiver-initiated transmission to energy harvesting nanonetworks. By coordinating the communication through the receiver in RIH-MAC, a transmitter adaptively selects its participation in the network load, allowing RIH-MAC to achieve low collisions, a high packet delivery ratio, and high power efficiency. More specifically, our contributions take the following thrusts: (I) We present a probabilistic and distributed coordinated MAC protocol, RIH-MAC, employing receiver-initiated transmissions, in order to control medium access in a scalable and harvesting-aware fashion. (II) Due to the receiver-initiated design, RIH-MAC not only substantially reduces overhearing, but also achieves a lower collision probability. (III) RIH-MAC is applicable to a large family of nanonetwork applications and two network topologies: centralized and distributed.

The remainder of this chapter is organized as follows. We first introduce related work in MAC design for nanonetworks in Section 6.1. Next, we introduce the system model of nanonodes and characterize the nanonetwork in Section 6.2. In Section 6.3, the RIH-MAC protocol is described, and in Section 6.4 it is evaluated through simulation. Finally the chapter is concluded in Section 6.5.

6.1 RELATED WORK

Due to special characteristics of nanonetworks, traditional wireless MAC protocols (e.g., TDMA, CDMA, CSMA/CA) or sensor network protocols (e.g., S-MAC [123], X-MAC [124]) are not applicable in the domain of nanonetworks. Carrier-sensing

techniques in classical MAC protocols cannot be used in pulse-based communication systems since there is no carrier for sensing. Only some solutions [97] proposed for Impulse Radio Ultra Wide Band (IR-UWB) networks could be considered, but their complexity limits their usefulness in the nanonetwork scenario. For example, generating and distributing orthogonal time hopping sequences is not a lightweight process for nanodevices. Moreover, the characteristics of the THz band as well as the limited processing capabilities of nanodevices are the major factors that necessitate the redesign of protocols for the networking of nanonodes.

The main limitation for nanodevices results from the limited energy that can be stored in nanobatteries or nanocapacitors. Therefore, energy harvesting-aware protocols are required. Recently, energy harvesting-aware designs for sensor networks have been studied ([86, 84, 125, 126]). However, most of the studies cannot be applied to nanonetworks. First, the energy storage of nanonodes is limited while in previous work [84], it is mainly considered infinite or extremely large. Second, most of the schemes (e.g.,[86, 84]) are too complex to run on nanonodes. Finally, the energy harvesting rate is usually considered very close to the consumption rate in previous work [86, 84, 125, 126]. However, in nanonetworks, the harvesting rate, for most energy resources, is smaller than the energy consumption rate. This needs to be considered in the design of nanonetworks.

Receiver-initiated protocols have been investigated in duty cycle sensor networks [125, 126]. However, those methods cannot be used directly for energy harvesting environments due to the stochastic properties of energy harvesting. Moreover, it is not clear how much these receiver-initiated protocols can be effective in energy harvesting-aware protocols. Here, we investigate the use of receiver-initiated protocols for energy harvesting nanonetworks.

Recently, some MAC protocols have been proposed for electromagnetic nanonetworks [13, 122]. Jornet et al. proposed and analyzed a MAC protocol, PHLAME [13]. This protocol chooses the optimal value of code weight and repetition to address energy consumption and reliability. The performance of PHLAME is analytically studied in terms of energy consumption, delay, and achievable throughput. However, implementation feasibility and energy efficiency evaluation of the method are still open questions. Later, Wang et al. [122] proposed an energy harvesting-aware and lightweight MAC protocol. The protocol attempts to achieve fair throughput and optimal channel access among nanosensors which are controlled by a nanocontroller.

However, the focus of the work is on the scheduling of packet transmissions by the nanocontroller, and thus it uses a centralized scenario. RIH-MAC, in contrast to previous MAC protocols for nanonetworks, is a receiver-initiated protocol that operates both in distributed and centralized topologies. Furthermore, RIH-MAC can adapt itself to various energy harvesting rates.

6.2 SYSTEM MODEL

6.2.1 ENERGY MODEL

Nanonodes need energy, mainly for their communication. Due to the limited size of nanonodes, they rely on harvesting methods, where nanoscale harvesters are required. Moreover, some of nanonode applications are designed for environments with no light or heat (e.g., inside the body, in liquid). Therefore, other sources of energy such as ambient vibration are considered [2] as the main method for energy harvesting. Advancements in nanowires and nanogenerators enable the production of nanoscale harvesters. A piezoelectric nanogenerator prototype [16] has shown promising results in harvesting energy from vibration at nanoscale. In piezoelectric harvesters, the amount of harvested energy depends on the vibration rate, not the acceleration amplitude. The variation in the vibration rate will result in a stochastic model for available energy for a nanonode at different times and different locations. Vibration in various environments represents a wide range of vibration rates [76, 75], e.g., from 1 Hz (person tapping his foot) to 2000 Hz (moving vehicle). In this work, we consider two scenarios: (I) when the energy harvesting rate is greater than the consumption rate; (II) when the energy harvesting rate is less than the consumption rate and follows a stochastic process. We show how RIH-MAC can adaptively operate in both scenarios. Moreover, we consider an ultra nano-capacitor with non-linear behavior as the energy storage of each nanonode [5].

6.2.2 NETWORK MODEL

We consider two models for a network of nanonodes: centralized and distributed. Though the network model depends on the application, we believe these two of nanonetworks will be applicable for the majority of nanonetwork applications that we assumed and described in Section 2.9. In the centralized model, a central node,

called a nanocontroller [122], is responsible for coordination among nanonodes. All traffic generated by nanonodes will be transmitted to the nanocontroller, and then the nanocontroller is responsible for transfering it to the micro and macro domains. The second model, namely distributed, is an ad hoc network of nanonodes, where each nanonode can only communicate with its neighbors, i.e., nanonodes in communication range. The nanonodes are responsible for forwarding the traffic of their neighbors. The forwarding mechanism would follow the probabilistic model introduced in Section 3.1.1. In both models, we are assuming that the topology would be static, i.e., nanonodes have no mobility.

6.2.3 APPLICATION REQUIREMENTS

We assume the delay requirement of applications for nanonodes is on the order of seconds. This assumption particularly applies to scenarios where the energy harvesting rate is lower than the consumption rate. In the THz band, the available bandwidth is very large (e.g., hundreds of gigabits per second). Therefore, the delay in packet transmission and propagation is on the order of picoseconds. The only delay imposed is from the time required to harvest enough energy to exchange packets.

Furthermore, applications are not loss sensitive. Therefore, we consider only a simple acknowledgement scheme and a limited number of retries for unsuccessful transmissions. This will be the main mechanism to compensate for packet loss due to molecular absorption and thermal noise. It also handles any loss due to collisions of packets. We mainly reduce the probability of collisions as part of our MAC design as will be discussed later in Section 6.3.

Moreover, we are assuming that the packets are generated at a constant rate. Also, in the distributed network model, we assume that the forwarding mechanism is designed in a way that the forwarding traffic rate would be almost equal for all nanonodes. Therefore, the packet transmission and reception rates of all nanonodes are almost equal.

Finally, in scenarios with limited available energy (the harvesting rate is lower than the consumption rate), the packet generation rate is designed in a way that there would not be any packet overflow at the source or intermediate nodes.

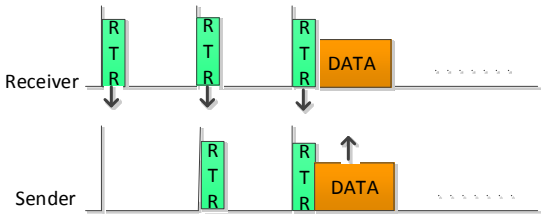


FIG. 44: *RTR* and *DATA* Packet Exchange.

6.3 RECEIVER-INITIATED COMMUNICATION

Our communication model between nanonodes is receiver-initiated. Time is divided into equal timeslots. In each timeslot, two packets are exchanged between a sender and a receiver. The receiver announces that it is ready to receive a packet by sending a *ready to receive* (RTR) packet. The recipient of the *RTR* packet may transmit a *DATA* packet accordingly. If required, the receiver can set a corresponding *ACK* field in the next *RTR* packet.

Figure 44 illustrates a sample sequence of *RTR* and *DATA* packets between a receiver and a sender. In the example, when the first *RTR* is transmitted, the sender does not receive it, which could be for many reasons such as lack of energy or communicating with another node. In the next slot, the sender receives the *RTR* packet, but does not transmit a *DATA* packet, which again could be due to many reasons, e.g., lack of energy. Upon receiving the third *RTR*, the sender transmits a *DATA* packet and the receiver receives it. The details of scheduling when to transmit and receive *RTR*s is part of RIH-MAC, which will be described later in this section. The *RTR* packet, as illustrated in Figure 45, contains the node ID, destination ID (0 for broadcast), number of neighbors, maximum known degree, current amount of energy, mode of communication (centralized or distributed), and other fields that will be described in the reminder of this section.

There are two reasons for choosing the receiver-initiated communication model. First, in a centralized topology, the nanocontroller is responsible for the management of communication among nanonodes. Due to the higher energy budget of the nanocontroller and the need for more efficient usage of energy on the transmitter side, the receiver-initiated communication model moves the load of energy consumption for the management of communication and packet handling to the nanocontroller.

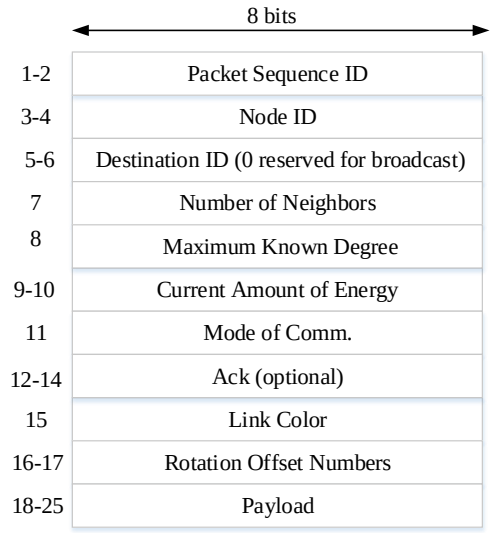


FIG. 45: *RTR* Packet.

Furthermore, since it is assumed there are abundant nanonodes, a significant portion of them may not be able to transmit a packet at each time slot. So, the receiver-initiated method enables the chance of having a fair traffic flow from different nanonodes while it does not need to be concerned about the energy level of nanonodes, as will be described in Section 6.3.1.

Second, it is better to initiate communication only when it is most likely that the receiver will have enough energy to receive a packet. Otherwise, many transmissions would be unsuccessful because of a high possibility of the receiver not having enough energy. Note that handshaking may not be an efficient method for small packet sizes that nanonodes can handle. However, there is still a need for scheduling, which is more complex for the distributed communication model. We will introduce our scheduling model for the distributed model in Section 6.3.2.

6.3.1 CENTRALIZED

In the centralized model, a nanocontroller receives information from nanonodes and then forwards it for further processing in the micro and macro domains. This model is valid in many applications where nanosensors collect information about their target phenomena. This model has been also used by other work [127, 122] and is the simplest and most scalable method to develop a nanonetwork.

To collect information, the nanocontroller repetitively broadcasts *RTR* packets. After each RTR, one or several nanonodes may transmit a *DATA* packet. The decision about which nanonode transmits its *DATA* packet follows a random process, where an arbitrary nanonode will participate with probability p. The nanonode will have enough energy for the reception of the *RTR* packet and the transmission of the consequent *DATA* packet with probability q. Also, the nanonode may have a *DATA* packet to transmit with probability r. We assume that p, q, and r are independent. Then, upon the reception of *RTR* packet, we want the nanonodes to participate in transmitting a *DATA* packet in a way that only one of them transmits. This will avoid collisions due to simultaneous transmissions which result in the waste of timeslots and energy. The expected number of concurrent transmissions X by the n nanonodes can be written as

$$E[X] = p \cdot q \cdot r \cdot n \,. \tag{39}$$

Setting the expected value equal to 1 will indicate the probability of participation to transmit a *DATA* packet by each node as

$$p = \frac{1}{q \cdot r \cdot n} \,. \tag{40}$$

The nanocontroller will transmit a sequence of *RTR* packets and receive the corresponding *DATA* packet. Each *RTR* can contain the corresponding *ACK* for the previously received *DATA* packet. This way, the participating nanonode can infer any possible collision or packet loss for retries. Furthermore, note that with the assumption of fixed size *RTR* and *DATA* packets, each nanonode knows the beginning of each timeslot for later transmissions, just after the reception of the first *RTR*.

The centralized model is scalable for a large number of nanonodes. Note that in Equation 40, the value of p could be greater than 1 when the number of nanonodes or the values of q or r are very small. In these circumstances, a value of p greater than 1, which is considered as 1, means that nanonodes should always participate in responding with a *DATA* packet. However, even with always participating, $E[X]$ would not be equal to 1, and simply would be qrn.

Also, in the case of no energy constraint, i.e., $q = 1$ and high packet rate, i.e., $r = 1$, RIH-MAC can provide a high data rate. Nanondes participate with the probability $p = \frac{1}{n}$. For example, transferring a terabyte piece of information between

two devices could be achieved by placing them in close proximity to each other, and constant transmission of *RTR* and *DATA* packets.

In some scenarios, it is required to transmit data from the nanocontroller to nanonodes, e.g., updating the functionality of nanonodes. For down-link, i.e., transmitting data from the nanocontroller to nanonodes, the same mechanism as uplink is used with a minor change in one field of the *RTR* packet. In this scenario, the *dir* field of the *RTR* packet is set to 1, which means that the nanocontroller is not expecting a *DATA* packet from nanonodes and instead will transmit a *DATA* packet. The nanonode that receives this *RTR* waits to receive the consequent *DATA* packet. The only overhead of this method is that in an energy limited scenario, this *DATA* should be sent several times until all nanonodes receive it. Assuming a similar model of participation as uplink, a *DATA* packet should be transmitted at least n times to make sure that the expected number of nanonodes that receive the *DATA* packet is n.

6.3.2 DISTRIBUTED

A distributed ad hoc formation of nanonodes looks to be unavoidable in many situations, e.g., when the nanocontroller cannot be in direct communication with all nanonodes. Here, we extend our RIH-MAC to support the ad hoc formation of nanonodes. As before, the communications are receiver-initiated, and the nanonodes may not necessarily have enough energy for communication at all timeslots.

Common random access methods such as CSMA/CA and their handshake extensions, e.g., RTS/CTS, are not applicable in nanonetworks mainly because synchronization and lack of energy make the handshake process inefficient for nanonodes. Therefore, new medium access mechanisms are required [13, 122].

Our medium access method relies on the receiver-initiated principle and distributed scheduling for nanonodes, which is energy-efficient, energy-adaptable, lightweight, and scalable. Energy adaptable means that scheme is adaptable to the various energy harvesting rates. Our scheme uses distributed scheduling for communication among nanonodes. Communication between a group of ad hoc formed nanonodes can be modeled as an edge-coloring problem, which is to determine the minimum number of colors needed to color the edges of a graph such that two edges incident on a common node do not have the same color. Each pair of nanonodes that are in the communication range of each other will have an edge between them.

All incident edges of a node should have different colors. Each color represents the timeslot in which a nanonode can communicate with one of its neighbors.

The edge coloring problem is NP-complete, and by Vizing's theorem [128], the number of colors needed to edge color a graph is either its maximum degree Δ or $\Delta + 1$. Most edge coloring solutions are centralized. Here, we are looking for a lightweight distributed solution. Among distributed solutions, we adopt the solution in [129] with minor changes. This method can color a graph with $(1+\epsilon)\Delta$ colors, for any positive ϵ in $O(\log \log n)$ rounds, where n is the number of nodes. The method finds a coloring solution for the problem with a high probability close to 1. Most of other distributed and deterministic solutions such as [130] are more complex and also do not offer a significant performance improvement. However, this algorithm satisfies the simplicity and distributed properties that we require. When this scheme fails to color properly, it can be run again at a low cost. Note that even though a network of nanonodes will be mainly static, its formation and topology can be dynamic over time (due to failure of nanonodes, or adding or removing some nanonodes), and therefore coloring will need to be run again.

Our distributed edge coloring algorithm is shown in Algorithm 3. Each edge $w = (u, v)$ between two arbitrary nanonodes u and v is initially given a palette of $(1+\epsilon) \cdot \Delta'$ colors, where Δ' is the maximum known degree of graph and is transferred in RTR packets. This palette is recorded locally at each nanonode. The formation of this palette is also done through receiving and transmitting some initial *RTR* packets where no *DATA* packets are sent in reply. A new nanonode that has no color assigned for its edges will transmit zero in the *color* field of its *RTR* packet. The main coloring process occurs in rounds. In each round, each uncolored edge independently picks a tentative color uniformly at random from its current palette. If no other edges of nodes u and v are using this color, it is picked as the final color of edge w. Otherwise, the coloring of this edge will be tried again in the next round. At the end of each round, the palettes are updated in the obvious way: colors successfully assigned are deleted from the current palette. The duration of each round would be equal to the exchange of *RTR* packets to announce the selected colors and receiving the selected colors from neighbors. Therefore, to reach the agreement or disagreement on a color with all neighbors through *RTR* packets, at most $2(\Delta + 1)$ timeslots are required for each round with the assumption of no *RTR* packet failure. If *RTR* packet failures are considered, more rounds are required. A colored graph is illustrated in Figure 46.

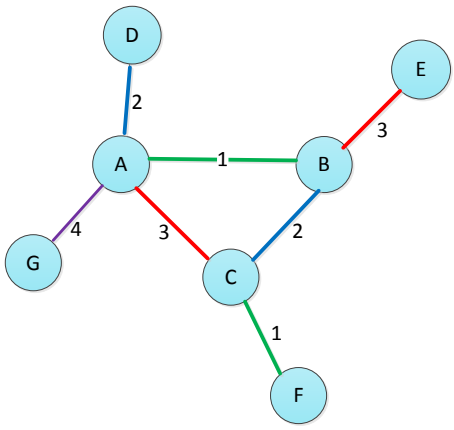


FIG. 46: A Colored Graph. Here each number represents a different color.

Colors are also labeled with numbers.

Each link between two nanonodes is bidirectional. One way to schedule the direction of communication is to extend Algorithm 3 to assign two colors per edge. However, since we assume a nanonode cannot transmit and receive at the same time, it would be similar to switching between the transmission and reception states, consecutively. For simplicity, we assume consequent changes of the communication direction as shown in Figure 47. A node with a lower ID, here alphabetically ascending, sends in the first slot and receives in the following slot for each link. For example, for the link with color 2 between nodes B and C, first B plays the role of sender at slot 3 (depicted as 2S) and C plays the role of receiver (depicted as 2R). In the next slot (4), B receives (depicted as 2R) and C transmits (depicted as 2S). Recall that the exchange of a *RTR* and *DATA* packet occurs in each timeslot with the receiver initiating it. Note that slots 7 and 8 are not used by B and C. It may appear to be a waste of slots, however, this is the cost for communication without collisions. We call the sequence of timeslots (here, eight timeslots) a *cycle*, which is repeated over and over.

Distributed RIH-MAC avoids collisions due to concurrent transmissions and is preferred to random access methods. First, the traffic rate of nanonodes are very similar to each other, so, there is no need to provide more access to the medium for one nanonode over another. Second, although there could be timeslots in DRIH-MAC that are not used by nodes with fewer neighbors, it is acceptable in scenarios where

Algorithm 3: Coloring Algorithm for DRIH-MAC

```
Void Color()
    output: Colors for each link
    Estimate the number of neighbors by listening to RTR packets;
    Announce my presence to neighbors with RTR packets;
    For link w between u and v, select a palette of colors with d = (1 + ε)Δ′
    colors;
    while w with unknown color
        select one color randomly from palette;
        if color is the same for w by both u and v
            Finalize the color;
```

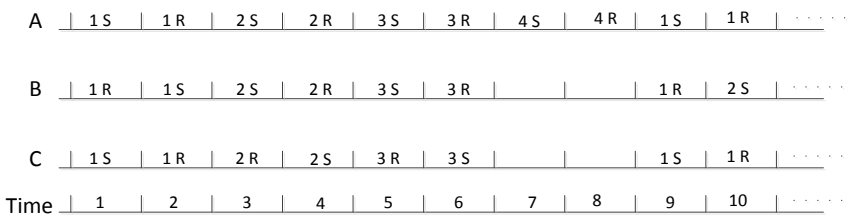


FIG. 47: Example Communication in DRIH-MAC. The nanonodes A, B and C from Figure 46 are shown. S indicates the sending mode, and R indicates receiving mode. The number preceding S/R indicates the *color*.

the energy harvesting rate is lower than the consumption rate, because some slots eventually will not be utilized due to lack of energy. An optimum energy consumption mechanism can coordinate its communication schedule with these empty slots to maximize energy and timeslot utilization.

DRIH-MAC still suffers from the *hidden terminal problem*. For example, when A is transmitting to D and B is transmitting to C, there could be problem at C in distinguishing pulses from B and A. Low code weights can be one approach to mitigate this problem. Another approach is to select the direction of communication to avoid hidden terminals. Nevertheless, finding the best approach is part of our future work.

6.3.3 ENERGY CONSUMPTION SCHEDULE

Distributed RIH-MAC can be executed stand-alone if there is no energy limitation on nanonodes. However, a *coordinated energy consumption schedule* (CECS) between two communicating nanonodes is required to achieve the highest performance. When there is no such coordination, many *RTR* packets would be sent with no *DATA* packet response. Similarly, transmitters may listen for *RTR* packets but receive no *RTR* packets due to lack of energy at the receiver side to transmit the *RTR* packet. In both scenarios, energy is wasted. Therefore, the *CECS* scheme tries to predict the energy level of each neighbor nanonode as well as their next consumption model to avoid these situations. Since the process of energy harvesting for neighbor nanonodes is not known exactly, *CECS* would be predictive. The prediction acts based on the amount of available energy of neighbors during the previous slot (which has been received in *RTR* packets) and a pre-defined consumption model. While CECS is a probabilistic approach, it improves energy consumption significantly.

We assume that nanonodes follow a similar harvesting model. The amount of current energy is received from each neighbor through *RTR* packets, which also contain the number of neighbors. We assume there is an *optimum policy*, which specifies for each nanonode how much energy should be spent per level of energy as we introduced it in Chapter 5.

Once there is an optimum energy consumption policy, we define the amount of energy consumption per *cycle*. For example, in Figure 46 for nanonode B with 3 neighbors, if, at an arbitrary cycle C_i, the optimal policy determines that only two packets can be received from the neighbors, CECS indicates the policy 2, in which

TABLE 15: Patterns Corresponding to Various Policies for Node B with 3 Links (policy number is equal to the number of receptions in one cycle).

(a) Pattern

Policy	**Pattern**
0	0 0 0
1	1 0 0
2	1 1 0
≥ 3	1 1 1

(b) Selected Pattern for Cycle C_i

Policy	**Pattern**
0	0 0 0
1	1 0 0
2	**1 1 0**
≥ 3	1 1 1

(c) Shift in Pattern for Cycle C_{i+1}

Policy	**Pattern**
0	0 0 0
1	**1 0 0**
2	*0 1 1*
≥ 3	1 1 1

links 1 and 2 are selected for communications. This selection is represented as a pattern of `0`s and `1`s. Table 15 presents a sample pattern for node B. At cycle C_i, the selected policy for node B would be policy number `2` as shown in Table 15. Next, at cycle C_{i+1}, based on the amount of energy, the oprimal policy selects another pattern, for example policy `1` in Table 15.

To provide a fair data flow among all neighbors, the selected pattern rotates at the end of each cycle. For example, the pattern for policy `2` after rotation would be `0 1 1` at cycle C_{i+1}, and remains until the next selection of policy `2`, after which the pattern will rotate again. All nanonodes will use the same pattern for different levels of energy. Table 16 shows the pattern for nodes with four and five links. Although nanonodes follow the same pattern, they will be independent in their own rotation. The *rotation offset number* for each nanonode is transferred in the *RTR* packets. Moreover, the patterns for transmission and reception are independent. A receiver

TABLE 16: Patterns Corresponding to Various Policies.

(a) Pattern for 4 Links

Policy	**Pattern**
0	0 0 0 0
1	1 0 0 0
2	0 1 0 1
3	0 1 1 1
≥ 4	1 1 1 1

(b) Pattern for 5 Links

Policy	**Pattern**
0	0 0 0 0 0
1	1 0 0 0 0
2	0 1 0 1 0
3	1 0 1 0 1
4	0 1 1 1 1
≥ 5	1 1 1 1 1

decides to transmit its *RTR* if it predicts that the transmitter is scheduled to receive the *RTR* based on the previous received rotation offset number. However, since this prediction can be incorrect, some *RTR*s may still be wasted, and consequently no *DATA* reply is received. This is avoidable only if the nanonodes decide about their energy consumption optimization model together, which seems to be implementable only with methods having significant overhead such as periodic status update packet exchange. Therefore, here we do not evaluate such a solution.

At each timeslot of a cycle, the transmitter S waits to receive a *RTR* from the receiver only, if based on the schedule, it expects a *RTR* from the receiver. Similarly, a receiver will transmit a *RTR* only if based on the transmitter schedule, it predicts that the transmitter will be waiting for a *RTR* to send its *DATA*. Note that these controls and predictions are simple enough to run on a nanonode. Through this method, the transmitter does not consume energy for the reception of *RTR* when one is not sent. Also, the receiver will not transmit any *RTR* if it predicts that the transmitter is not scheduled to receive the *RTR* and send a *DATA* packet.

A detailed analysis that ensures the existence of slots in which both the transmitter and receiver will be scheduled to send and receive at the same time can be

found in Appendix A. Briefly, it can be described as follows. When the transmitter and receiver do not happen to have `1` in their pattern at the same timeslot, they will jump into other states of energy due to changes in energy consumption and harvesting. Therefore, they will go to another state and pattern where they will eventually exchange packets. To make it clearer, we also show the measurements in simulation results (Section 6.4), which numerically evaluate the performance of *CECS*.

6.4 PERFORMANCE EVALUATION

We ran several experiments to evaluate the performance of RIH-MAC. For our simulation, we modified and enhanced the *Nanosim* module [127], which enables simulation of electromagnetic nanonetworks in ns-3. The major modifications were to the energy module and channel model. Nanonodes have harvesters that follow the harvesting model developed in [5]. To evaluate the effect of the harvest rate, we characterize the harvest rate as a probability distribution function, where it is discretized to adapt to the simulation environment. Each nanonode has an ultra-nanocapacitor as the energy storage with 100 picojoule capacity.

Nanonodes are considered to be operating in an environment with 10% water vapor with the corresponding channel path loss model [5] in the 100-300 GHz frequency band. Energy consumption is modeled as 1 femtojoule for the transmission of each pulse and 0.1 femtojoule for the reception of each pulse [5, 131, 122]. The size of packets is selected based on the method we developed in [17], where we model and find the optimum packet size for several optimization functions. In these experiments, we set the size of RTR packets to 25 bytes and the size of DATA packets to 250 bytes. There is always a back-log of packets ready in a queue to transmit. We present the results of simulation for the centralized and distributed RIH-MAC in the following sections.

6.4.1 CENTRALIZED

In this scenario, nanonodes are distributed in a sphere with a radius of 10 mm. A nanocontroller is placed in the center. The nanonodes can communicate directly with the nanoncontroller. Every 100 ms, the nanocontroller transmits a *RTR* packet and waits for the reception of a *DATA* packet from one of the nanonodes. Nanonodes decide on their probability of transmitting a *DATA* packet based on (40). Figure 48 illustrates the percentage of time the nanocontroller receives a *DATA* packet. The

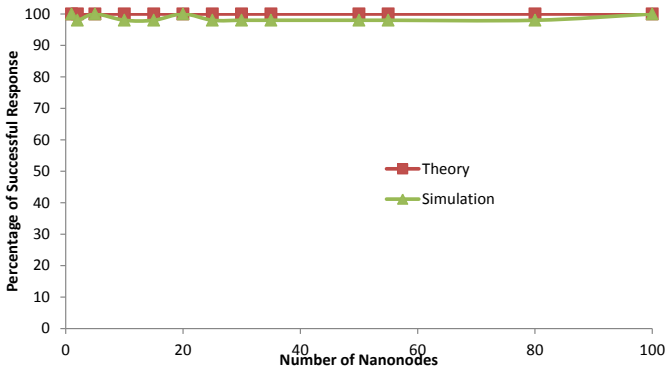


FIG. 48: Percentage of *DATA* Packet Receptions in Response to *RTR* Packet Transmissions in the Centralized Topology.

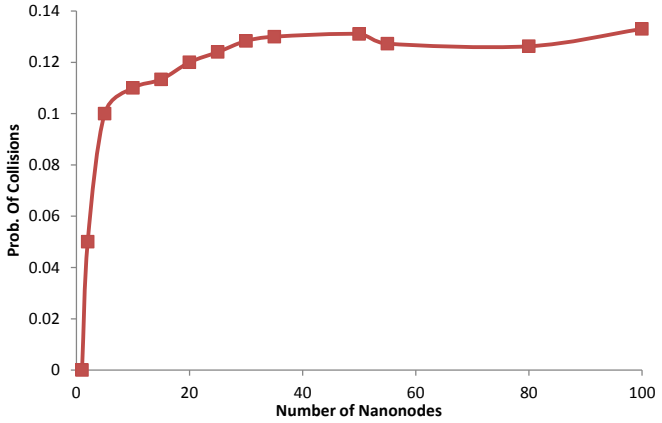


FIG. 49: Probability of Collisions in the Centralized Topology.

theory and simulation results are very close. As can be seen, RIH-MAC is scalable, i.e., with the growth in nanonodes, the percentage of *DATA* receptions remains almost the same. Also, as illustrated in Figure 49, the probability of collision (i.e., simultaneous transmission of two or more nanonodes) becomes almost constant with an increase in the number of nanonodes.

6.4.2 DISTRIBUTED

In this scenario, nanonodes are distributed uniformly in a cube of size 100 × 100 × 10 mm. Before evaluating the performance of the *CECS*, we first show the

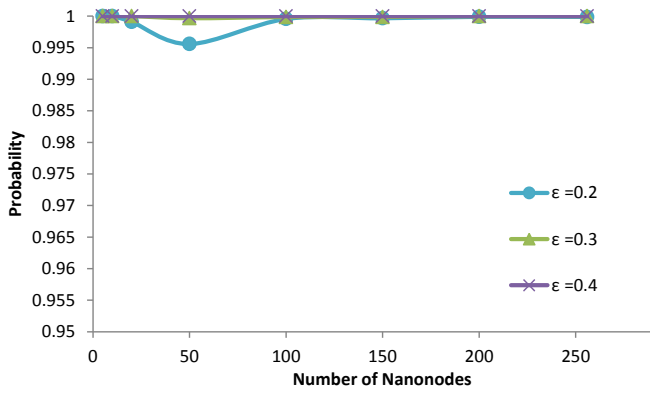


FIG. 50: Probability of Successful Edge Coloring.

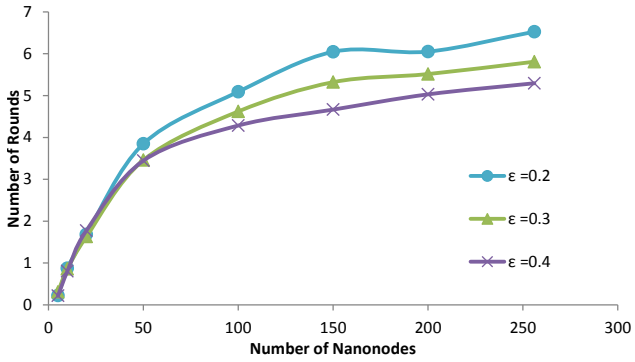


FIG. 51: Number of Rounds to Color Edges.

performance of edge coloring. We want to show (I) the probability of successful coloring and (II) the time it takes to color. Figure 50 shows the probability of successful coloring of the nanonode graph for various values of ϵ. As can be seen for all values, the probability of success is more than 99%, and the higher ϵ, the higher the probability of successful coloring.

Figure 51 depicts the number of rounds required until all edges are colored properly. Clearly, for a higher number of nanonodes, it takes more rounds to color, but it still is a reasonable number of rounds. Recall that the duration of one round is equal to the exchange of $2 \cdot (\Delta + 1)$ *RTR* packets. Since the duration of *RTR* packets is very short, the scheme converges quickly, e.g., less than one nanosecond in the scenario with no energy limit and 256 nanonodes.

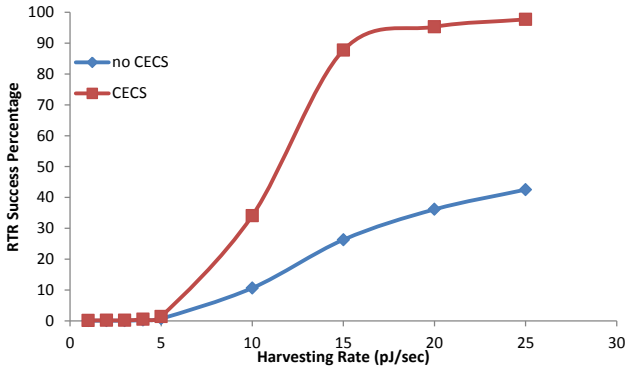


FIG. 52: *RTR* Success Percentage with Exponential Energy Harvesting.

To evaluate the performance of CECS, we define the following metric.

$$RTR\_Success = \frac{RTR_c}{RTR_c + RTR_u + RTR_w},$$

where RTR_c is the number of *RTRs* with a successful *DATA* response, RTR_u is the number of *RTRs* which are not heard by the targeted sender due to lack of energy, and RTR_w is the number of *RTRs* which are received, but cannot be replied to due to lack of energy. Note that the value of RTR_w for CECS is zero since a nanonode will not listen to *RTRs* if it knows that it will not have energy for transmission.

Figure 52 illustrates the performance of CECS in comparison to the scenario where there is no scheduling of the transmission of *RTR*s. CECS achieves close to 100% success as the harvesting rate increases. The no-CECS case has a slower slope of improvement. The *RTR* success percentage is independent of the number of nanonodes as illustrated in Figure 53.

In general, as the harvesting rate is increased, *RTR_Success* becomes closer to 100% because energy would exist at all times, and RTR_u becomes zero. This observation can also be seen in Figure 54, where the no-CECS scheme becomes closer to the CECS faster for the lognormal distribution of energy arrival as compared to the exponential distribution used in Figure 52.

Next, we measure the fairness index for communication with neighbors. Let x_i represent the number of successful packet exchanges with the ith neighbor, then the fairness index for communication with n neighbors is defined as

$$\mathcal{J}(x_1, x_2, \ldots, x_n) = \frac{(\sum_{i=1}^{n} x_i)^2}{n \cdot \sum_{i=1}^{n} {x_i}^2}.$$

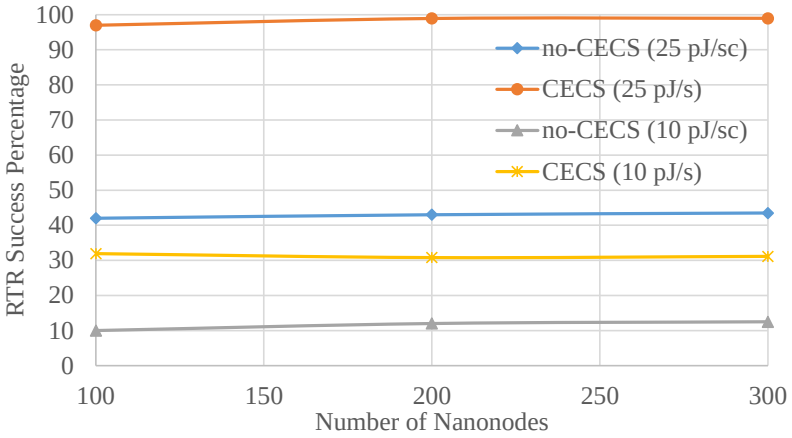


FIG. 53: *RTR* Success vs. Number of Nanonodes with Exponential Energy Harvesting Rate.

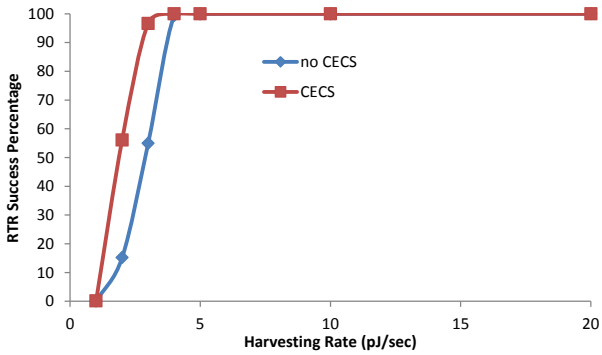


FIG. 54: *RTR* Success Percentage with Lognormal Energy Harvesting - $\sigma^2 = 0.5 \cdot \mu$.

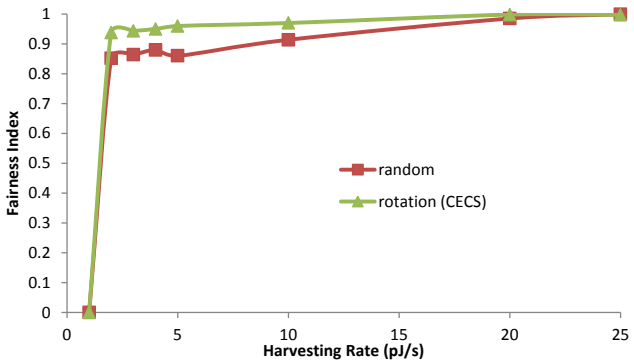


FIG. 55: Fairness Index vs. Harvesting Rate - Poisson Energy Arrival.

As shown in Figure 55, CECS achieves a better fairness index than the random selection of neighbors at each cycle. Furthermore, it can be observed that with an increase in the harvesting rate, fairness is increased, which actually occurs because of a more successful chance of message reception. The fairness index, indeed, confirms that not only will CECS result in communication between a nanonode and all of its neighbors, but it will do so in a balanced fashion.

6.4.3 CAPACITY UTILIZATION

In this experiment, we want to evaluate the performance of DRIH-MAC with a random MAC protocol to evaluate the utilization of energy harvesting rate. In this scenario, a nanonode will transmit *RTR* packets constantly. That is, immediately after the reception of the corresponding *DATA* packet, it will transmit the next *RTR* and so on. Clearly, with lack of energy for either the transmission or reception, the packet transfers will not occur. The energy is utilized properly only if a transmitted packet is received. Therefore, the utilization is defined as

$$\mathbb{U} = \frac{Recv}{Recv_{max}}, \tag{41}$$

where $Recv$ represents the number of successful receptions and $Recv_{max}$ represents the number of receptions for the maximum harvesting rate.

As shown in Figure 56, DRIH-MAC outperforms random transmission of packets. The difference is higher for a moderate harvesting rate, i.e., 5 to 50 pJ/s. In this scenario, there are no collisions between these two nanonodes.

To investigate the effect of collisions where there is more than one neighbor, the

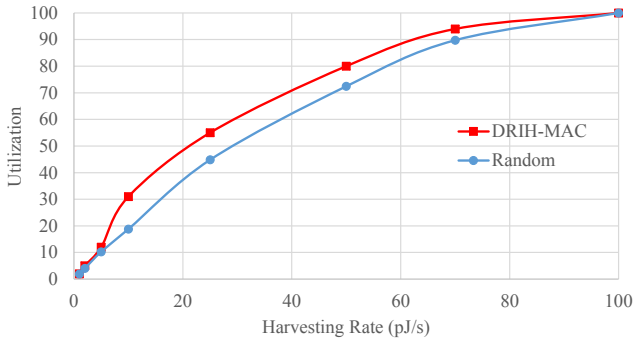


FIG. 56: Energy Utilization of a Single Link.

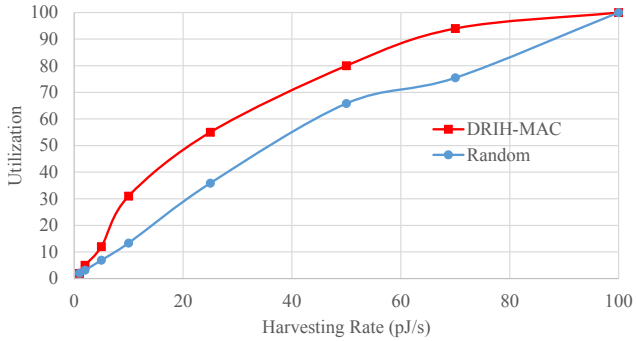


FIG. 57: Energy Utilization of a Single Link in a Network.

following scenario is considered. We evaluate a scenario where 100 nanonodes are distributed uniformly in a cube of size $1 \times 1 \times 0.05\ cm^3$. The utilization $\mathbb{U}$ is illustrated in Figure 57. DRIH-MAC again performs better than the random transmission of packets, with a utilization more than 10% higher than the random protocol. In this scenario, the utilization is reduced for the random protocol since there is no coordination for transmissions among neighbor nanonodes.

Figure 58 illustrates the energy utilization for various numbers of nanonodes. With an increase in the density of the network, more collisions among simultaneous transmission occurs due to the existence of hidden terminals. Consequently, the energy utilization drops with the increase in the number of nanonodes. However, DRIH-MAC still outperforms the Random protocol. One interesting observation is that the DRIH-MAC protocol performance for the energy harvesting rate of 10 pJ/s is very close to the performance of the Random protocol with a higher energy harvesting rate, i.e., 25 pJ/s. This observation represents the efficiency of DRIH-MAC in energy

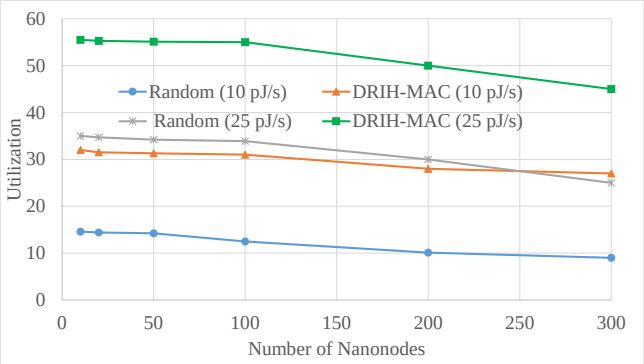


FIG. 58: Energy Utilization versus Number of Nanonodes.

utilization.

6.5 SUMMARY

In this chapter, we introduced a receiver-initiated MAC protocol for electromagnetic wireless nanonetworks. Nanonodes of such a network rely on energy harvesting to supply energy for their communication. Our receiver-initiated protocol, RIH-MAC, takes into account the energy harvesting properties of nanonodes, where they may form a centralized or distributed network. RIH-MAC is scalable with the increase in the number of nanonodes and also leads to a low number of collisions. This protocol is adaptable to be deployed in a large family of nanonetwork applications, where delay and packet loss are not hard QoS requirements. RIH-MAC is developed based on a receiver-initiated communication model, which results in a better utilization of harvested energy.

CHAPTER 7

EVALUATING SAMPLE APPLICATIONS

With the current state-of-art in nanonetworks, it is still too early to evaluate an application comprehensively and accurately. However, we can evaluate the performance of some basic applications with the protocols and schemes we developed in previous chapters. We evaluate two applications. First, we evaluate the performance of a medical monitoring application in which nanonodes will transfer their measurement of intra and/or on body phenomena to the micro and macro domain. This application is representative of a large category of nanonode applications. In the second application, we evaluate a new emerging application of nanonodes in the creation of a wireless network on chip (NoC). This application shows the performance of the schemes and protocols in a very dense network with a high traffic rate.

7.1 MEDICAL MONITORING APPLICATION

In this section, we evaluate the use of the RIH-MAC protocol for a simple application of medical monitoring. The goal is to evaluate the deployment of RIH-MAC for a particular application, but RIH-MAC is a general MAC protocol that could cover many applications as long as the proper design selection of system parameters are met in terms of number of nanonodes, energy storage capacity, and energy harvesting rate. Here, we show this design for a particular blood monitoring application, which can help the diagnosis, prevention, and cure of many diseases such as diabetes, blood pressure disorders, and various infections.

The scenario is as follows. The nanonodes are distributed in the veins along the arm. The number of nanonodes required for effective measurement depends on the fabrication of devices and the required measurement accuracy. However, with artery diameters between 0.5-10 mm and nanonodes of 10 μm, there could be 1-10 nanonodes at each point, with the assumption that nanonodes do not occupy more than $0.5 - 1\%$ of artery diameter, to avoid interference with blood flow.

We assume a network of 300 nanonodes uniformly distributed in an area of $30 \cdot 10 \cdot 10\ cm^3$. Moreover, we consider nanonodes to be operating in an aqueous environment

TABLE 17: Simulation Parameters.

Duration	60 s
Packet Interval	1 s
RTR Packet Size	25 B
DATA Packet Size	250 B
Harvesting Rate	0.2-5 pJ/s
Nanonode Communication Range	15 cm
Forwarding Value (g)	1

since between 50-70% of human body is composed of water. Refer to Appendix B for the derivations of path loss in an aqueous environment

The nanonodes sense various blood components (e.g., glucose, cholesterol). There is one nanocontroller that will gather results and send them to the interface outside of the body. The nanocontroller has higher energy storage, 300 pJ, and can harvest energy at the maximum harvest rate of 20 pJ/s. It is assumed that the nanonodes will harvest energy from the motion of the body. Various parts of the body can generate vibrations over a wide range of rates, e.g., from 1 Hz (person tapping his foot) to more than 300 Hz (person running) [76, 75]. Considering the nanowire energy harvester model [5], these vibration rates will result in energy harvesting rates of approximately 0.2 to 10 pJ/s. The other parameters of our scenario are presented in Table 17.

Not all nanonodes can communicate directly with the nanocontroller. Therefore, they are responsible for forwarding the traffic of other nanonodes towards the nanocontroller. The main metric to evaluate the performance of RIH-MAC is to show the delay in receiving the recent blood monitoring information. We are assuming that this information is required at least every 5 seconds. Figure 59 illustrates the delay corresponding to various energy harvesting rates. As can be seen, for any energy harvesting larger than the 0.5, the delay would be less than 5 seconds for *RIH-MAC* while the energy harvesting should be almost 1 pJ/s for *Random* MAC to achieve delay less than 5 seconds. Notice again that this delay is mainly due to the waiting time to harvest enough energy for communication.

7.2 NOC APPLICATION

The nanonodes will be deployed mainly in applications, where the size of nanonodes are limited, e.g., inside the body, attached to a paper. However, if this size

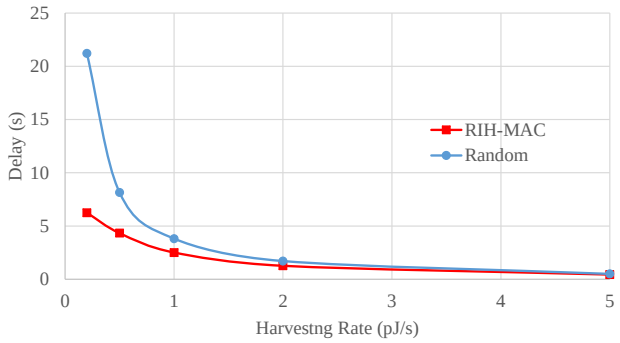


FIG. 59: Delay for Various Harvesting Rate.

limitation is not considered, the high data rate of THz can be utilized when a large energy storage and a high energy harvesting rate is considered. In this section, we analyze those parameters. Also, we show how this could be used in a sample application of network on chip systems.

Considering a symbol interval of $\beta = 100$ ps, it is possible to transfer 10^{10} bps, or 10 Gbps, when RS-TOOK is used. Since with current energy harvesting technology, it is not likely to harvest considerable amounts of energy in less than a second, we assume the energy for communication during one second should have already been stored. This means for code weight W, the energy requirement would be

$$E = \frac{W \cdot E_{tx}}{\beta}, \tag{42}$$

where E_{tx} is the energy for the transmission of one symbol. Substituting the corresponding values in (42) will results in $5mJ/s$. Therefore, $5mJ$ would be the minimum energy storage required for a nanonodes to achieve the upper bound of data rate.

With the assumption of having nanonodes with a large energy storage, at least $5mJ$, we now look at their usage in a high energy consumption rate and high data rate system, i.e., a network of cores. Network on Chip (NoC) defines the dominant paradigm to realize Chip MultiProcessor (CMP) systems through creating on-chip interconnections. In other words, it applies the principles of packet switching networking to on-chip communications. The NoC design process demands a high data rate of communication [34]. Particularly, the issue is related to the significant growth communications between cores in a mesh grid network. Traditional wire solutions limit the scalability and efficiency of NoC solutions [34]. Therefore, wireless networking of cores is favored. However, wireless NoCs (WNoC) are limited because of the

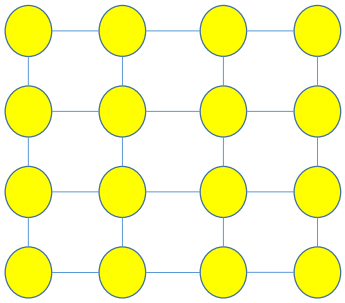


FIG. 60: A grid of 4×4 cores.

impossibility of integrating at least one antenna within each core, as future metallic antennas will be hundreds of micrometers long [132] while cores continue to shrink until reaching sizes of a few hundreds micrometers.

The unique properties of graphene antennas enable producing 5 μm long and 1 μm wide antennas to radiate in the Terahertz band [35]. This antenna enables integration of one antenna per core as well as providing data rates up to tens of Terabits per second (Tbps) [35]. In this way, nanonetworks can be used to create WNoC [36]. As an application of RIH-MAC for scenarios with unlimited energy harvesting, we simulate the performance of a NoC with RIH-MAC. It is assumed that cores are organized in a grid topology as illustrated in Figure 60. In simulations, a grid of 16×16 cores in a 256 mm^2 area is considered. Source rates which are produced by each core are related to the packets that are generated by the core itself or a packet that should be forwarded to other nodes. We want our evaluation be independent of the forwarding mechanism. Therefore, we do not include any specific traffic for packets and we assume that each core transmits a packet to each neighbor based on the coloring scheme. Note that for fixed grid topologies, the best coloring scheme is designed permanently to maximize the throughput.

Figure 61 illustrates the throughput of RIH-MAC in comparison to a random MAC protocol. RIH-MAC can handle the traffic generated by cores completely while random protocol performance degrades as the source rate is increased and is almost 50% for 20 Gbps/core traffic.

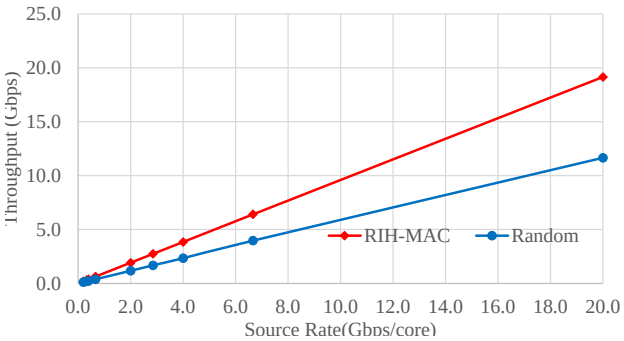


FIG. 61: Throughput of a Multi-Core System.

7.3 SUMMARY

In this section, we evaluated the performance of RIH-MAC in two possible application of nanonetworks. We first showed the performance of a medical monitoring application, where the nanonodes rely on energy harvesting. Simulation results show that RIH-MAC has a better performance in comparison with a random MAC protocol. Furthermore, we presented an application of RIH-MAC in *NoC* systems where several cores communicate with each other at very high rates. RIH-MAC can handle these very high data rates and has 50% better performance than a random protocol.

CHAPTER 8

CONCLUSION AND FUTURE WORK

8.1 SUMMARY

In this dissertation, we investigated the main issues in the networking of nanonodes in the context of energy harvesting. Because of the limitation on the availability of energy for communication at each point of time, new protocols and schemes are required to maximize the utilization of harvested energy and achieve the maximum performance. Toward this goal, we addressed these problems. First, a model for the simultaneous evaluation of energy harvesting and consumption processes was developed. The model provides the flexibility to be used in various traffic schemes and network topologies. In the energy harvesting process, both the exponential and general distributions for energy arrival were investigated. Second, the optimization of some parameters, i.e., packet length, code weight and repetition, which can provide the optimal usage of energy while satisfying some of the application requirements, was evaluated. Next, we defined the problem of optimal energy consumption policy where the stochastic energy harvesting is considered. Next, we developed a receiver-initiated energy harvesting-aware MAC (RIH-MAC) protocol, which operates in both centralized and distributed topologies of nanonodes. Finally, these models are combined to develop and evaluate the performance of two sample applications in a nanonetwork.

Our contributions are summarized as follows.

- We developed a model that shows the effect of various parameters on the combined process of energy harvesting and energy consumption. The model showed that packet size, code weight, and repetition can affect the process significantly [19, 17].

- We developed a multi-objective optimization model which indicates the optimum values for packet size, code weight, and repetition based on the application requirements in terms of delay, reliability, and energy consumption [18, 17].

- We developed a Markov decision process which can specify the amount of energy consumption for each timeslot. The goal is to maximize the utilization of energy. The process takes into account both the rate of transmissions and the rate of receptions. We also analyzed the required energy storage capacity which can satisfy the conditions of not going to the *full* energy state or the *out* of energy state. We also developed a heuristic model which performs close to optimal. Our scheme works for both linear and non-linear energy storage [20].

- We developed a receiver-initiated energy harvesting-aware MAC protocol (RIH-MAC) for communication among nanonodes. Nanonodes can communicate with a nanocontroller, which forms a centralized topology. Moreover, nanonodes can form a multi-hop ad hoc network. RIH-MAC will indicate the schedule of communication for each nanonode with its neighbors based on a distributed coloring algorithm. In addition, each nanonode predicts the availability of energy at its neighbors to minimize energy waste [21].

- Finally, we developed a sample application of medical monitoring where a collection of nanonodes take measurements of blood and transmits it to a gateway through a multi-hop network. We also showed the performance of our scheme in the context of no energy limitation. This could have applications in domains such as wireless network-on-chip. We showed that RIH-MAC can provide a high throughput for such an application [22].

8.2 FUTURE WORK

8.2.1 NEAR TERM

The *adaptive* scheme with its simplicity shows an acceptable level of performance in comparison with optimal model, especially in avoiding going to *full* or *out of energy* states. However, we believe that the performance of the *adaptive* scheme could be improved to achieve a higher energy efficiency closer to the *optimal* scheme. Particularly, for nonlinear storage models, this can be included as a factor for future improvements of the *adaptive* scheme.

One of the applications of nanonetworks is the Internet of nano-Things (IonT). The smaller size of nanonodes, which makes them more comfortable to use, as well

as the low energy consumption of nanonodes could be a significant motivation to develop IonT. The performance evaluation of methods that have been developed here with the operational conditions of things would be of interest. Even though the dominant applications of IonT (i.e., collecting the sensed information or a very coarse grain localization) would fit in the category of applications we considered to develop our schemes, new applications may have different requirements. For example, if objects are moved very often, the coloring approach of RIH-MAC may not be able to maintain the communication scheduling. A nanonetwork which considers a hierarchical structure with several nanocontrollers would be required. With this approach, a nanonetwork could be clustered and solutions would be scalable.

8.2.2 LONG TERM

One of the applications of nanonetworks is to create a network among nano-robots [133]. Nano-robots which are also called programmable matters or utility fogs, are a collection of tiny self-organized and self-configured robots. They coordinate with each other to accomplish the mission on demand. Having the abilities of coordinated self-assembly and self-reconfiguration could allow nanorobots to adapt to different environments on-the-fly. For example, they are particularly well suited to situations in which they must adapt to tasks not known *a priori* such as search and rescue applications in unstructured environments, planetary exploration, and deep space exploration. Sometimes, these nanorobots have the potential to exploit self-healing abilities with a reserve supply of low cost robot modules.

THz communication is a very desirable candidate method of communication between nanorobots since the energy consumption for communication in the range of centimeters is very low. The new nature of communication between a collection of self-organized nanorobots in addition to the structure of a very dense network necessitate the development of new protocols for communication among them. Moreover, if the mission is long, they may rely on energy harvesting. Therefore, again an energy harvesting-aware design is required.

As introduced in Chapter 7, WNoC could be a major application for nanonetworks. Designing the network based on the application of WNoC is an open question. WNoC may not necessarily form a grid or mesh network because of the nature of the application. Therefore, the evaluation of current RIH-MAC or tailoring it for other topologies is an interesting topic to look at. Also, since the nanonodes could

change their communication range to be able to communicate with various nodes at different times, this could open up new opportunities for more sophisticated protocols for communication among nanonodes. For example, this could include increasing the communication range for broadcast communication and reducing it for point to point communications when nanonodes are deployed as WNoCs.

REFERENCES

[1] I. F. Akyildiz, F. Brunetti, and C. Blazquez, "Nanonetworks: A new communication paradigm," *Computer Networks*, vol. 52, no. 12, pp. 2260–2279, 2008.

[2] I. F. Akyildiz and J. M. Jornet, "Electromagnetic wireless nanosensor networks," *Nano Comm. Networks*, vol. 1, no. 1, pp. 3–19, 2010.

[3] J. Wang and Q. Wang, *Body Area Communications: Channel Modeling, Communication Systems, and EMC*. Wiley, New York City, 2012.

[4] B. R. Watson PE, Watson ID, "Total body water volumes for adult males and females estimated from simple anthropometric measurements," *Am J Clin Nutr*, vol. 33, pp. 27–39, Jan. 1980.

[5] J. Jornet and I. Akyildiz, "Joint energy harvesting and communication analysis for perpetual wireless nanosensor networks in the Terahertz band," *IEEE Trans. on Nanotechnology*, vol. 11, no. 3, pp. 570–580, May 2012.

[6] I. Akyildiz and J. Jornet, "The internet of nano-things," *IEEE Wireless Communications*, vol. 17, no. 6, pp. 58–63, December 2010.

[7] S. Balasubramaniam and J. Kangasharju, "Realizing the internet of nano things: Challenges, solutions, and applications," *Computer*, vol. 46, no. 2, pp. 62–68, 2013.

[8] F. Dressler and F. Kargl, "Towards security in nano-communication: Challenges and opportunities," *Nano Communication Networks*, vol. 3, no. 3, pp. 151 – 160, 2012.

[9] J. Jornet and I. Akyildiz, "Channel capacity of electromagnetic nanonetworks in the Terahertz band," in *IEEE International Conference on Communications (ICC)*, May 2010, pp. 1–6.

[10] ——, "Low-weight channel coding for interference mitigation in electromagnetic nanonetworks in the Terahertz band," in *IEEE International Conference on Comm. (ICC)*, Jun 2011, pp. 1–6.

[11] J. M. Jornet and I. F. Akyildiz, "Graphene-based nano-antennas for electromagnetic nanocommunications in the terahertz band," in *Proc. of the European Conference on Antennas and Propagation*, April 2010.

[12] ——, "Information capacity of pulse-based wireless nanosensor networks," in *Proceedings of IEEE SECON*, 2011, pp. 80–88.

[13] J. M. Jornet, J. C. Pujol, and J. S. Pareta, "PHLAME: A physical layer aware MAC protocol for electromagnetic nanonetworks in the Terahertz band," *Nano Communication Networks*, vol. 3, no. 1, pp. 74–81, 2012.

[14] K. MacVittie, J. Halamek, L. Halamkova, M. Southcott, W. D. Jemison, R. Lobel, and E. Katz, "From "cyborg" lobsters to a pacemaker powered by implantable biofuel cells," *Energy Environ. Sci.*, vol. 6, pp. 81–86, 2013.

[15] P. P. Mercier, A. C. Lysaght, S. Bandyopadhyay, and A. P. C. K. M. Stankovic, "Energy extraction from the biologic battery in the inner ear," *Nature Biotechnology*, vol. 1, no. 30, p. 12401243, 2012.

[16] S. Xu, B. J. Hansen, and Z. L. Wang, "Piezoelectric-nanowire-enabled power source for driving wireless microelectronics," *Nat Commun*, vol. 1, pp. 1–12, Oct. 2010.

[17] S. Mohrehkesh and M. C. Weigle, "Optimizing communication energy consumption in perpetual wireless nanosensor networks," in *Proceedings of the IEEE Globecom*, Atlanta, GA, December 2013, pp. 545–550.

[18] S. Mohrehkesh, M. Weigle, and S. K. Das, *Energy Harvesting in NanoNetworks*, T. N. J. Suzuki and M. Moore, Eds. book chapter in "Modeling, Methodologies and Tools for Molecular and Nano-scale Communications," Springer, New York City, 2015.

[19] ——, "A survey on energy harvesting and consumption in electromagnetic nanonetworks," *submitted to IEEE Communication Magazine*, 2015.

[20] S. Mohrehkesh and M. C. Weigle, "Optimizing energy consumption in terahertz band nanonetworks," *IEEE Journal on Selected Areas in Communications,*, vol. 32, no. 12, pp. 2432–2441, Dec 2014.

[21] ——, "RIH-MAC: Receiver-initiated harvesting-aware MAC for nanonetworks," in *Proceedings of The First ACM Annual International Conference on Nanoscale Computing and Communication (NANOCOM)*, 2014, pp. 6:1–6:9.

[22] S. Mohrehkesh, M. Weigle, and S. K. Das, "DRIH-MAC: A distributed receiver-initiated harvesting-aware MAC for nanonetworks," *to appear in IEEE Trans. on Molecular, Biological, and Multi-Scale Communications*, 2015, Available at: http://cs.odu.edu/∼smohrehk/.

[23] H. Iwai, "Future of silicon integrated circuit technology," in *International Conference on Industrial and Information Systems*, 2007, pp. 571–576.

[24] "International Technology Roadmap for Semiconductors, 2012 update overview," Available at: http://public.itrs.net/Links/2012ITRS/2012Chapters/2012Overview.pdf, 2012.

[25] A. Tseng, K. Chen, C. Chen, and K. Ma, "Electron beam lithography in nanoscale fabrication: recent development," *IEEE Transactions on Electronics Packaging Manufacturing*, vol. 26, no. 2, pp. 141–149, 2003.

[26] M. Meyyappan, J. Li, J. Li, and A. Cassell, "Nanotechnology: An overview and integration with MEMS," in *Proceedings of 19th IEEE International Conference on Micro Electro Mechanical Systems*, 2006, pp. 1–3.

[27] B. Atakan and O. Akan, "Carbon nanotube-based nanoscale ad hoc networks," *IEEE Communications Magazine*, vol. 48, no. 6, pp. 129–135, June 2010.

[28] G. Scrinisa and K. Lyons, "The emerging nano-corporate paradigm: Nanotechnology and the transformation of nature, food and agri-food systems," *International Journal of Sociology of Agriculture and Food*, vol. 15, pp. 22 – 44, 2007.

[29] A. Prez-de Luque and D. Rubiales, "Nanotechnology for parasitic plant control," *Pest Management Science*, vol. 65, no. 5, pp. 540–545, 2009.

[30] C. I. Moraru, C. P. Panchapakesan, Q. Huang, P. Takhistov, S. Liu, and J. L. Kokini, "Nanotechnology: A new frontier in food science," *Food Technology*, vol. 57, pp. 24–29, 2003.

[31] J. W. Aylott, "Optical nanosensors-an enabling technology for intracellular measurements," *Analyst*, vol. 128, pp. 309–312, 2003.

[32] D. Tessier, I. Radu, and M. Filteau, "Antimicrobial fabrics coated with nano-sized silver salt crystals," in *Technical Proceedings of the 2005 NSTI Nanotechnology Conference and Trade Show*, May 2005, pp. 762–764.

[33] I. Genuth, "Socks with sensors to prevent injuries, improve running," Available at: http://thefutureofthings.com/news/11592/socks-with-sensors-to-prevent-injuries-improve-running.html, May 2013.

[34] J. Owens, W. Dally, R. Ho, D. N. Jayasimha, S. Keckler, and L.-S. Peh, "Research challenges for on-chip interconnection networks," *IEEE Micro*, vol. 27, no. 5, pp. 96–108, Sept. 2007.

[35] J. Jornet and I. Akyildiz, "Channel modeling and capacity analysis for electromagnetic wireless nanonetworks in the Terahertz band," *IEEE Transactions on Wireless Communications*, vol. 10, no. 10, pp. 3211–3221, October 2011.

[36] S. Abadal, M. Iannazzo, M. Nemirovsky, A. Cabellos-Aparicio, H. Lee, and E. Alarcon, "On the area and energy scalability of wireless network-on-chip: A model-based benchmarked design space exploration," *IEEE/ACM Transactions on Networking*, vol. 1, no. 99, pp. 1–1, 2014.

[37] D. Patra, S. Sengupta, W. Duan, H. Zhang, R. Pavlick, and A. Sen, "Intelligent, self-powered, drug delivery systems," *Nanoscale*, vol. 5, pp. 1273–1283, 2013.

[38] Y. J. Heo and S. Takeuchi, "Implantable biosensors: Towards smart tattoos: Implantable biosensors for continuous glucose monitoring," *Advanced Healthcare Materials*, vol. 2, no. 1, pp. 2–2, 2013.

[39] N. Agoulmine, K. Kim, S. Kim, T. Rim, J.-S. Lee, and M. Meyyappan, "Enabling communication and cooperation in bio-nanosensor networks: toward innovative healthcare solutions," *IEEE Wireless Communications*, vol. 19, no. 5, pp. 42–51, 2012.

[40] T. Nakano, A. W. Eckford, and T. Haraguchi, *Molecular communication*. Cambridge University Press, Cambridge, UK, 2013.

[41] M. Moore, A. Enomoto, T. Nakano, R. Egashira, T. Suda, A. Kayasuga, H. Kojima, H. Sakakibara, and K. Oiwa, "A design of a molecular communication system for nanomachines using molecular motors," in *Proceedings of Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops*, March 2006, pp. 553–559.

[42] M. Gregori and I. Akyildiz, "A new nanonetwork architecture using flagellated bacteria and catalytic nanomotors," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 4, pp. 612–619, May 2010.

[43] L. Parcerisa Giné and I. F. Akyildiz, "Molecular communication options for long range nanonetworks," *Comput. Netw.*, vol. 53, no. 16, pp. 2753–2766, Nov. 2009.

[44] T. Nakano, M. Moore, F. Wei, A. Vasilakos, and J. Shuai, "Molecular communication and networking: Opportunities and challenges," *IEEE Transactions on NanoBioscience*, vol. 11, no. 2, pp. 135–148, June 2012.

[45] S. Luryi, J. Xu, and A. Zaslavsky, *Future Trends in Microelectronics: Up the Nano Creek*. John Wiley & Sons - IEEE, New York City, 2007.

[46] P. Avouris, "Carbon nanotube electronics and photonics," *Physics Today*, vol. 62, p. 3440, Jan. 2009.

[47] P. Kim, "Toward carbon based electronics," in *Proceedings of Device Research Conference*, 2008, p. 9.

[48] B. Sensale-Rodriguez, R. Yan, M. M. Kelly, T. Fang, K. Tahy, W. S. Hwang, D. Jena, L. Liu, and H. G. Xing, "Broadband graphene terahertz modulators enabled by intraband transitions," *Nature Communications*, vol. 3, p. 780, 2012.

[49] Y. M. Lin and et al., "100-GHz transistors from wafer-scale epitaxial graphene," *Science*, vol. 327, p. 662, Feb. 2010.

[50] D. Grischkowsky, S. Keiding, M. van Exter, and C. Fattinger, "Far-infrared time-domain spectroscopy with terahertz beams of dielectrics and semiconductors," *J. Opt. Soc. Am. B*, vol. 7, no. 10, pp. 2006–2015, Oct. 1990.

[51] T. Kleine-Ostmann and T. Nagatsuma, "A review on terahertz communications research," *Journal of Infrared, Millimeter, and Terahertz Waves*, vol. 32, no. 2, pp. 143–171, 2011.

[52] P. Siegel, "Terahertz technology," *IEEE Transactions on Microwave Theory and Techniques*, vol. 50, no. 3, pp. 910–928, 2002.

[53] M. Tonouchi, "Cutting-edge terahertz technology," *Nature Photonics*, vol. 1, no. 1, pp. 97–105, 2007.

[54] E. Laskin, K. Tang, K. H. K. Yau, P. Chevalier, A. Chantre, B. Sautreuil, and S. Voinigescu, "170-GHz transceiver with on-chip antennas in SiGe technology," in *Proceedings of IEEE Radio Frequency Integrated Circuits Symposium*, 2008, pp. 637–640.

[55] J. Federici and L. Moeller, "Review of terahertz and subterahertz wireless communications," *J. Appl. Phys.*, vol. 117, pp. 111 101–111 122, 2010.

[56] H.-J. Song and T. Nagatsuma, "Present and future of terahertz communications," *IEEE Transactions on Terahertz Science and Technology*, vol. 1, no. 1, pp. 256–263, 2011.

[57] C. M. Armstrong, "The truth about terahertz," *IEEE Spectrum*, vol. 49, no. 9, pp. 36–41, August 2012.

[58] T. Krner, "Whats next? wireless communication beyond 60 GHz (tutorial IG THz)," Available at: https://mentor.ieee.org/802.15/dcn/12/15-12-0320-02-0thz-what-s-next-wireless-communication-beyond-60-ghz-tutorial-ig-thz.pdf, July 2012.

[59] S. Yamauchi, Y. Aoyagi, A. Okamoto, Y. Imai, and M. Tonouchi, "Effect of water cluster on absorption property in Terahertz time-domain spectroscopy," in *Proceedings of Joint 32nd International Conference on Infrared and Millimeter Waves and 15th International Conference on Terahertz Electronics*, 2007, pp. 664–665.

[60] T. Suzuki, K. Takayama, S. Yamauchi, Y. Imai, and M. Tonouchi, "Measurement of water absorption coefficient using Terahertz time-domain spectroscopy," in *Proceedings of 34th International Conference on Infrared, Millimeter, and Terahertz Waves*, 2009, pp. 1–2.

[61] V. V. Zhirnov and R. K. Cavin, *Microsystems for bioelectronics: the nanomorphic cell*, ser. Micro and nano technologies series. William Andrew/Elsevier, Oxford, UK, 2011.

[62] M. Akkas, I. Akyildiz, and R. Sokullu, "Terahertz channel modeling of underground sensor networks in oil reservoirs," in *Proc. of IEEE Global Communications Conference (GLOBECOM)*, 2012, pp. 543–548.

[63] R. Fontana, "Recent system applications of short-pulse ultra-wideband (UWB) technology," *IEEE Transactions on Microwave Theory and Techniques*, vol. 52, no. 9, pp. 2087–2104, Sept. 2004.

[64] M. Hamalainen, R. Tesi, J. Iinatti, and V. Hovinen, "On the performance comparison of different UWB data modulation schemes in AWGN channel in the presence of jamming," in *Proceedings of IEEE Radio and Wireless Conference*, 2002, pp. 83–86.

[65] L. Ge, G. Yue, and S. Affes, "On the BER performance of pulse-position-modulation UWB radio in multipath channels," in *Proceedings of IEEE Conference on Ultra Wideband Systems and Technologies*, 2002, pp. 231 – 234.

[66] M. Kocaoglu and O. Akan, "Minimum energy coding for wireless nanosensor networks," in *Proceedings of IEEE INFOCOM*, 2012, pp. 2826–2830.

[67] K. Chi, Y. Zhu, X. Jiang, and X. Tian, "Optimal coding for transmission energy minimization in wireless nanosensor networks," *Nano Communication Networks*, vol. 4, no. 3, pp. 120 – 130, 2013.

[68] J. Irudayaraj, *Biomedical Nanosensors*, ser. Pan Stanford Series on Biomedical Nanotechnology Series. Pan Stanford Publishing, Stanford, 2011.

[69] Z. L. Liangzhu Feng, "Graphene in biomedicine: Opportunities and challenges," *Nanomedicine*, vol. 2, pp. 317–324, 2011.

[70] M. Moon, "Ultra-thin e-skin could lead to advances in medicine, cool wearable computing," Available at http://www.engadget.com/2013/07/26/e-skin-tokyo-university/, July 2013.

[71] B. Buntz, "Nanotech sensors keep a close eye on orthopedic implants," *Qmed medtech pulse blog.*, March 2013.

[72] S. Sudevalayam and P. Kulkarni, "Energy harvesting sensor nodes: Survey and implications," *IEEE Communications Surveys Tutorials*, vol. 13, no. 3, pp. 443 –461, 2011.

[73] S. Chalasani and J. Conrad, "A survey of energy harvesting sources for embedded systems," in *IEEE Southeastcon*, Apr 2008, pp. 442–447.

[74] J. Gilbert and F. Balouchi, "Comparison of energy harvesting systems for wireless sensor networks," *International Journal of Automation and Computing*, vol. 5, pp. 334–347, 2008, 10.1007/s11633-008-0334-2.

[75] S. Roundy, P. K. Wright, and J. Rabaey, "A study of low level vibrations as a power source for wireless sensor nodes," *Computer Communications*, vol. 26, no. 11, pp. 1131–1144, 2003.

[76] S. Roundy, "On the effectiveness of vibration-based energy harvesting," *Journal of Intelligent Material Systems and Structures*, vol. 16, no. 10, pp. 809–823, 2005.

[77] M. Gorlatova, J. Sarik, M. Cong, I. Kymissis, and G. Zussman, "Movers and shakers: Kinetic energy harvesting for the internet of things," Available at: http://arxiv.org/pdf/1307.0044v1.pdf, July 2013.

[78] S. Roundy, P. Wright, and J. Rabaey, *Energy Scavenging for Wireless Sensor Networks: With Special Focus on Vibrations*. Kluwer Academic Publishers, New York City, 2004.

[79] P. Mitcheson, "Energy harvesting for human wearable and implantable biosensors," in *IEEE Engineering in Medicine & Biology Society*, Sep. 2010, pp. 3432–3436.

[80] M. Gorlatova, A. Wallwater, and G. Zussman, "Networking low-power energy harvesting devices: Measurements and algorithms," in *Proceedings of IEEE INFOCOM*, April 2011, pp. 1602 –1610.

[81] N. Shenck and J. Paradiso, "Energy scavenging with shoe-mounted piezoelectrics," *IEEE Micro*, vol. 21, no. 3, pp. 30–42, 2001.

[82] T. Starner, "Human-powered wearable computing," *IBM Systems Journal*, vol. 35, no. 34, pp. 618–629, 1996.

[83] A. M. Zungeru, L.-M. Ang, S. Prabaharan, and K. P. Seng, *Chapter 13. Radio Frequency Energy Harvesting and Management for Wireless Sensor Networks*, H. Venkataraman and G.-M. Muntean, Eds. CRC Press, New York City, 2012.

[84] V. Sharma, U. Mukherji, V. Joseph, and S. Gupta, "Optimal energy management policies for energy harvesting sensor nodes," *IEEE Transactions on Wireless Communications*, vol. 9, no. 4, pp. 1326–1336, April 2010.

[85] M. Rentmeesters, W. Tsai, and K.-J. Lin, "A theory of lexicographic multi-criteria optimization," in *Proceedings of Second IEEE International Conference on Engineering of Complex Computer Systems*, 1996, pp. 76–79.

[86] R.-S. Liu, K.-W. Fan, Z. Zheng, and P. Sinha, "Perpetual and fair data collection for environmental energy harvesting sensor networks," *IEEE/ACM Trans. on Networking*, vol. 19, no. 4, pp. 947–960, Aug. 2011.

[87] M. Khouzani, S. Sarkar, and K. Kar, "Optimal routing and scheduling in multihop wireless renewable energy networks," in *Proceedings of Sixth Information Theory and Applications Workshop (ITA)*, Feb 2011, pp. 1–8.

[88] J. Z. Y. Luo and K. B. Letaief, "Training optimization for energy harvesting communication systems," in *Proceedings of IEEE Globecom*, 2012.

[89] K. Wu, Y. Jiang, and D. Marinakis, "A stochastic calculus for network systems with renewable energy sources," in *Proceedings of IEEE Conference on Computer Communications Workshops (INFOCOM Workshops)*, Mar. 2012, pp. 109 –114.

[90] 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, Feb. 2010.

[91] K. Kar, A. Krishnamurthy, and N. Jaggi, "Dynamic node activation in networks of rechargeable sensors," *IEEE/ACM Trans. Netw.*, vol. 14, no. 1, pp. 15–26, Feb. 2006.

[92] J. Hsu, S. Zahedi, A. Kansal, M. Srivastava, and V. Raghunathan, "Adaptive duty cycling for energy harvesting systems," in *Proceedings of the International Symposium on Low Power Electronics and Design*, 2006, pp. 180–185.

[93] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, "Power management in energy harvesting sensor networks," *ACM Trans. Embed. Comput. Syst.*, vol. 6, no. 4, Sep. 2007.

[94] D. Niyato, E. Hossain, and A. Fallahi, "Sleep and wakeup strategies in solar-powered wireless sensor/mesh networks: Performance analysis and optimization," *IEEE Transactions on Mobile Computing*, vol. 6, no. 2, pp. 221–236, Feb. 2007.

[95] D. K. Noh and T. F. Abdelzaher, "Efficient flow-control algorithm cooperating with energy allocation scheme for solar-powered WSNs," *Wireless Communications and Mobile Computing*, vol. 12, no. 5, pp. 379–392, 2012.

[96] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Comm. Magazine*, vol. 40, no. 8, pp. 102 – 114, Aug. 2002.

[97] A. Gupta and P. Mohapatra, "A survey on ultra wide band medium access control schemes," *Comput. Netw.*, vol. 51, no. 11, pp. 2976–2993, Aug. 2007.

[98] M. Gorlatova, A. Wallwater, and G. Zussman, "Networking low-power energy harvesting devices: Measurements and algorithms," *IEEE Trans. on Mobile Computing*, vol. 12, no. 9, pp. 1853–1865, 2012.

[99] D. J. Leith and P. Clifford, "Convergence of Distributed Learning Algorithms for Optimal Wireless Channel Allocation," in *Proceedings of Conference on Decision and Control*, 2006.

[100] D. Leith and P. Clifford, "A self-managed distributed channel selection algorithm for wlans," in *Proceedings of 4th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, Apr. 2006, pp. 1 – 9.

[101] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *Proceedings of ACM MobiCom*, 1999, pp. 151–162.

[102] N. Wisitpongphan, O. Tonguz, J. Parikh, P. Mudalige, F. Bai, and V. Sadekar, "Broadcast storm mitigation techniques in vehicular ad hoc networks," *IEEE Wireless Comm.*, vol. 14, no. 6, pp. 84–94, Dec. 2007.

[103] A. Salhieh, J. Weinmann, M. Kochhal, and L. Schwiebert, "Power efficient topologies for wireless sensor networks," in *Proceedings of International Conference on Parallel Processing*, Sept. 2001, pp. 156–163.

[104] F. Li, Z. Chen, and Y. Wang, "Localized topologies with bounded node degree for three dimensional wireless sensor networks," in *Proceedings of Seventh International Conference on Mobile Ad-hoc and Sensor Networks (MSN)*, Dec. 2011, pp. 20–27.

[105] S. Ross, *Stochastic processes*, ser. Wiley Series in Probability and Statistics. Wiley, New Yoork City, 1996.

[106] L. Hu, "Topology control for multihop packet radio networks," *IEEE Transactions on Communications*, vol. 41, no. 10, pp. 1474–1481, Oct. 1993.

[107] J. Liu and B. Li, "Distributed topology control in wireless sensor networks with asymmetric links," in *Proceedings of IEEE Global Telecommunications Conference*, vol. 3, Dec. 2003, pp. 1257–1262.

[108] K. Deb, "Multi-objective optimization," in *Search Methodologies*, E. Burke and G. Kendall, Eds. Springer US, New York City, 2005, pp. 273–316.

[109] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II," in *Proceedings of International Conference on Parallel Problem Solving from Nature (PPSN)*, 2000, pp. 849–858.

[110] A. Seyedi and B. Sikdar, "Energy efficient transmission strategies for body sensor networks with energy harvesting," *IEEE Transactions on Communications*, vol. 58, no. 7, pp. 2116–2126, 2010.

[111] 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.

[112] Z. Wang, A. Tajer, and X. Wang, "Communication of energy harvesting tags," *IEEE Transactions on Communications*, vol. 60, no. 4, pp. 1159–1166, 2012.

[113] J. Lei, R. Yates, and L. Greenstein, "A generic model for optimizing single-hop transmission policy of replenishable sensors," *IEEE Transactions on Wireless Communications*, vol. 8, no. 2, pp. 547–551, Feb 2009.

[114] O. Ozel, K. Tutuncuoglu, J. Yang, S. Ulukus, and A. Yener, "Transmission with energy harvesting nodes in fading wireless channels: Optimal policies," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 8, pp. 1732–1743, September 2011.

[115] C. K. Ho and R. Zhang, "Optimal energy allocation for wireless communications with energy harvesting constraints," *IEEE Transactions on Signal Processing*, vol. 60, no. 9, pp. 4808–4818, Sept. 2012.

[116] J. Yang and S. Ulukus, "Optimal packet scheduling in an energy harvesting communication system," *IEEE Transactions on Communications*, vol. 60, no. 1, pp. 220–230, January 2012.

[117] L. Huang and M. J. Neely, "Utility optimal scheduling in energy-harvesting networks," *IEEE/ACM Trans. Netw.*, vol. 21, no. 4, pp. 1117–1130, Aug. 2013. [Online]. Available: 10.1109/TNET.2012.2230336

[118] K. Tutuncuoglu and A. Yener, "Optimal power policy for energy harvesting transmitters with inefficient energy storage," in *Proceedings of 46th Annual Conference on Information Sciences and Systems (CISS)*, March 2012, pp. 1–6.

[119] N. Michelusi, K. Stamatiou, and M. Zorzi, "Transmission policies for energy harvesting sensors with time-correlated energy supply," *IEEE Transactions on Communications*, vol. 61, no. 7, pp. 2988–3001, July 2013.

[120] D. Gunduz, K. Stamatiou, N. Michelusi, and M. Zorzi, "Designing intelligent energy harvesting communication systems," *IEEE Communications Magazine*, vol. 52, no. 1, pp. 210–216, January 2014.

[121] M. Puterman, *Markov decision processes: discrete stochastic dynamic programming*, ser. Wiley Series in Probability and Statistics. Wiley-Interscience, New York City, 2005.

[122] P. Wang, J. M. Jornet, M. A. Malik, N. Akkari, and I. F. Akyildiz, "Energy and spectrum-aware MAC protocol for perpetual wireless nanosensor networks in the terahertz band," *Ad Hoc Networks*, vol. 11, no. 8, pp. 2541 – 2555, 2013.

[123] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proc. of IEEE INFOCOM*, 2002, pp. 1567–1576.

[124] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: A short preamble MAC protocol for duty-cycled wireless sensor networks," in *Proc. of the 4th ACM SenSys*, 2006, pp. 307–320.

[125] Y. Sun, O. Gurewitz, and D. B. Johnson, "RI-MAC: A receiver-initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks," in *Proceedings of the 6th ACM SenSys*, 2008, pp. 1–14.

[126] L. Tang, Y. Sun, O. Gurewitz, and D. Johnson, "PW-MAC: An energy-efficient predictive-wakeup MAC protocol for wireless sensor networks," in *Proceedings of IEEE INFOCOM*, 2011, pp. 1305–1313.

[127] G. Piro, L. A. Grieco, G. Boggia, and P. Camarda, "Nano-sim: simulating electromagnetic-based nanonetworks in the network simulator 3," in *Proc. of Workshop on NS- 3*, Cannes, France, Mar. 2013.

[128] R. Merris, *Edge Colorings*. John Wiley and Sons, Inc., 2000, pp. 195–210.

[129] D. A. Grable and A. Panconesi, "Nearly optimal distributed edge colouring in O(log log n) rounds," in *Proceedings of the Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, Philadelphia, PA, USA, 1997, pp. 278–285.

[130] S. Gandham, M. Dawande, and R. Prakash, "Link scheduling in wireless sensor networks: Distributed edge-coloring revisited," *Journal of Parallel and Distributed Computing*, vol. 68, no. 8, pp. 1122 – 1134, 2008.

[131] T. Otsuji, S. Tombet, A. Satou, M. Ryzhii, and V. Ryzhii, "Terahertz-wave generation using graphene: Toward new types of terahertz lasers," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 19, no. 1, 2013.

[132] S.-B. Lee, S.-W. Tam, I. Pefkianakis, S. Lu, M. F. Chang, C. Guo, G. Reinman, C. Peng, M. Naik, L. Zhang, and J. Cong, "A scalable micro wireless interconnect structure for CMPs," in *Proceedings of the 15th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '09, 2009, pp. 217–228.

[133] D. D. Nicolas Boillot and J. Bourgeois, "Using nano-wireless communications in micro-robots applications," in *Proceedings of The First ACM Annual International Conference on Nanoscale Computing and Communication (NANOCOM)*, 2014, pp. 1–9.

[134] D. Segelstein, "The complex refractive index of water," Master's thesis, University of Missouri, Kansas City, Missouri, USA, 1981.

[135] J. Xu, K. Plaxco, S. Allen, J. Bjarnason, and E. Brown, "0.15 THz absorption of aqueous salts and saline solutions," *Applied Physics Letters*, vol. 90, no. 3, pp. 031 908–031 908–3, Jan 2007.

[136] D. Werber, A. Schwentner, and E. M. Biebl, "Investigation of RF transmission properties of human tissues," *Advances in Radio Science*, vol. 4, pp. 357–360, 2006.

APPENDIX A

CECS FUNCTIONALITY

Here, we analyze and prove how our coordinated energy consumption schedule (CECS) ensures packet exchange between a receiver and transmitter although they have not scheduled communication during one particular time slot. The proof follows two steps. We first show that there exists a probability for being in any of the levels for a *policy* and then show that the probability is non-zero for two nanonodes using the patterns resulting in packet exchanges.

A.1 ENERGY HARVESTING AND CONSUMPTION PROCESSES

The energy harvesting and consumption processes can be modeled as a Markov chain, which we denote as M. For each nanonode, energy harvesting follows a random variable, while energy consumption is defined based on a set of available actions on how much energy is to be consumed. We consider a discrete time model, in which the time is slotted into intervals of unit length. In timeslots of a *cycle*, some energy is harvested and added to the energy storage, and similarly some energy is consumed and deducted from the energy storage based on the consumption scheme. We assume that the energy storage is ideal and there is no significant leakage.

We denote the system states by $S = S_1, S_2, \cdots, S_s$, where $s = C + 1$ for energy storage of capacity $C \cdot E_{min}$ units of energy, E_{min} denotes the unit of energy, e.g., 1 pJ.

It is assumed that there are always packets ready for transmission. Being in the transmitter and receiver roles will consume E_{tx} and E_{rx} units of energy, respectively, to exchange a *DATA* packet and a *RTR* packet. The consumption strategy of a nanonode, i.e., the number of times the nodes serves as the transmitter and receiver per *cycle*, is denoted as $a_{(i,j)}$, for $i, j \geq 0$, which is selected from

$$A = \{a_{(0,0)}, a_{(1,0)}, a_{(0,1)}, a_{(1,1)}, \cdots, a_{(m,n)}, a_{(m+1,n)}, a_{(m,n+1)}, a_{(m+1,n+1)}, \cdots\}.$$

The action $a_{(i,j)}$ corresponds to the node being the receiver i times and the transmitter j times in the *cycle*, where the sum of the energy consumption, denoted as E_k, cannot

exceed the maximum consumption per cycle, E_c, $0 \leq E_c \leq C$, i.e., $i \cdot E_{rx} + j \cdot E_{tx} \leq E_c \quad \forall\, i, j$. We denote S_A as the number of members of A. For simplicity of presentation, we define $i \cdot E_{rx} + j \cdot E_{tx}$ as E_k, with the corresponding $a_k\,, 1 \leq k \leq S_A$. The consumption action taken for each state of energy depends on the design of consumption model. For example, in an optimum design, there would be only one action per state.

The probabilities of transferring between states depend on the current state, the amount of energy harvested, and the action taken. Formally, the state transitions can be written as

$$P(S_i, S_j) = p_u\,, \quad \sum_{x=1}^{s} P(S_i, S_x) = 1\,,\ 1 \leq i, j \leq s\,,$$

and j is specified as

$$j = i + h_u - E_k, \tag{43}$$

where $E_k < i$, $i + h_u \leq s$, $0 \leq u \leq D$, and $1 \leq k \leq S_A$. The value of j represents the energy state after the harvesting of h_u units and consumption of E_k units of energy for action a_k taken. The condition $E_k < i$ limits the actions which can be taken to avoid consuming more energy than is harvested and stored. The condition $i + h_u \leq s$ limits the harvested energy to the available capacity of energy storage. When $j = 1$, the system falls into the *out of energy* state, i.e, the node has consumed all of energy that it has stored and harvested. When $j = s$, the system falls into the *full energy* state, i.e., even after consumption, the system has stored and harvested up to the capacity C.

Now, we show that this Markov chain for the energy harvesting and consumption is *ergodic*, which means it would have a stationary solution.

A.2 MARKOV CHAIN PROPERTIES AND RELATION WITH CECS

Lemma A.2.1 *The Markov chain M is irreducible.*

Proof: From Equation (43), it is straightforward to show that any other state can be accessed in one or many transitions, i.e., $P^n(S_i, S_j) > 0$.

Lemma A.2.2 $P(S_0, S_0) > 0$

Proof: When no energy exists and none is harvested, the system stays in the same state, i.e., $P(S_0, S_0) > 0$

Lemma A.2.3 *The Markov chain is ergodic*

Proof: Using Lemmas (A.2.1) and (A.2.2), it is concluded that M is ergodic.

Corollary A.2.4 *For any ergodic Markov chain, there is a unique stationary solution with probabilities* $\pi_i > 0, 1 \leq i \leq s$.

Theorem A.2.5 *The probability of two neighbor nanonodes being in the* `1` *of their CECS scheduling pattern simultaneously is non-zero.*

Proof: Consider two arbitrary nanonodes that are not in their first state at the same time. They are in states i and j respectively, with their corresponding patterns denoted as r_i and r_j. The *rotation* of patterns means that the probability of being in any rotation offset of a pattern would be

$$p_r = \frac{1}{l} > 0, \tag{44}$$

where l is the number of neighbors for a nanonode. We define this probability for nanonodes i and j as p_{r_i} and p_{r_j}, respectively. Let us define V as the event that two nanonodes are in `1`s of their CECS scheduling pattern simultaneously. Next, we can write

$$Pr(V) = \pi_i \cdot p_{r_i} \cdot \pi_j \cdot p_{r_j} > 0 \tag{45}$$

If two nanonodes are in their first states at the same time, then the probability that they will not stay there in the future would be

$$1 - \pi_1 \cdot \pi_1 > 0, \tag{46}$$

because $\pi_1 \neq 1$. Therefore, they will go to two other states and then Equation (45) will apply to them.

APPENDIX B

CALCULATION OF PATH LOSS IN AQUEOUS ENVIRONMENT

Here, we calculate the path loss in an aqueous environment. Path loss consists of two main components: absorption loss and free space propagation loss. We present the calculation of these losses in the 0.1-10 THz in an aqueous environment.

For liquids, transmittance is related to absorbance A (not to be confused with absorptance) as

$$A = -log(T) = -log(\frac{I}{I_0}), \tag{47}$$

where I is the intensity of radiation (after transmission through liquid) and I_0 is the intensity of radiation before it passes through the material.

Similarly, the transmission (transmissivity) is given by

$$T = (\frac{I}{I_0}) = 10^{-\alpha \cdot d}, \tag{48}$$

where d represents the distance and α is the attenuation coefficient.

From (47) and (48), absorbance can be written as

$$A = \alpha \cdot d \cdot log(10)$$

The value of α is calculated as follows

$$\alpha = \frac{4 \cdot \pi \cdot k}{\lambda_0} \cdot d,$$

where λ_0 is the vacuum wavelength (the wavelength of the light in free space), and k is the imaginary part of the refractive index. The refractive index of materials varies with the wavelength. In opaque media, the refractive index is a complex number, where the real part describes refraction and the imaginary part accounts for absorption.

The attenuation coefficient (α) can be approximated with the absorption coefficient. We used the values of the absorption coefficient, collected by Segelstein

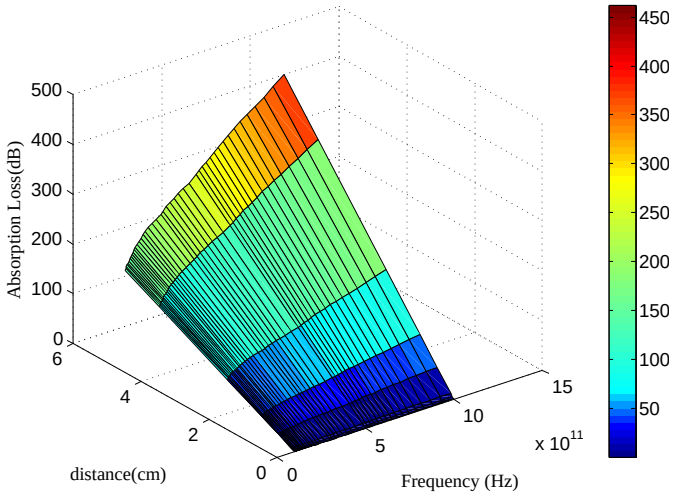


FIG. 62: Absorption Loss at Different Distances for Water.

[134, 135], to calculate the absorption loss in dB. We plot the absorption loss in Figure 62 for 0.1-1 THz and distances up to 5 cm.

As shown, the absorption loss is less than 100 dB in less than 1 cm. Also, for lower frequencies, e.g., 100-300 GHz, the absorption loss would be under 10 dB. Note that fat and muscles have lower attenuation values [136]. Therefore, the calculations here are valid for inside the body communication.

Free-space propagation loss is another parameter that affects the effective range of communications. The free-space propagation loss is defined as

$$P_L(f,d) = \left(\frac{4\pi f d}{c}\right)^2, \tag{49}$$

where f is the frequency of interest, d is the distance between the transmitter and the receiver, and c is the speed-of-light in a vacuum. The free space propagation loss in dB can be represented as

$$P_L = 10 \cdot log(P_L(f,d)) \tag{50}$$

Figure 63 illustrates the path loss with the change of frequency in the range of 0.1-1 THz for 0.01, 0.1 and 1 m distances.

Combining the absorption loss and path loss shows that communication in water would only be possible in centimeter distances. Figure 64 represents the total loss

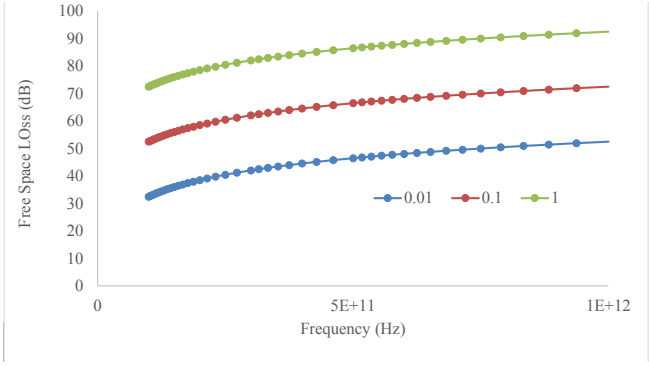


FIG. 63: Path Loss at Different Distances.

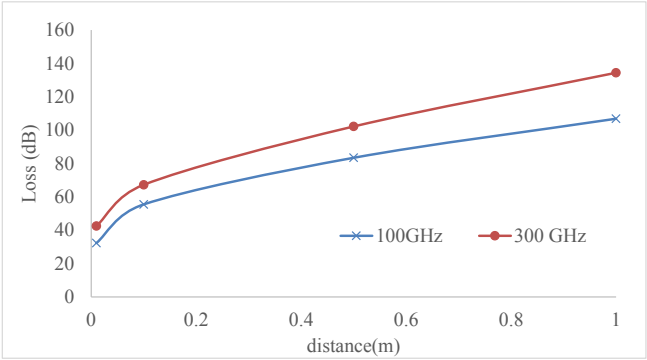


FIG. 64: Path Loss at Different Distances.

(sum of absorption and free-space loss) for various distances at 100 GHz and 300 GHz.

As can be viewed the loss would be higher than 50 dB at distances less than 0.5 m.

These calculations are used in the simulation of medical monitoring applications in Chapter 7.

VITA

Shahram Mohrehkesh
Department of Computer Science
Old Dominion University
Norfolk, VA 23529

Shahram Mohrehkesh joined at the PhD Program of Computer Science at Old Dominion University in Fall 2010. Before that, Shahram received his Bachelor Degree in Computer Engineering from AmirKabir University of Technology in August 2003. He received his Master's Degree in Computer Engineering from Iran University of Science and Technology in August 2006. Shahram's research interests include nanonetworks, big data analysis, wireless networks, mobile computing, and green computing and communication. His current research is about energy harvesting-aware solutions for nanonetworks. He has received the 2nd rank at Nokia Mobile Data Challenge at 2012. He has published over 20 publications in peer-reviewed journals and conferences related to wireless and mobile networks. He received the best paper awards in ACM MiseNet 2013 and 2014. He was selected as the outstanding graduate research assistant of the CS department in Spring 2015. Shahram is a student member of Phi Kappa Phi, ACM and IEEE. He was selected as the award recipient of Phi Kappa Phi scholarship award from the College of Sciences in April 2015.

Typeset using LaTeX.