ENERGY EFFICIENT AND SECURE WIRELESS SENSOR NETWORKS DESIGN

by

AFRAA ZUHAIR ATTIAH
B.S. Umm al-Qura University, 2007
M.S. University of Central Florida, 2012

A dissertation submitted in partial fulfilment of the requirements
for the degree of Doctor of Philosophy
in the Department of Electrical and Computer Engineering
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Fall Term
2017

Major Professor: Cliff C. Zou

ABSTRACT

Wireless Sensor Networks (WSNs) are emerging technologies that have the ability to sense, process, communicate, and transmit information to the destination, and they are expected to have significant impact on the efficiency of many applications in various fields. The resource constraint such as limited battery power, is the most challenging aspect in a WSN design as it affects the lifetime and performance of the network. An energy efficient, secure, and trustworthy system is significantly important where some information in WSNs is highly sensitive. Thus, it is critical to design energy efficient and secure mechanisms while at the same time maintaining the desired level of quality of service. Motivated by such trends, this dissertation is dedicated to exploiting optimization and game theoretic approaches/solutions to handle several important issues in WSN communication such as energy efficiency, latency, congestion, dynamic traffic load, and security. We present several novel mechanisms to improve the security and energy efficiency of WSNs. Two new schemes are proposed for the network layer stack to enhance energy efficiency through optimized sleep intervals, and, at the same time, considers the underlying dynamic traffic load. We further work on developing the routing protocol in order to handle the wasted energy, congestion, and clustering. We propose efficient routing and energy-efficient clustering algorithms based on optimization and game theory. Furthermore, we propose a dynamic game theoretic framework (i.e., hyper defense) to analyze the interactions between attacker and defender as a non-cooperative security game that considers the resource limitation. All the proposed schemes are validated by extensive experimental analyses, obtained by running simulations depicting various situations in WSNs in order to represent the real world scenarios as realistically as possible. The results show that the proposed schemes achieve high performance in different terms, such as network lifetime, compared with the state-of-the-art schemes.

Keywords– Wireless sensor networks, energy, efficient, security, game theory, routing, clustering, attack, defense, duty cycle, network lifetime, optimization, quality of service.

EXTENDED ABSTRACT

In this dissertation, we first propose two novel and efficient mechanisms for the layer stack in Wireless Sensor Networks (WSNs). First, we propose EE-MAC, an Energy-Efficient MAC protocol for distributed WSNs. EE-MAC achieves a low-duty-cycle and hence low energy consumption through optimized sleep intervals while transitioning data between sleep and active states. We consider a weighted linear combination of delay and energy saving as the performance metrics. Next, we introduce ADP, an ADaPtive energy efficient approach that meets the requirement of low energy consumption and, at the same time, considers the underlying dynamic traffic load. ADP enhances energy efficiency by dynamically adjusting sensor nodes sleep and wake-up cycles. ADP utilizes a cost function intended to strike a balance between the conflicting goals of conserving energy (waking up as rarely as possible), and minimizing sensed events reporting latency (waking up as frequently as possible). It also incorporates a feedback mechanism that constantly monitors residual energy levels, and the importance of the event to be reported, as well as predicting the next sensing event occurrence time.

The second contribution of this dissertation is the research work on developing the routing and clustering algorithms in WSNs by proposing an efficient routing algorithm and developing an energy-efficient clustering algorithm based on optimization and game theory. The proposed routing algorithm utilizes an evolutionary game theoretic approach to show how sensor nodes in a WSN could evolve their routing strategies to transmit data packets in an efficient and stable manner. The proposed equilibrium solution aims to alleviate congestion and thereby improve the network lifetime. In addition, we propose a Cost and Payment-based clustering Algorithm (CoPA) for achieving energy efficiency in wireless sensor networks under a game theoretical framework. The analysis is based on a non-cooperative, repeated general-sum game, where each node behaves selfishly in order to maximize its lifespan (payoff). We demonstrate that the correlated equilibrium is a practical solution for clusterhead selection, which provides better performance than the Nash

Equilibria. Correlated equilibrium provides a balance between the fully cooperative solution and the fully non-cooperative solution in terms of implementation overhead.

The final contribution is our proposal of a dynamic game theoretic framework to analyze the interactions between the attacker and the defender as a non-cooperative security game (i.e., hyper defense). The key idea is to model attackers/defenders to have multiple levels of attack/defense strategies that are different in terms of effectiveness, strategy costs, and attack gains/damages. Each player adjusts his strategy based on the strategy's cost, potential attack gain/damage, and effectiveness in anticipating of the opponent's strategy. We study the achievable Nash equilibrium for the attacker-defender security game where the players employ an efficient strategy according to the obtained equilibrium. Furthermore, we present case studies of three different types of WSNs attacks and put forth how our hyper defense system can successfully model them.

Through extensive simulations, the performance of the proposed schemes is validated. We observe reduced energy consumption at the cost of increased delay in EE-MAC. Simulation experiments with different traffic loads have shown that ADP improves energy efficiency while keeping latency low as well. The results also show that the proposed system of evolutionary routing game is successful in converging strategy choices to evolution stable strategy (ESS) even under dynamic network conditions. CoPA achieves better performance in terms of network lifetime and throughput compared to other popular clustering techniques. In addition, simulation results show that the proposed hyper defense approach achieves a high performance compared to two other fixed-strategy defense systems.

To my mom..

To the soul of my dad..

To my husband and kids..

To my sister and brother..

To my supportive friends, family, teachers and staff..

for always being there

for your love, encouragement and unconditional support.

ACKNOWLEDGMENTS

This dissertation would not have been possible without the guidance of my advisor and my committee, the help and support from my family and friends, and their love and encouragement. I would like to express my deepest gratitude to my academic adviser, Dr. Cliff Zou, for his continued guidance, strong support, and valuable advising throughout my Ph.D. journey. He has always been supportive of my choice of research problems and been available whenever I need his advice. Without his guidance and persistent help this dissertation would not have been possible. To him, I am eternally grateful.

I would especially like to express my deepest appreciation and eternally gratitude to Dr. Mainak Chatterjee for his important insight, strong support, and advice on numerous occasions during my study at UCF. I would also like to thank Dr. Jun Wang, Dr. Ronald DeMara, Dr. Morgan Wang, and Dr. Murat Yuksel, for their efforts in serving on my dissertation committee and providing valuable guidance and suggestions on my dissertation.

I would like to thank my lovely family and my sweet friends for providing me with the support I needed. Their love and support has always been so important for my Ph.D. study, and for my life and career. My great mother has been keeping me up through this experience by praying, taking care of me and my kids, and daily phone calls to cheer me up. I also want to thank my soul mate and my friend, Hani Shaikh Omar, my husband, for always believing in me and supporting me in following my dreams from the day we met. My wonderful brother and sister have offered unwavering love, and wholehearted support with a sense of humor. My adorable children have been patient and making my PhD study a joyful undertaking. I am also truly grateful to have known a very good friends during my years in the USA. We have been together through the best and worst of times, and they have shown me how true friends really should be.

I also would like to thank my friends, colleagues, and support staff in the University of Central Florida who have provided me with a nurturing and fruitful research environment. Thank

you for being good friends and making my Ph.D. study such an enjoyable and memorable experience. I am also appreciative to King Abdulaziz University for providing me with an opportunity and all the funding to pursue my PhD degree.

One final word of gratitude goes to someone who always waited for this moment, a person whom I always dreamt of writing these words to, a person who saw his dream in me but left too soon; my father. I know that you hear me. Thank you is not enough!

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

CHAPTER 1: INTRODUCTION

A wireless network of sensor nodes is one of the most promising contemporary technologies that unifies the physical and virtual world. A Wireless Sensor Network (WSN) is a special network composed of some autonomous small devices, called sensor nodes, scattered over regions of interest in order to monitor the physical conditions of the environments, such as temperature, pressure, sound, etc., and transmit the collected data to a central location.

WSN is one of the core next-generation application fields, including, but not limited to civil engineering, environment monitoring, medical monitoring, industrial automation, home security, military systems, and transportation. A WSN typically consists of a large number of sensor nodes, which are capable of sensing and communicating wirelessly to transmit the sensed data to the destination for future processing. Figure 1.1 presents the concept of WSNs based on a simple equation [1].

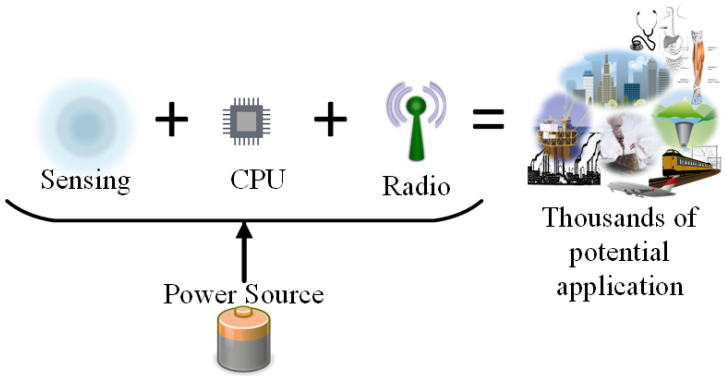


Figure 1.1: Concept of wireless sensor networks based on the equation

The resources of sensor nodes in WSNs are limited in terms of energy, computational capability, communication radius, and storage memory, where the nodes operate on battery power and are often deployed in rough environments. Due to the environmental constraints, it is usu-

ally cost-prohibitive or even impossible to replace exhausted batteries, and the networks could also become vulnerable to attacks where the data can be easily exposed during the transmission. Furthermore, the sensors generally have weak defense capabilities against the network's attacks/threats. Therefore, energy efficiency and security are critical design goals to prolong the lifetime of a WSN. Consequently, energy efficient and secure mechanisms are employed at various layers of the protocol stack to ensure longevity and trustworthiness for the nodes and the network in general.

1.1 Problem Statement and Motivation

Sensor nodes in the wireless networks are responsible for many tasks such as, sensing events, aggregating data, processing data, and sending and receiving data, where some of this data is highly sensitive. Furthermore, the large number of sensors of WSNs have unique characteristics such as autonomy, limited energy, constraint processing capability, and contested radio environment which make their tasks of sensing and communication difficult. The sensors are expected to run autonomously on their battery power for a long period. On one hand, ensuring security, availability, and confidentiality of data in WSNs has become critical. Design of sustainable WSNs becomes even more challenging in resource-constrained environments. This implies that a node must effectively utilize its resources, and increase its lifetime by closely monitoring the energy consumption and security. On the other hand, WSNs are designed for specific applications ranging from small-sized health care systems to large-scale tactical military systems, and have to satisfy a set of specific requirements that varies from one application to another. In light of this type of networking constraints, energy efficiency and security has attracted considerable research attention during the past few years [2, 3, 4, 5, 6]. However, it still requires much research effort to develop energy efficient and secure schemes of the existing algorithms in WSNs.

In a distributed sensor network, the design of the MAC protocol is particularly impor-

tant since it resolves channel contention among nodes and determines which node should access the shared channels and for how long. Quality of service provisioning poses additional challenges to the design of MAC protocols as guaranteeing delay requirements and sustaining bandwidth constraints can be compromised due to increased mutual access interference [7]. MAC protocols developed for WSNs can be broadly classified into two main categories: scheduling-based and contention-based. Each protocol is designed for specific topologies or applications [8]. Scheduling-based approaches form schedules, which allow each node in the network to access the channel and communicate with other nodes at predetermined time periods. Therefore, knowledge of the network topology is required in such kind of approaches. The schedule can be arranged according to specific approaches, such as collision avoidance or fairness among nodes. In contention-based approaches, nodes compete for the wireless medium to acquire the access for data transmission.

One of the well-known MAC protocols is S-MAC [9], where nodes periodically sleep and wake-up in order to reduce energy consumption. As events being sensed could be sporadic, sensors do not need to activate sensing function at all time. Each node turns off its radio for a certain time and then wakes up periodically to check for receptions. The listen and sleep states form a frame. Though listening time is dictated by the limitations of MAC and PHY layers, there are no such restrictions for the sleep time. Thus, the *duty cycle,* defined as the ratio of listen interval to frame duration, is small for large sleep time and vice-versa. With events being sensed sporadic, it is not necessary that the sleep duration time remains fixed. In Chapter 3 of this dissertation, we argue that the sleep duration time should be optimized depending on the frequency of sensed activities.

The design of a wireless sensor network depends specifically upon the application for which it is being deployed. Among all others, energy efficiency is regarded as one of the most critical concerns. Most of the recent studies have focused on how to maximize the lifetime of the system without sacrificing other factors, such as latency and throughput.

In wireless sensor network applications, the underlying sensing traffic load can vary dur-

ing different time of the day, and each node may have a very different participating intensity in activities based on its location in the system, e.g., whether the node is close to the sink or not. Consequently, different sensor nodes have different levels of energy consumption during sensing and communication. The level of energy consumption is one of the most important factors for sensor networks because sensor nodes usually have very limited battery capacity. Dissipation of sensor energy results in quickly diminishing the network lifetime and thus significantly affects network performance. Sensor sleeping is an effective technique to prolong a network lifetime by reducing the energy wasted in idle listening. The technique to schedule a sensor's sleep and wake-up cycle can be used in any level of the protocol stack, such as the application layer, network layer, and data link layer. In sleep mode, a node turns off the radio and goes to sleep in order to save energy instead of staying idle.

Optimization can be conducted either dynamically or statically. Both of the optimization methods assist designers in meeting the application requirements: The static optimization method remains fixed for the WSN's lifetime and is appropriate for stable environmental events. However, dynamic optimization is appropriate for changing application requirements and real environmental events; In addition, dynamic methods provide more flexibility and accuracy [10]. The current approaches to sleep and wake-up scheduling are mostly static, i.e., a node always wakes up after a fixed sleep time. This static approach is not efficient for most WSN applications since it does not consider the dynamic occurrence behaviors of most underlying sensing events. Moreover, a static approach does not consider the critical factor of the remaining energy resource of a sensor node. It is intuitively clear that when a sensor node has less and less remaining battery, it should be more cautious and conservative in waking up to report sensing data in order to prolong its lifetime. Therefore, a more energy-efficient sleep and wake-up scheduling scheme should be an intelligent and dynamic approach.

Routing in a WSN is an especially challenging task as it involves energy consumption from all the nodes that lie on a given route for a source-destination pair. Thus, designing routing

protocols in WSNs requires approaches that are able to extend the network lifetime by utilizing the sensors' limited battery as efficiently as possible [11]. Because of ever-increasing deployment of customized WSN applications, research is still being pursued that tries to improve network capabilities and to meet the quality-of-service demands for the application in question.

Congestion is one of the vital issues in WSNs since it has a significant negative impact on network performance and energy consumption [12]. While transmitting packets toward their destinations, the nodes in a WSN have multiple paths to choose from. Each path can potentially have a different associated cost as per the various routing metrics. Such variation in the cost of energy through different routes would mean that some routes/paths could be considered to be *better* than others. Therefore, nodes are expected to have a clear preference over a set of available paths. To avoid the overheads of retransmitting dropped packets due to collision, which can cause an additional drain on battery life, every node has an incentive to choose the path with the lowest cost while transmitting packets. When many nodes take this same routing strategy, this rational behavior of sensor nodes will intuitively result in further congestion on the same path and lead to energy depletion of the nodes along that path. A centralized mechanism will balance the traffic load across various paths. However, in the absence of a centralized mechanism, it is challenging to achieve long-term dynamic traffic load balance.

Clustering is a grouping technique where a network is partitioned into several clusters– each of which has a clusterhead [13]. Selection of clusterheads using energy efficient clustering algorithms in a WSN is very crucial as it affects the lifetime and performance of the network. Typically, a clusterhead is responsible for efficient communication between its cluster members and across other clusters. Typically, a cluster member would communicate with its clusterhead which in turn will communicate with other clusterheads or the base station (BS)/sink of the network. Thus, the identification of clusterheads must be done in a way that prolongs the lifetime of the entire network and improves the overall scalability of the network. Chapter 5 of this dissertation seeks to address these routing and clustering challenges.

With the adoption of newer networking technologies for better connectivity, we are witnessing an era of unprecedented networking attacks. Ensuring confidentiality, integrity, and availability (CIA) of data, devices, networks, and users has become of utmost critical importance. This becomes even more challenging in resource-constrained environments, such as wireless sensor networks (WSNs), where energy, computing, and communication resources are strictly limited. Designing defense strategies against unauthorized intrusion has been impacted by the fast of adoption of WSNs applications. Energy, computing, and communication limitations of WSNs make the security solutions very challenging. Most academic research has typically focused on a static model with a particular attack or defense on security without considering: (i) the dynamic attack intensity or the dynamic environmental conditions of the system, and (ii) the continuous interactions between the attackers and the defenders where each of them is constantly adjusting its attack/defense strategies in order to gain the upper hand. However, these two phenomena exist in almost all network security problems in the real world. Thus, besides finding a specific defense algorithm, it is equally or even more important to design a dynamic defense system that can adjust its strategies to achieve the best defense performance against intelligent attackers and under various attack situations. In Chapter 6 of this dissertation, we argue that the dynamic nature of attack intensity, network conditions, and the continuous interaction between attackers and defenders must be considered in order to operate WSNs in a secure way.

Game theory provides many effective tools to model strategic interactions between entities. Numerous areas of research have employed various concepts of a game theoretic approach involving conflict, cooperation, fairness, and competition. Game theory has been applied in different areas of wireless communication for modeling, analyzing, and predicting the rational and selfish behaviors of agents that may or may not be cooperative in nature. Nash Equilibrium (NE) is a significantly important solution concept in game theory, describing a steady state condition of the game. Among the various models of computation in game theory, evolutionary game provides a powerful modeling tool to 1) study the behavior of populations and 2) design efficient strategies in

communication networks.

Although WSNs have achieved great success in many fields, the research on this topic is still far from full-fledged in terms of both theory and application. There are many open issues in developing better energy efficient and secure schemes for designing WSNs. Therefore, it is essential to study two of the major issues in wireless sensor networks: energy efficiency and security, while considering the characteristics of the sensor networks. These characteristics give rise to numerous challenges in WSNs, which form the motivation of this dissertation. We conduct our research from four aspects in order to improve the energy efficiency and security designs for WSNs, which we will subsequently highlight. The first aspect is to design an energy efficient sensor MAC layer protocol. Designing an adaptive feedback approach for energy-efficient WSNs is the second aspect of this dissertation. The third aspect is to formulate an evolutionary game theoretic framework and an anti-coordination game for the efficiency of the routing layer, and clustering. The last is to design an energy efficient defense mechanism against several types of threats in WSNs while considering the limitation of the network resources and dynamic intensity of attacks.

1.2 Contributions

Motivated by the great potential of designing wireless sensor networks for different applications, as well as the limitations of the current research, we focus on the energy efficiency and security research in this dissertation. Specifically, this dissertation addresses the above fundamental challenges for designing wireless sensor networks, and its major contributions are the energy efficiency and secure mechanisms that are employed at various layers of the protocol stack in WSNs in order to ensure longevity and trustworthiness for the nodes, and the networks. In addition, we focus on extending the sensor nodes' lifetime and at the same time maintaining the quality of service.

In our first contribution [14], EE-MAC, an Energy Efficient MAC layer protocol with vari-

able sleep intervals for WSNs is designed. This work is motivated by the well-known MAC protocol S-MAC [9], where nodes sleep in a periodic manner to reduce energy consumption. We compute the duty cycle usage of EE-MAC and propose the selection of the sleep intervals based on a two-state Markov model [15]. We define the duty cycle as the fraction of time a node is active and that is used to define the consumed energy and the incurred delay. As for the objective function, we propose a weighted linear combination of energy and delay after normalization. The objection function is then minimized to find the optimal value of the sleep time. Through extensive simulations, we show how EE-MAC performs better compared with S-MAC in terms of energy consumption and delay.

We further propose an additional solution for the energy efficiency challenge in WSNs considering the underlying dynamic traffic load [16]. We focus on extending sensor node's lifetime by saving on energy consumption and keeping latency low. We introduce an energy efficient dynamic, and adaptive sensing scheduling approach for each sensor node wake-up/sleep time called ADP. It aims to adjust the optimal sleeping period of each sensor node adaptively according to three feedback factors: The prediction of the next occurrence time of an underlying sensing event, the sensor node's residual battery, and the importance of reporting an event by this sensor node. Control of a sensor waking up can be internal or external [17]. We follow the internally controlled wake-up policy, wherein the node periodically wakes up (duty cycling).

ADP is designed to maximize the network lifetime and save on energy consumption by optimizing the duty cycle of a node. When the frequency of the sensing traffic is high, the node should be adjusted to wake up more frequently in order to quickly report each sensing event without much latency. When the sensor node has a low battery level, its sleeping time will be adjusted to be longer in order to extend its lifetime. When the sensing event is more critical to report, the node should wake up more frequently in order to reduce the reporting latency. Our simulation experiments show that ADP could greatly extend a sensor node's lifetime compared with a well known scheduling base approach [18] without introducing much latency, which is especially

suitable for a scenario where sensing events occur with varying frequency.

Another area of focus in this dissertation has been directed towards the clustering and routing layer of the WSNs protocol stack. Though the route selection problem in a WSN is a well investigated problem, we are motivated to explore further where the objective is to alleviate energy consumption and collisions through a game theoretic framework. Game theory is a powerful mathematical tool that has been applied to numerous areas of wireless communications for analyzing and predicting the rational and selfish behaviors of various entities– the decisions of which determine the outcome of the game [19].

In this dissertation, we leverage concepts from evolutionary game theory and model the routing decisions in a WSN as a non-cooperative evolutionary game [20]. We prove that the mixed strategy Nash Equilibrium (NE) in our routing game is the evolutionary stable strategy (ESS); where there are no other strategies except this ESS that can dominate the population. The payoff for every node, also referred to as a player, is determined by the packet transmitting cost, which depends on the distance between the nodes. In the routing game, choosing the shortest distance between the source and the next neighbor hop is preferable for each player because it will consume the least amount of energy for the transmission, thereby increasing the payoff. The players who transmit the packets through the shortest path will gain a higher payoff (lower cost) compared with the players who transmit through longer paths. However, if every player tries to select the shortest path to the target, it will result in collisions and lead to energy depletion at the nodes. Thus, forwarding the packet through the lowest energy path may not always be optimal for network lifetime.

To model the adaptation of the hop selection strategies and to show the behavior of the system over a period of time, we present the replicator dynamics of our game. We study how the sensor nodes improve their strategy selection over time until they converge to an evolutionary stable strategy. Furthermore, once the strategies converge to ESS, the population cannot be invaded by any other populations of the nodes, and the system will reach stability. The process of selecting

the path of transmission for the packets in our routing game continues until the destination node is reached. The objective of the game is to reduce the load and avoid collisions on the most used routes by distributing the data transmission task on all possible routes.

Furthermore, we take a game theoretic approach to devise a clustering algorithm for WSNs. In our approach, the nodes are the players who play the clustering game. We propose a Cost and Payment-based clustering Algorithm (CoPA) where we formalize the profits and losses for each node. CoPA has the provision to alternate the responsibility of a clusterhead among the nodes, thereby balancing energy using a weighted metric that combines the transmission power and energy of each node. An anti-coordination clustering game is formulated for 2 players as well as N players using only local information. We derive the correlated equilibrium (CE) for the clustering game by solving the linear optimization. An adaptive regret matching (no-regret) algorithm is used to guarantee convergence of the probability distribution to the CE. Moreover, we prove and discuss the optimality of CE solution for the clustering game, and compare it to the pure and mixed strategy Nash Equilibrium (MSNE) solutions in terms of the efficiency and fairness among the nodes. We also evaluate the performance of our clustering algorithm with two popular clustering techniques, and demonstrate that CoPA has superior performance in terms of network lifetime and system throughput.

Finally, we design a network-warfare framework, rooted in game theory, which considers dynamic interactions and evolutions between attackers and defenders. We introduce a novel approach for a defense mechanism against several types of attacks/threats on WSNs– a hyper defense mechanism that considers the limitation of the resources as well as the security value of the asset of the network. Our model provides suitable responses for a defender by considering different intensities of attacks and the relative cost to launch them. We model the interactions between the attackers and defenders as a network-warfare game, as it has proven to be a highly efficient mathematical method for analyzing and modeling scenarios with conflicting objectives. Furthermore, in order to control future threats in security systems, game theory is useful in suggesting various

probable actions and in predicting their related outcomes. We present *a non-cooperative zero-sum attacker-defender* game. We formulate the security game between an attacker and defender to study the dynamic interactions between rational players with conflicting interests. In addition, we attain optimal strategies for the defender and the attacker considering that they can dynamically choose their strategies in order to maximize their own payoff based on cost minimization. Generally speaking, we classify the actions of either attacking or defending into three categories: *level zero, level one, and level two*. The attacker can alternate between these three strategies, where level zero represents no attack, level one represents a low intensity of attack, and level two represents a high intensity attack. Likewise, we classify the defender's actions into three corresponding defense levels. For level zero, the defender decides to not defend at all. The second one is a low level of defense, which could cost some of the resources (i.e., energy, or memory space, etc.). The third one is a high level of defense, which requires more computational, battery power, or memory, but gains strong countermeasures against the threats. In practice, the strategies of attackers and defenders for any network security problems could be categorized into more fine-grained levels, but for the sake of clarity and modeling purposes, we believe such a three-level classification of attack or defense is generalized enough and can well represent attack and defense activities in real practice. Simulation results show that the proposed system achieves a high performance compared to two other fixed-strategy defense systems.

1.3 Dissertation Organization

The dissertation is organized as follows: In Chapter 2, we survey and discuss the significant literature review related to this dissertation. In Chapter 3, we present an energy efficient medium access control protocol for distributed wireless sensor networks. Chapter 4 presents an dynamic and adaptive energy efficient approach for sensor networks. Two an energy-efficient routing, and clustering algorithms based on game theory in WSNs are proposed in Chapter 5. In Chapter 6, we

present a game theoretic approach to model WSNs attack and defense strategies. Finally, Chapter 7 presents the simulation models and results. Conclusions and future works are drawn in Chapter 8.

CHAPTER 2: LITERATURE REVIEW

In this chapter, a review of the relevant literature is surveyed to discuss and analyze the research that has been done in this area, and determine the status of our current research within the large paradigm of wireless sensor networks communication. We introduce the literature pertaining to the energy efficient design in medium access control layer, the feedback approach design in any layers of the protocol stacks by utilizing sleeping technique, the energy efficient design in the routing layer, and secure design in WSNs under game theoretic frameworks.

2.1 Medium Access Control (MAC) in WSNs

There is a rich literature on energy efficient MAC protocols in WSNs [21]. The proposed protocols focus on reducing all sources of wasted energy such as idle listing or overhearing. The collisions also waste energy due to extra transmissions to handle the discarded packets. Control packet overhead can consume extra energy by the unnecessary transition unless designed according to the network requirements.

Ye et al. [9] proposed S-MAC, a contention-based MAC protocol for WSNs. S-MAC establishes low-duty-cycle operation to reduce energy consumption on the sensor nodes by periodically putting nodes into sleep and active states. Nodes coordinate their sleep schedules rather than having random sleep periods. Each node chooses a schedule and exchanges it with its neighbors before starting its low-duty-cycle operation. The node select its time schedule randomly if it does not hear any a schedule from another node. Then, the node broadcasts its schedule in a SYNC message and the node receiving the schedule sets up the same schedule. T-MAC by van Dam and Koe [22] performs better than S-MAC in terms of traffic load. The active period in T-MAC ends when no activation occurs for a certain time. This can be advantageous for energy consumption but it affects the channel throughput [8]. ADC-SMAC by Hu et al. [23] is another improved version of

S-MAC, designed specifically for the chain and cross topologies. ADC-SMAC adjusts duty cycle dynamically based on average sleeping delay, the upper and lower bound cycles.

QA-MAC by Gao [24], which is also based on S-MAC protocol, improves energy efficiency by coordinating the contention window dynamically. AsyMAC by Wang et al. [25, 26] is designed for wireless networks with asymmetric links. AsyMAC uses a set of concepts and metrics characterizing the ability of MAC to silence nodes which could cause collisions. Adaptive Coordinated Medium Access Control (AC-MAC) protocol proposed by Ai et al. [27] is a contention-based MAC protocol for WSNs. AC-MAC introduces adaptive duty cycle technique that depends on the different loads of traffic and provides optimized trade-off strategies for energy, throughput and latency.

DCMAC by Zheng et al. [28] uses a dynamic duty cycle with dynamic sleeping intervals and a fixed listening interval. DCMAC reduces energy consumption and the latency by utilizing the cooperation of the dual channel and the selection strategy for candidate nodes. Cho and Bahk [29] uses a multi-hop data packet in a single duty cycle in Hop Extended MAC (HE-MAC) to set up the path for multi-hop transmission. This approach also utilizes a state to extend the relay of the packet beyond the start of the sleep period. Multi-token based MAC protocol with sleep scheduling for WSNs [30] by Dash et al. aims to improve energy efficiency along with faster data transmission, data aggregation, data accuracy and low latency in hop-by-hop delivery. The limitation of this protocol is the high latency for finding a new neighbor. E-BMA by Shafiullah et al. [31] is proposed to achieve energy efficiency for wireless data communication networks with low and medium traffic. Sender-centric MAC (SC-MAC) by Liu et al. [32] is an asynchronous duty cycling MAC protocol designed for bursty traffic loads. SC-MAC provides a collision free environment without additional overhead. A latency optimization mechanism is also introduced by SC-MAC for multi-hop networks.

Liu and Yao [33] propose An Appointment Based MAC Protocol (AB-MAC), which also improves the asynchronous duty cycle and overcomes the effect of channel contention of multiple

senders. AB-MAC utilizes a fusion appointment scheme to enable scheduled batch transmission for multiple senders with low overhead to improve the transmission efficiency, latency and energy efficiency in many-to-one traffic pattern.

2.2 Sleeping Techniques in any Layers of the Protocol Stack

Most wireless sensor networks' protocols have been based on application requirements. Recently, researchers have been using sleeping techniques for reducing energy consumption in all layers of the protocol stack in wireless sensor networks [17]. Previous works have shown a broad range of the use of sleeping techniques in different categories. The sleeping techniques can be divided into scheduled wake-up, radio controlled wake-up, and environmentally controlled wake-up. Scheduled wake-up is divided based on time synchronization, where it could be synchronous or asynchronous duty cycling [17] [34].

ER-MAC [35] is a TDMA based MAC protocol that selects the sleep and wake schedules based on a node's criticality by letting the more critical nodes sleep longer. The sleeping techniques can also save energy in routing protocols as some studies showed [17]. [36] proposed a sleeping multipath routing approach that can be applied to any routing protocol by selecting the minimum numbers of disjoint paths to meet the reliability demands and by turning off the rest of the sensor nodes. GTC (Geographical Topology Control protocol) [37] extends the network lifetime by dividing the network into zones and selecting one active node in each zone.

Sensors have two major operations: sensing and forwarding data [38]. In our dissertation, we focus on producing an energy-efficient way to sense an event based on the feedback. Other researches, such as PW-MAC [39], focus on the forwarding and transmission of sensed data. PW-MAC [39] is an energy-efficient predictive wakeup MAC protocol that enables senders to accurately predict receivers wakeup times. The protocol minimized idle listing and overhearing by enabling a sender to rendezvous with a receiver quickly according to the predicted receiver wake-

up. It could be beneficial to combine PW-MAC technique and our proposed approach together to have a complete energy efficient scheduling system.

2.3 Routing and Clustering in WSNs under Game Theoretic Framework

In this section, some of the literature pertaining to the routing and clustering in WSNs is provided as well as applications of some game theoretic solution concepts in the context of communication networks.

2.3.1 Optimal Route in WSNs

Finding optimal routes is one of the most interesting research topics in communication networks. Various research tools have been proposed to investigate this issue, including game theory. Game theoretic technique have been applied to numerous areas of wireless communication for analyzing and predicting the rational behaviors of agents that have also proven very useful in the design of wireless sensor networks [40]- [41]. Important and essential issues in WSNs, including routing protocol design, energy saving, packet forwarding, security, and other sensor management tasks, have been modeled and described by the game theoretic approaches for efficient solutions that maximize the network lifetime. In one of our publication work [20], we provide a game theoretic model with utility functions considering forwarding and routing in the presence of adversaries.

The pricing and payment model is presented as a cooperative game in [42]. The goal of the game is to find an optimal path in a WSN by considering reliability, energy, and traffic load, where the nodes have incentives to cooperate in the game. Buttyan and Hubaux [43] proposed Nuglets, which is virtual currency in the system, to stimulate the cooperation of the nodes participating in forwarding packets in mobile ad hoc networks. Furthermore, a reliable length-energy constrained routing scheme in WSNs has been presented in [44], where a game-theoretic approach is utilized.

In this approach, the sensors cooperate as rational agents in order to find the optimal route and maximize their payoffs in the game. Two different possible payoff models and utility functions were illustrated.

The issue of energy efficiency in WSNs has been addressed in [45]. It provided a game theoretic adaptive algorithm in order to manage sensor behavior for achieving complete decentralized control in an energy-constrained sensor network. Evolutionary game theory has emerged as a robust tool to investigate and solve dynamic networking issues. An evolutionary game theory was applied in [46] where the authors proposed a three-dimensional game theoretic energy balance (3D-GTEB) routing protocol to enhance the routing decisions and to decrease the overhead in a WSN. They addressed the unbalanced energy consumption problem by applying evolutionary and classical game theory at two levels of game theoretic decision making. The two levels were called wedge level energy balance and node level energy balance. In this dissertation, we formulated this routing problem by utilizing an evolutionary game to study the behavior of the population and induce the equilibrium even under dynamic wireless sensor network conditions.

In [47], a joint duty cycle scheduling and energy aware routing approach (DREG) is presented based on evolutionary game theory. The solution for this game is proposed as evolutionary equilibrium. The authors aimed to prolong the network lifetime in WSNs by finding an optimal wakeup/sleep scheduling policy, based on a trade-off between network throughput and energy efficiency for each sensor. The issues of duty cycle scheduling and energy conservation are modeled as a multi-agent non-cooperative game, and the game is repeated until a steady state is reached. Authors of [48] have also applied the evolutionary game theory to solve the routing problem in a general network topology. The authors consider the link costs that are linear in the link flow.

Furthermore, authors of [49] model the evolutionary game to study the dynamic cooperative behavior of selfish nodes under AODV routing. In the game, packet-forwarding is repeated, and includes two distinct modes, in order to learn and predict the neighbors' node behavior to improve network performance. The first mode is deterministic to analyze the behavior of the network

for standard strategic patterns. Random mode is the second one that applies a genetic algorithm to predict the best strategy randomly. Proposed in [50] is an adaptive and distributed routing algorithm for correlated data that gathers and exploits the data correlation between nodes based on a game theoretic framework.Specifically, the issue of effective energy minimization is addressed and a routing solution is presented. The energy metric, interference awareness and opportunity for multi-hop partial data aggregation are considered. The authors formulate the game by incorporating a general multi-hop data aggregation model into the problem definition to describe data reduction in a congestion game.

A reliable delivery routing issue in WSNs is addressed in [51] through the game theoretic framework. The authors aim to ensure stable cooperation among nodes for delivering the packet and minimizing the routing cost as well. The proposed reliable coalition formation routing protocol (RCFR) is presented using a coalitional game theory, which selects the route according to the principle of lowest cost. In order to introduce a fair allocation method for payoff division, a characteristic function is designed by leveraging performance metrics. RCFR protocol is elaborated by extending the AODV protocol, where the path with minimum cost will be selected to transmit packets, and route maintenance is achieved by adding route residual energy ratio monitoring.

In our dissertation, we provides a game theoretic model, with utility functions, considering forwarding and routing. We leverage concepts from evolutionary game theory and model the routing decisions in a WSN as an anti-coordination evolutionary game. We provide detailed analysis of the system stability and fairness of the solution as well. The payoff for every node, also referred to as a player, is determined by the packet transmitting cost, which depends on the distance between the nodes. We study the behavior of the population and induce the equilibrium even under dynamic network conditions.

2.3.2 Clustering in WSNs

Clustering in WSNs is an interesting topic especially when it is studied under the game theoretic framework. Various clustering algorithms have been proposed such as the well-known LEACH [52] where the mechanism of selecting a clusterhead is to ensure rotation of the roles between the nodes in a probabilistic manner. Weighted Clustering Algorithm (WCA) [53] is another clustering scheme that considers several metrics such as ideal degree, transmission power, mobility, and battery power of the nodes. The algorithm can be optimized according to particular needs of the applications and adapt itself with changing topology of the network. In addition, the algorithm distributes the load as much as possible between the nodes, and it executed only on-demand, instead of periodically.

In [54], the authors proposed a an energy-efficient Adaptive Clustering Hierarchy routing algorithm based on game theory. The clusterhead selection is centralized and decided by the sink based on the locations and remaining energy of the nodes. The authors show that the algorithm is suitable for the statically distributed WSNs and more energy-efficient than a random one. However, no theoretical analysis has been provided beside the centralized selection mechanism that could lead to higher energy consumption.

In CROSS [13] (Clustered Routing for Selfish Sensors), each sensor behaves selfishly in a non-cooperative manner in order to conserve its energy. The authors provided the pure and mixed strategy NE and the related expected payoffs of the games. The possibility clusterhead absence could occur continuously because of the dependency on selecting the clusterheads based on each node's probability. In [55], the authors proposed a clustering algorithm based on game theory for energy efficiency in WSNs. The probability that a node serves as a clusterhead depends on the energy model. Furthermore, game theory based energy efficient CH selection approach is proposed in [56] based on the Subgame Perfect NE (SPNE). The clusterheads are selected based on SPNE decision.

In our dissertation, we attempt to provide a new solution from a game theory prescriptive for clustering in WSNs where we study the correlated equilibrium and its properties. The CE achieves strictly better performance compared to the NE and therefore maximizes the network lifetime and throughput [57].

2.4 Security in WSNs under Game Theoretic Framework

Security under a game theoretic framework is an interesting topic, where several probable actions along with the predicted outcome can be suggested through game theoretic methods in order to control future threats. Game theory is suitable for modeling various issues and have been successfully used in cyber security including communication networks [58] [6] [59]. Various issues in security and privacy in networking and mobile application have been addressed and modeled through game theoretic framework [60].

In [61], the author addresses the issue of defending against denial-of-service attacks in the network and proposes a puzzle-based defense solution that can be distributed or non-distributed using the concept of Nash equilibrium. A non-distributed DoS attack and the puzzle-based defense were modeled as a two-player infinitely repeated game of discounted payoffs and the optimal defense strategy that would be gained for the service provider. The defense strategy is determined by the difficulty of the puzzle level. A distributed DoS attack is considered as two-player stochastic game as well, and the solution is based on the non-distributed DoS solution.

The authors of [62] propose a Bayesian game approach for intrusion detection in wireless ad hoc networks to analyze the interactions between pairs of attacking and defending nodes. The concept of Nash equilibrium is utilized in both static and dynamic scenarios. A player can be either a malicious or regular node. A defending node can chose to a monitor or not monitor, whereas a malicious node can employ the attack or not attack strategy. Two methods are proposed in order to reduce the resource consumption. The first method is adopting a probability of defending when

there is a sign of attack. The second one is employing two different monitors (i.e., lightweight and heavyweight monitors).

In [63], a secure routing protocol is proposed by modeling the interaction of nodes in WSN and intrusion detection system as a Bayesian game formulation. In the game, at least one player has incomplete information about the other players, and each node player a reputation score. The selfish nodes can cooperate by sending the packets (to avoid detection) or dropping packets, and malicious nodes would then be eliminated from the network. Therefore, the nodes are motivated to act rationally and gain the score of reputation through this approach. Two Bayesian Nash equilibriums are provided to detect selfish nodes or to force them to cooperate.

Unlike most security mechanisms that focus on a particular attack or defense, we provide in this dissertation a dynamic defense system that considers the variation in the intensity of attack and defense.

CHAPTER 3: EE-MAC: AN ENERGY EFFICIENT SENSOR MAC LAYER PROTOCOL

3.1 Overview

Energy efficiency is of utmost importance for wireless sensor networks deployed without any possibility of battery replenishment. In this chapter, we propose an Energy Efficient Medium Access Control protocol (EE-MAC) for wireless sensor networks, which achieve a low-duty-cycle and low energy consumption through optimized sleep intervals based on a 2-state Markov model. The duty cycle is defined as the fraction of time a node is active. Energy consumption, and the incurred delay when the node switch between sleep and active states, are defined as well. We formulate a weighted linear combination in order to find the optimal sleep time for each node.

The rest of the chapter is structured as follows: Section 3.2 describes details of the proposed EE-MAC protocol, and section 3.3 provides a summary of the chapter.

3.2 EE-MAC Protocol

The main goal of EE-MAC is to reduce energy consumption and to optimize delay performance. This goal is achieved by determining the optimal value of the sleep interval based on prevailing conditions. EE-MAC changes the state of each node between sleep and active. During the sleep state, a node's radio is turned off which decreases the power consumption. During the active state, the node wakes up and listens, receives, or transmits data. As a result, the consumed power increases.

3.2.1 State Model

The node activities in EE-MAC can be represented using the Gilbert-Elliott model [15] [64]. This 2-state Markov model is shown in Fig. 3.1 where transitions from 'sleep' state to 'active' state

occurs with probability P_{sa}. Similarly, transitions from 'active' state to 'sleep' state occurs with probability P_{as}. Transitions from each state to itself is also shown. Thus, the probability of being in the active state is:

$$P_a = P_{sa} + P_{aa}.$$

Similarly, the probability of being in the sleep state is:

$$P_s = P_{as} + P_{ss}.$$

It is to be noted that we do not treat receiving, transmitting, and listening as different states as they are included in the 'active' state.

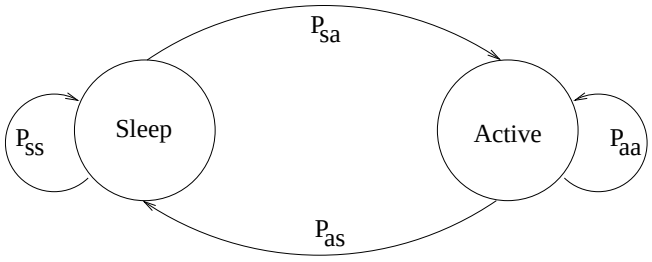


Figure 3.1: 2-state (active and sleep) Markov model.

It is assumed that the active times and sleep times are exponentially distributed. Let us define $\overline{t_a}$ as the average time a node spends in the active state. Similarly, $\overline{t_s}$ is the average time a node spends in the sleep state. Thus, we can define the duty cycle of the node as:

$$\rho = \frac{\bar{t}_a}{\bar{t}_a + \bar{t}_s}$$

i.e., the fraction of time the node is active. It can be noted that, $P_a = \rho$ and $P_s = (1 - \rho)$.

3.2.2 Energy and Delay

Although it is desirable to have a low duty cycle, it compromises the delay performance. For instance, if a node sleeps while there is data transmission to it, the node will incur some delay in its response, which increases as the sleep times become longer. Thus, while optimizing the sleep intervals, the deterioration in the delay response must be taken into account. Given different energy consumptions of two states, we define the total energy consumed per unit time per node, E, as follows:

$$E = E_{Active} + E_{Sleep} \tag{3.1}$$

where E_{Active} is the average energy consumed per unit time in active state and E_{Sleep} is the average energy consumed per unit time in sleep state. If W_a and W_s are the energy consumed per unit time during the active and sleep states respectively, then $E_{Active} = \rho W_a$ and $E_{Sleep} = (1-\rho)W_s$. Thus, the total consumed energy is defined as follows:

$$E = \rho W_a + (1-\rho)W_s \tag{3.2}$$

For a sleeping node, the expected time to wake is $\bar{t}_s$, irrespective of the time it has been sleeping. This is a result of the assumption of exponential sleep time distribution, hence memoryless. Thus, delay can be defined as $D = \bar{t}_s$.

3.2.3 Normalization of Energy and Delay

To include both energy E and delay D in a combined metric, we must normalize them in a way so that they map to a number between 0 and 1. If we assume $max(W_a, W_s) = W_a$ as energy spent in active mode is more than the energy spent in the sleep mode, then the maximum value for E is W_a. This happens when $\rho = 1$, i.e., the node is always in the active state. Thus, we define the

normalized energy, E_{norm}, as:

$$E_{norm} = \frac{\rho W_a + (1-\rho) W_s}{W_a} \tag{3.3}$$

Similarly, we seek a function for D such that when $t_s \to 0$, $D \to 0$ and when $t_s \to \infty$, $D \to 1$. We define the normalized delay, D_{norm}, as $D_{norm} = 1 - \frac{1}{t_s}$.

3.2.4 Combined Metric

We define the combined metric as a linear combination of E_{norm} and D_{norm} as:

$$U = w_1 \times E_{norm} + w_2 \times D_{norm} \tag{3.4}$$

where w_1 and w_2 are the corresponding weighing factors and $w_1 + w_2 = 1$.

We seek to find the value of t_s for which U will be minimized. Thus, we take partial derivatives and equate to 0. Thus,

$$\left[\frac{\partial U}{\partial t_s}\right] = \left[\frac{\partial E_{norm}}{\partial t_s}\right] + \left[\frac{\partial D_{norm}}{\partial t_s}\right] = 0 \tag{3.5}$$

Solving equation (3.5), we get

$$\bar{t}_s = \sqrt{\frac{w_1 W_a \bar{t}_a - w_2 W_a}{W_s w_1 \bar{t}_a}} \tag{3.6}$$

For $\bar{t}_s$ to have a real value, $w_1\, \bar{t}_a \geq w_2$.

3.3 Summary

Achieving energy efficiency in WSNs is of utmost importance. Since sensor nodes consume more power while sensing and transmitting compared to idle time, achieving a low duty

cycle improves the performance in terms of energy consumption. We achieve this goal by putting nodes to sleep at the cost of degraded delay performance. We presented energy efficient medium access control layer protocol, Energy Efficient MAC layer protocol, called EE-MAC. We derived the energy consumption and the incurred delay when the node switches between the two states (i.e.,sleep and an active state). We also proposed a combined metric which is a linear sum of the two and find the optimal sleep time.

CHAPTER 4: ADP: AN ADAPTIVE FEEDBACK APPROACH FOR ENERGY-EFFICIENT WSNs

4.1 Overview

Numerous design of WSNs has been considered in order to satisfy the requirements of real-world applications. One of the most common approaches for energy conservation is to alternate each sensor node between sleep and wake-up states in order to address the challenges aspect of protocols design: limited battery power in the sensor nodes. In this chapter, ADP is proposed as ADaPtive energy efficient approach that meets the requirement of low energy consumption and, at the same time, considers the underlying dynamic traffic load. ADP enhances energy efficiency by dynamically adjusting sensor nodes' sleep and wake-up cycles. ADP thereby utilizing a cost function to strike a balance between the conflicting goals of conserving energy (waking up as rarely as possible) and minimizing sensed events' reporting latency (waking up as frequently as possible), simultaneously. It also constantly monitors and provides feedback concerning the residual energy level and the importance of the event to be reported, as well as predicting the next sensing event occurrence time.

The rest of this chapter is structured as follows: Section 4.2 highlights the motivation for the proposed idea. In section 4.3, we present the ADP approach. Section 4.4 is a further detailed discussion. Finally, we summarize the chapter in section 4.5.

4.2 Motivation for the Proposed Idea

In all layers of the protocol stack in wireless sensor networks, sleeping techniques for reducing energy consumption have been used by researchers to satisfy the application requirements of the protocols design [17]. Sensors have two major operations: sensing and forwarding data [38]. In this chapter, we focus on producing an adaptive and energy-efficient scheduling approach for

sensors to sense and report events. It can be readily combined with many previous developed systems that focus on energy-efficient data forwarding in order to have a completely energy efficient scheduling system that covers both data sensing and data forwarding operations of sensors. Compared with a well-known scheduling base approach [18], our simulation experiments show that ADP can extend a sensor node's without introducing much latency, an especially suitable for a scenario since sensing events can occur with varying frequency. Our approach can be used on different scenarios of underlying sensing events. Other research studies, such as [39], has focused on the forwarding and transmission of sensed data. Part of our research puts forth a new dynamic and adaptive scheduling approach that aims to adjust the optimal sleeping period of each sensor node adaptively according to the following: the importance of reporting an event for this sensor node, the prediction of the next occurrence time of an underlying sensing event, and the sensor node's residual battery. The ADaPtive energy approach (ADP) is designed to maximize network lifetime and save on energy consumption by optimizing the duty cycle of the node. The node should be adjusted to wake-up more frequently when it senses traffic is high in order to quickly report each sensing event without much latency. In order to extend its lifetime, the sensors' sleeping time will be adjusted when the sensor node has a low battery level. Conversely, the node should wake-up more frequently when the sensing event is more critical, thereby reducing the reporting latency.

4.3 Proposed ADP Approach

4.3.1 Wake-up Technique

Waking a node up and putting it to sleep periodically instead of keeping the node awake all the time saves significant amount of energy. A periodical scheduling technique could be synchronized, where all the nodes will adjust to the periodic wake-up time synchronously. On the other hand, the scheduling technique could be asynchronous, where each node's wake-up time does not require any synchronization, and each node can adjust its own periodic wake-up time indepen-

dently [65]. Some existing approaches [18] use a base approach of wake-up technique that gives a node a fixed period of sleeping time throughout the node's lifetime. In the base approach, the node wakes up after a fixed amount of time, which is not suitable with dynamic changing sensing events. As an example application, sensors for monitoring a bridge condition may have very dynamically changing sensing activities to monitor and report throughout a day. During rush hours, sensor nodes will be busy and need to be awake more frequently to report sensing events than during nighttime, when vehicular traffic over the bridge is dramatically decreased. Clearly, fixed wake-up time scheduling depletes sensor nodes an unnecessary high amount of energy at night, and at the same time, sensor nodes may not wake up quickly enough during rush hours in order to sense and report events on time. In contrast, our approach adapts the node waking up scheduling based on the occurrence frequency of environmental events.

Although ADP runs on each sensor node independently, if some sensor nodes have exactly the same settings and observe the same sequence of events, executing ADP on these sensors will enable them to have identical sleep/wake-up scheduling, i.e., they are in synchronous mode. On the other hand, two sensors are in asynchronous mode if they have different settings or observe different events. Therefore, we can say that ADP is a hybrid approach by combining synchronous and asynchronous modes.

4.3.2 Criticality of Sensor Node

Unlike existing methods, where all nodes are treated equally all the time, we treat each node in ADP according to its own conditions (which we call criticality), and adapt its sleep/wake-up duty cycle with underlying sensing traffic density. We measure the criticality of a sensor node by the following two parameters:

- **Residual energy of a sensor node:** each node has its own residual energy level, and it varies according to the node activity and past energy consumption during its lifetime.

- **Importance of reporting data:** due to the application requirements, types of sensing data, and node locations, each node could have different values measuring the importance of reporting events it needs to sense and report.

4.3.3 Sensing Event Modeling and Prediction

The static behavior of the traditional system under varying sensing event density may increase the energy wastage and could reduce the efficiency of a sensor network. The main idea of ADP is to adjust the optimal sleeping time for each node and to adapt the network sensor node to be appropriate with an environmental dynamic-changing traffic load. In ADP, a sensor node will stay awake for a certain amount of time. If there is no event to report, it will go back to sleep immediately; if there is an existing event to report, it will report the data and then go back to sleep. Its next wake-up time is determined according to the node's criticality and the prediction of the next event arrival time.

We assume that the underlying sensing events follow the Poisson Distribution with the dynamically changing rate λ_t at time t. We estimate λ_t in each sleep/wake-up cycle based on previous observations of event arrivals (i.e., the T_i sequence) and the prior estimated value of λ_t (denoted by $\overline{\lambda_{t'}}$). We apply the idea of estimating the new arrival rate via a low-pass filter [66].

$$\overline{\lambda_t} = (1-\alpha)\lambda_i + \alpha\overline{\lambda_{t'}} \tag{4.1}$$

where λ_i is the Poisson arrival rate based on the most recent arrival event, $\overline{\lambda_t}$ is estimated arrival rate, and α is a filter gain coefficient to adjust how smooth we want the estimated $\overline{\lambda_t}$.

Here we explain how we obtain λ_i based on the most recent observation. Figure 4.1 illustrates sensing events occurrence over time. We denote T_i as the inter-arrival time between the i-th event and the previous $(i-1)$-th event. Since we assume that sensing events follow the Poisson arrival process, and sensor nodes know when each previous event happened, we use the observed

most recent inter-arrival time T_i to estimate the current time Poisson process rate λ_i, and, hence, we set the value of λ_i as:$\lambda_i = \frac{1}{T_i}$.

For reader's convenience, we list the main mathematical notations used in this paper in Table 4.1.

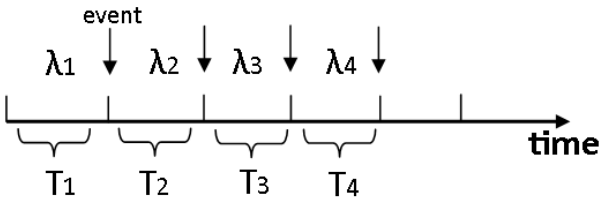


Figure 4.1: Illustration of sensing events arrival. T_i is the inter-arrival time between the i-th event and the previous $(i-1)$-th event; λ_i is the estimated Poisson arrival rate based on T_i where $\lambda_i = 1/T_i$.

4.3.4 Feedback Optimization

4.3.4.1 Feedback Optimization Model for General Distribution

Our proposed approach is based on optimizing a cost function with the goal of minimizing the cost of energy consumed while matching with traffic density and maintaining an acceptable latency. The optimization tries to achieve a balanced trade-off between the energy consumption and sensing data report latency. There are two types of cost that we consider in our formula:

- The cost of energy wastage when the node wakes up without any sensing event happening during its previous sleep period.
- The cost of sensing data report latency when the node is sleeping during the occurrence of an event, thus introducing a time delay when it wakes up and reports the event.

The first case happens when the occurrence frequency of underlying sensing events is low and the node wakes up too often. The node will consume an undesired amount of energy in that awake time without reporting any events. In the second case, the cost of latency becomes high when the occurrence frequency of underlying sensing events is high, and the node wakes up less frequently. In this case, the sensor node sleeps longer than desired, whereas there are some events the node needs to report more responsively. Let random variable X denote the inter-arrival time between sensing events. We define the general formula of the combined cost function as:

$$f(t_s) = w_1 r P + w_2 c d_{t_s} Q \tag{4.2}$$

where r represents the criticality of remaining battery of a sensor node, c represents the importance of a sensed event, and t_s is sleeping time. The average latency is represented by d_{t_s}. $P = Prob.(X > t_s)$ is the probability of wasting energy when waking up in the absence of a sensing event (first case); $Q = Prob.(X \leq t_s)$ is the probability of finding an event occurrence during the prior sleep period (second case).

w_1 and w_2 are weight factors that should be set up by the network operator to achieve a balance between energy saving and data report latency. The cost function shows that the absolute values of w_1 and w_2 do not matter; what matters is the relative values of these two weight factors. Thus we can let:

$$w_1 + w_2 = 1 \tag{4.3}$$

In order to find the optimal sleeping time t_s^* based on the cost function (4.2), we just need to take partial derivative of the cost function against t_s and set it equal to zero, as $\left.\frac{\partial f}{\partial t_s}\right|_{t_s^*} = 0$.

4.3.4.2 Feedback Optimization Model based on Poisson Distribution

The above optimization model based on general distribution is theoretical and abstract. In order to illustrate how we can utilize this feedback optimization model in many sensor network

applications, in this section we describe the traffic arrival process as a Poisson distribution and explain how to use the feedback optimization model to improve energy efficiency in a concrete way.

Table 4.1: List of Notations

Notation	Definition
λ_t	The dynamic Poisson arrival rate for sensing events at time t
$\overline{\lambda_t}$	Estimated Poisson arrival rate at time t for sensing events
T_i	Inter-arrival time between the $(i-1)$-th event and i-th event
t_s	Sleeping time
ξ	Remaining battery of sensor node
r	Critical factor of remaining battery of sensor node, $r = \frac{1}{\xi}$
c	Factor of importance of reporting sensed event
P	Prob. of wasting energy when a sensor node wakes up without any event to report
Q	Prob. of finding event occurred during the sensor node's prior sleep period
d_{t_s}	Average sensing data report latency
w_1, w_2	Weight factors in cost function, where $w_1 + w_2 = 1$
t_s^*	Optimal sleeping time

Poisson distribution is the most suitable distribution for the majority of sensor network applications. If there exist a large number of entities each of which has a very small probability to independently generate sensing events, then such event occurrence can be modeled accurately by a Poisson distribution. One example of such an application is in using sensors to monitor the condition of a bridge and the traffic flowing over it. There could be millions of vehicles in the local area of the bridge, but the probability of any one vehicle going over the bridge at a specific time is very small. A similar instance can be found in sensors monitoring wildlife, where the population of wildlife is large, but the probability of a specific animal appears in the specific area for the sensor to detect is small.

As we described above, X represents the inter-arrival time between sensing events. Since we assume the sensing event occurrence follows the Poisson process with a dynamically changing

rate λ, this random variable X follows exponential distribution with the same rate λ. $(X > t_s)$ denotes the absence of a sensing event during the time interval $[0, t_s]$. The probability of absence of sensing event when waking up is $Prob.(X > t_s)$, which is given by the following formula:

$$P = Prob.(X > t_s) = e^{-\lambda t_s} \tag{4.4}$$

Similarly, the probability of event occurrence during the sleep time interval $[0, t_s]$ is:

$$Q = Prob.(X \leq t_s) = 1 - e^{-\lambda t_s} \tag{4.5}$$

Because of the following Poisson process Theorem: "Given that $N(t = n)$, then those n arrival times $S_1, ..., S_n$ have the same distribution as the order statistics corresponding to n independent random variables uniformly distributed on the time interval $(0, t)$" [67], we define the average latency d_{t_s} in our cost function (4.2) as half of the sleeping time $d_{t_s} = \frac{t_s}{2}$.

In addition, we define the critical factor of remaining battery of a sensor node as $r = \frac{1}{\xi}$, where ξ is the fraction of remaining battery energy as compared with the battery's full capacity. The importance of sensed events parameter c is specified manually by the operator for each sensor node according to its location and sensing data type.

After deriving the formulas for all the variables, the cost function becomes:

$$f(t_s) = [w_1 \frac{1}{\xi}(e^{-\lambda t_s})] + [w_2 c \frac{t_s}{2}(1 - e^{-\lambda t_s})] \tag{4.6}$$

In the above cost function equation, the first part is the cost of wasting energy, and the second part represents the cost of sensing data report latency. In order to drive the optimal sleeping time t_s^*, we need to take partial derivative of the cost function (4.6) in terms of t_s. Since we don't know the true value of λ, we use the estimated $\overline{\lambda_t}$ from Equation (4.1) in the cost function. The optimal sleeping time t_s^* should make the derivative equal to zero, which means that t_s^* can be

derived from the following equation:

$$\left.\frac{\partial f}{\partial t_s}\right|_{t_s^*} = \frac{w_2 c}{2} + e^{-\overline{\lambda_t} t_s^*}\left[\frac{w_2 c t_s^* \overline{\lambda_t}}{2} - \frac{\overline{\lambda_t} w_1}{\xi} - \frac{w_2 c}{2}\right] = 0 \tag{4.7}$$

Algorithm 4.1: Procedure of proposed adaptive scheduling approach (ADP).

Result: Each node computes the optimal sleep time period t_s^* for the next sleep-wake duty cycle, determines the next wake-up time

Initialization: The network operator sets the values of data importance factor c for each sensor node and sets the value of weight factors w_1 and w_2 in order to achieve a balance between energy saving and data report latency;

begin

When a node wakes up

1. Prediction: estimate the new $\bar{\lambda}_t$ using Equation 4.1 based on past observations and prior estimated value.
2. Updating: update the feedback information, i.e., derive $r = \frac{1}{\xi}$ and $\overline{\lambda_t}$.
3. Optimization: derive t_s^* based on Equation 4.7 by using the feedback value of $r = \frac{1}{\xi}$ and the estimated Poisson arrival rate $\overline{\lambda_t}$.

if *event has happened during prior sleep period* **then**

Action (i.e., report the event);

end

Schedule the node next wake-up time:;

next-wake-up-time=current-time + t_s^* ;

The node goes to sleep.

end

Since Equation (4.7) does not have a closed-form solution, we apply Bisection algorithm [68] for estimating the root of the Equation (4.7). When a node wakes up, its value of $r = \frac{1}{\xi}$ updates based on the current remaining battery energy. In addition, the estimated event arrival rate $\overline{\lambda_t}$ updates by the estimation Equation (4.1), then ADP relies on Equation (4.7) to determine the node's optimal sleeping time t_s^* for the next round.

Algorithm (4.1) shows the procedure of the proposed adaptive scheduling approach. It contains three steps in each wake-up cycle: prediction, updating, and optimization. The first step, Prediction, is used to predict when will the next sensing event will happen based on the event's

statistical model and the previous events observation. It will make the system adaptive to the dynamics of sensing events. The second step, Updating, is to update all the parameters in the cost function (4.6). The last step, Optimization, is to derive the optimal next-round sleeping time t_s^* based on the partial derivative (4.7).

4.4 Discussion

Our feedback optimization model is not restricted to the Poisson process. The model of sensing event occurrence could follow different distributions according to the sensor network applications, such as Pareto distribution, ON/OFF Markov models [69], and Weibull distribution.

Figure 4.2 shows this simulation-based configuration process. The values of weight factors w_1 and w_2 in our feedback optimization model (4.2) are critical for system performance. Their values can be configured in two ways by the sensor network operator: first, based on the experience of the operator and on the previous usage of the system. Second, if the operator has the model for the sensor network application based on previous observations, the optimal values of w_1 and w_2 can be defined by running the simulation of the system (like what we did in our performance evaluation) repeatedly to achieve the best simulation results.

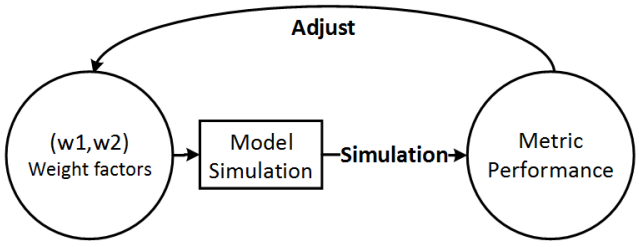


Figure 4.2: Simulation-based framework for designing weight factors w_1 and w_2 based on existence model.

4.5 Summary

In this chapter, an ADaPtive feedback approach is introduced, called ADP, for wireless sensors. It aims to effectively extend the network lifetime by saving on energy consumption and keeping data sensing report latency low. ADP utilizes a cost function intended to strike a balance between the conflicting goals of conserving energy and at the same time minimizing sensed events reporting latency. Also, a feedback mechanism that constantly monitors residual energy level and the importance of the event to be reported are incorporated, as well as predicts the next sensing event occurrence time.

CHAPTER 5: ENERGY-EFFECINT ROUTING AND CLUSTRING ALGORITHMS UNDER GAME THEORTIC FRAMEWORK

5.1 Basics of Game Theory

Game theory is a powerful mathematical tool that models strategic interaction and analysis of competition, conflict, or cooperation with multiple entities, where the constraints and payoff for actions are taken into consideration. Fundamentally, it is the study of decision-making and analysis of the behavior of two or more participants in a situation involving rewards or punishments. Different techniques available in game theory can be utilized to perform tactical analysis of the all possible situations.

A player can be a person, sensor node, machine, or group of any entities within a game. Players are a basic entity in a game and may be either cooperative or non-cooperative while aiming to maximize their outcomes according to their preference (utility function). The utility in any game is expressed by the motivation of the players. A systematic description of how the game will be played through employing the best/optimal possible strategies and the related outcomes is a solution concept of the game. A strategy is a player action throughout the game, which describes a complete plane of each player choices in all possible situations. The strategy can be either pure or mixed strategy; pure strategy is specific to take a unique action for the player in a situation, and mixed strategy specifies a probability distribution for all possible actions [59, 70].

A fundamental concept of the game theory is the ability to examine the huge number of possible situations, and game theory can also provide different methods for suggesting several probable actions along with the predicted outcome. Nash Equilibrium (NE) is one of the most significant common solution that describes a steady state condition of the game; no player can benefit by changing her/his strategy while the other players keep their strategies unchanged. In addition, this solution dose not specify how the steady state in the game can be reached. Nash Equilib-

rium classified into two major types: Pure Nash Equilibrium (PNE) and Mixed Nash Equilibrium (MNE). MSN is a probability distribution over the set of pure strategies.

Evolutionary game theory is another elegant means in game theory which models and studies the evolution of the population, and the interaction among rational agents, towards the optimal strategies that evolve over time by focusing more on the dynamics of strategic change (i.e., strategy adaptation over time). The evolutionary game provides an effective modeling tool to describe and analyze models of population behavior as well as design efficient strategies in communication networks. The difference compared with a classical game theory is that evolutionary game theory focuses more on the dynamics of strategy change, where the decision processes can be seen as the strategy evolution over time. An evolutionary stable strategy is a behavior that, when adopted by a population of players, cannot be invaded by an alternative strategy.

5.2 Overview

One of the major challenges in a wireless sensor network (WSN) is to extend the network's lifetime by minimizing the energy consumption. One of the ways to do so is to reduce network congestion as it increases delays and introduces additional packet collisions— thus, adversely affecting network performance. The heterogeneity of the paths can be in the sense that each path is associated with different costs according to the various routing metrics. Paths with lower cost in terms of transmission energy are more attractive for sensor nodes as compared with higher cost paths. However, if every node tries to select the shortest path to its target, it will result in collisions and lead to quick energy depletion among nodes. Thus, forwarding packets through the lowest energy-consumed path may not always be optimal for the network lifetime. As a result, nodes are expected to have a clear preference over a set of available paths and every sensor node should have an incentive for altruism to avoid the overheads of retransmitting dropped packets due to a collision, which can cause more depletion of the energy.

Selection of clusterheads using energy efficient clustering algorithms in WSNs is another crucial issue in WSNs algorithms design. As clusterheads and cluster members (i.e., non-clusterhead nodes) have different energy consumption rates, it is necessary that all nodes resort to some rational scheme such that the connectivity and proper functioning of the network is not compromised.

In this chapter, we address the challenges that raise due to the absence of a centralized enforcement mechanism and present an evolutionary routing congestion game that would ensure long-term routing with a fair distribution of heterogeneous paths among sensor nodes. We derive the evolutionary stable strategy (ESS) of the game and prove that the derived incumbent strategy cannot be invaded by a greedy strategy i.e., mutant strategy. Furthermore, we derive the replicator dynamic of the proposed game in order to show the behavior of the sensors in selecting the paths. The mechanism of the replicator dynamics also shows how the nodes learn from their strategic interactions and modify their strategies at every stage of the game until reaching a stable strategy (ESS). In addition, we propose a Cost and Payment-based clustering Algorithm (CoPA) for achieving energy efficiency in wireless sensor networks under a game theoretical framework. The analysis is based on a non-cooperative, repeated general-sum game, where each node behaves selfishly in order to maximize its lifespan (payoff). We demonstrate that the correlated equilibrium is a practical solution for clusterhead selection, which provides better performance than the Nash Equilibria. Correlated equilibrium provides a balance between the fully cooperative solution and the fully non-cooperative solution in terms of implementation overhead. CoPA produces a balanced distribution of responsibilities and energy consumption between the sensor nodes as well as maximizing the minimum payoff for every node.

The rest of the chapter is structured as follows: the details of an evolutionary game for efficient routing in WSNs are proposed in subsection 5.3. In subsection 5.4, we propose a Cost and Payment-based clustering Algorithm (CoPA) for achieving energy efficiency in WSNs under a game theoretical framework in particular. Summary is drawn in the last section 5.5 of this chapter.

5.3 An Evolutionary Game for Efficient Routing in WSNs

In this section, we take an evolutionary game theoretic approach to analyze the congestion issue in routing in order to show how sensor nodes in a WSN could evolve their routing strategies to transmit data packets in an efficient and stable manner. We derive the equilibrium state for the routing game and prove that there is no mutant— an individual node that adopts another strategy to invade the evolutionary stable strategy (ESS). In addition, we introduce a replicator dynamic model to show the behavior of nodes with various strategies over time. Aiming to alleviate congestion and thereby improves the network lifetime, we propose the equilibrium solutions.

The rest of the section is organized as follows: We highlights the motivation for the idea in section 5.3.1. System model and assumptions are proposed in Section 5.3.2. Game structure, investigation of the NE and ESS, and fairness analysis for the game solution are proposed in section 5.3.3.

5.3.1 Motivation for the Proposed Idea

One of the most motivating research topics in communication networks is finding optimal routes. Game theory is one of various research tools that have been proposed to investigate the routing issue, where game theoretical methodologies have been successfully used in sensor networks [41]. A game theoretic model with utility functions considering forwarding and routing in the presence of adversaries is introduced in this section. Though the route selection problem in a WSN is a well investigated problem, we are motivated to explore further where the objective is to alleviate energy consumption and collisions through a game theoretic framework. An evolutionary game theory provides a useful modeling tool of the various models of computation in game theory to design the strategies and study the behavior of populations in communication networks. We leverage concepts from evolutionary game theory and model the routing decisions in a WSN as a non cooperative evolutionary game. The mixed strategy Nash Equilibrium (NE) in our routing

game is proved to be an evolutionary stable strategy (ESS); where there are no other strategies except this ESS, can dominate the population. The payoff for every node, also referred to as a player[1], is determined by the packet transmitting cost, which in turn depends on the distance between the nodes.

Choosing the shortest distance between the source and the next neighbor hop is preferable for each player in the routing game because it will consume the least amount of energy for the transmission, thereby increasing the payoff. The players who transmit the packets through the shortest path will gain a higher payoff/lower cost compared with the players who transmit through longer paths. Selection of the shortest path to the target by every player, however, results in collisions and leads to energy depletion. Thus, forwarding the packet through the lowest energy path may not always be optimal for the network lifetime. The replicator dynamics of our game is presented in order to show the behavior of the system over a period of time and model the adaptation of the hop selection strategies. We study how the sensor nodes improve their strategy selection over time until they converge to an evolutionary stable strategy. In addition, the population cannot be invaded by any other populations of the nodes once the strategies converge to ESS, and the system will reach stability. Reducing the load and avoiding collision on the most used routes by distributing the data transmission task on all possible routes is the game's objective.

5.3.2 System Model and Assumptions

5.3.2.1 System Model

We consider an anti-coordination routing game where there is a set of $\mathcal{N}$ homogeneous sensors (i.e., players) that are randomly distributed in a designated area. Each player has to select a path to transmit packets. We model the set of next hops that are available for a node $\mathcal{R} = \{1, 2, 3, ..r\}$. We consider a routing game where each packet's path is controlled indepen-

[1]Throughout this chapter, we use the terms **player** and **node** interchangeably.

dently by a rational player in order to minimize the cost of transmission and latency. Furthermore, each node takes its own decision to transmit a packet without cooperation with other nodes. Each selected hop (i.e., hop r) has a specific cost C_r which is related to the distance between the transmitter and receiver (different hops sustain diverse transmission energy cost). For example, if the distance between the next hop and the transmission node is increased, the cost of transmission will also increase. This is because all receivers must have the signal to interference and noise ratio (SINR) above a certain threshold in order to decode received signals correctly. Players are assumed to be non-cooperative and rational, i.e., they are interested in minimizing their own cost of transmission and they do not share a common goal to cooperate with each other. The energy model will determine the transmission cost C and payoff u for selecting a specific hop, which will be introduced in the following subsections. As demonstrated subsequently, the evolutionary game is concerned with *the evolution of the strategies, payoffs, and stability* [71]. Thus, the number of sensor nodes is not significant in the game model.

5.3.2.2 Cost Model

Most of the sensors' energy is used during packet forwarding. Many energy models [72, 73] have been used for energy consumption in WSNs. In our model, the total cost C of forwarding a packet consists of two parts: i) the energy spent for transmitting the packet and ii) the energy consumed for receiving the packet. Thus,

$$C = C_{tx}(d) + C_{rx} \tag{5.1}$$

where $C_{tx}(d)$ is the cost of transmission the packet to another over distance d, and C_{rx} is the cost of receiving it. C_{tx} is defined as:

$$C_{tx}(d) = e_{(tx-elec)} + e_{amp} \cdot d^{\alpha} \tag{5.2}$$

where $e_{tx-elec}$ is the energy consumption of transmission circuit, and e_{amp} is the transmit amplifier dissipation in order to achieve the required signal level. α represents the propagation loss exponent (i.e., typically $\alpha = 2$ for free space). The cost of receiving the packet is:

$$C_{rx} = e_{(rx-elec)} \tag{5.3}$$

where $e_{(rx-elec)}$ is the receiving circuitry dissipation. In our game model, it is noteworthy that any other positive value for the cost of packet forwarding derived from other energy models can be used in the game without affecting our analysis and the outcome.

5.3.2.3 Assumptions and Notations

The assumptions of the incentive game model as following:

- *Populations:* All sensor nodes are grouped into several populations according to their geographical positions, and we model the game as an asymmetric routing game between two populations (i.e., $\upsilon = \{\mathcal{A}, \mathcal{B}\}$). All nodes in each population have the same strategy set and payoff matrix. In an evolutionary game, the number of nodes does not play any role in the game model, where the payoff of a strategy depends on the strategy adopted by the others, but not on who is playing the strategy [74].

- *Strategy space:* each node has a set of available actions/strategies represented as $\mathcal{S} = \{s_r | r \in \mathcal{R}\}$, where $\mathcal{R}$ is the set of next hops available in the game.

- *Payoffs and cost:* Obtaining the nearest hop will result in a lower transmission cost and thus a higher payoff. Similarly, selecting a farther hop will result in a higher transmission cost and a lower payoff. The next hops selected by different players simultaneously may interfere with each other, raising the contention situation, and wasting the transmission energy of all nodes in question. Each selected hop for either node will incur a specific amount of energy

that is the cost of transmitting the packet. This cost denoted by C (as was defined in equation (5.1)). As an example, selecting r as the next hop to transmit the packet individually from population $\mathcal{A}$ will cost $C_{\mathcal{A}r}$.

- *Non-cooperative behavior:* All sensor nodes are independent as they do not cooperate with each other for a common goal. Nodes are expected to have a clear preference of selecting the best paths over a set of available choices, and the nodes are always interested in transmitting packets through the route with the least possible minimum cost (i.e., the minimum value of C). Therefore, if many nodes take this same routing strategy, this rational behavior of sensor nodes will intuitively result in further congestion and lead to energy depletion of the nodes along those paths.

For reader's convenience, we list the main mathematical notations and acronyms Table 5.1.

5.3.3 An Evolutionary Routing Game

In this section, we first provide some a basic concept of evolutionary game theory as well as the structure of our routing game. Then, we derive the equilibrium state for the game as a solution for 2-hop scenario, followed by extension for multi-hop scenario by driving the so-called Replicator Dynamics of the game.

The incentive anti-coordination routing game proposed in this chapter is a non-cooperative repeated game with perfect information, where the nodes have perfect knowledge about the utility function, which is a common information to all nodes. The nodes are able to know other nodes' selection and their payoffs in the past. Furthermore, each node in WSNs behaves rational and selfishly in order to obtain the best route to forward his own packets with minim cost of energy consumption (maximize the own utility).

The evolutionary game provides an effective modeling tool to describe and analyze models of population behavior as well as design efficient strategies in communication networks. The

difference compared with a classical game theory is that evolutionary game theory focuses more on the dynamics of strategy change, where the decision processes can be seen as the strategy evolution over time. An evolutionary stable strategy is a behavior that, when adopted by a population of players, cannot be invaded by an alternative strategy. In this paper, we consider the action of selecting a specific hop as nodes' strategy in our routing game. We need to provide the evolutionary stability analysis of Pure Strategy Nash Equilibrium (PSNE) and Mixed Strategy Nash Equilibrium (MSNE) in the game in order to seek a fair and stable solution for the long term. In addition, we prove that MSNE can not be invaded by a greedier strategy (i.e., mutant strategy).

5.3.3.1 Routing Game Structure

The evolutionary routing game is represented as $\mathcal{G} =< \mathcal{R}, \mathcal{S}, \mathcal{U} >$, where $\mathcal{R}$ represents the set of next hops available in the game; $\mathcal{S} = \{s_r | r \in \mathcal{R}\}$ is the strategy space, which is the set of actions that are available for the players. The payoff for playing strategy s_r and s_t is denoted by $u(s_r, s_t) \in \mathcal{U}$ when competing against each other. This happens when the player who is adopting the strategy s_r meets another player who is adopting the s_t strategy. In our game, the cost of transmission is permanently preferred to be low, which will increase the payoff and prevent energy wastage. Thus, we define the payoff as:

$$u(s_r, s_t) = \begin{cases} (\frac{1}{C_{\upsilon r}}, \frac{1}{C_{\upsilon t}}) & \text{when} \quad r \neq t, \quad \upsilon \in \{\mathcal{A}, \mathcal{B}\} \\ (0, 0) & \text{when} \quad r = t \end{cases} \tag{5.4}$$

where $C_{\upsilon r}$ is the transmission cost of the packet through hop r, which either belongs to the population $\mathcal{A}$, or belongs to the population $\mathcal{B}$. For example, $C_{\mathcal{B}r}$ denotes the cost of selecting hop r by the player, who belongs to population $\mathcal{B}$.

We define the routing game as a strategic matrix shown in Table 5.2 with a player set composed of players that comprise $\upsilon = \{\mathcal{A}, \mathcal{B}\}$ populations. The payoff for players playing

Table 5.1: List of Notations and Acronyms

Notation	Definition
NE	Nash Equilibrium
ESS	Evolutionary Stable Strategy
PSNE	Pure Strategy Nash Equilibrium
MSNE	Mixed Strategy Nash Equilibrium
$\mathcal{R}$	Set of available hops in the game
$\mathcal{S}$	Strategy space, (set of actions that are available for the players ($S = \{s_r \| r \in R\}$))
$\mathcal{U}$	Set of hops' utilities
s_r	Strategy of selecting hop r
u_r	Utility for selecting hop r.
$u(s_r, s_t)$	The payoff for playing strategy s_r and s_t when competing against each other
s_i	Strategy played by player i
s_i^*	Strategy of player i which is the best response to s_i^*
s_{-i}^*	Best strategy played by player other than player i
$\upsilon \in \{A, B\}$	Population
$C_{\upsilon r}$	Transmission cost of the packet through hop r
$C_{\upsilon t}$	Transmission cost of the packet through hop t
$\hat{P}$	Probability distribution over set of of pure strategies for any player (collection of wights in MSNE)
$(\grave{p}, \grave{q})$	Incumbent strategy/ESS probability distribution over set of hops (MSNE)
$(\hat{p}, \hat{q})$	A mutant strategy that is greedier than ESS
$EU_{\upsilon}(s_r)$	Expected Utility from selecting hop r

strategies s_r and s_t, which are competing against each other, is denoted by $u(s_r, s_t)$. For the sake of clarity in analysis and without loss any generality, we assume that $u_r > u_t$ regarding to the variety of the available routes in the network, and transmitting the packet by using the strategy s_r will cost less than transmitting the packet by using strategy s_t according to the distance between the nodes. Thus, it is preferable for all the nodes to forward the packets through hop r, which produces a high payoff. In addition, transmitting the packet through the same hop (i.e., r or t) will

cause a collision, and hence, the payoff will be zero (see Eqn. 5.4).

In addition, we initially consider a 2-available hops game i.e., we show competition between the two strategies s_r and s_t as a demonstration to clarify and analyze the performance of the game besides derives its PSNE and MSNE. Later, we utilize the same technique in the case of having multiple hops, as will be presented in the experimental results in Chapter 7. The players in our game adopt one of the two available hops (i.e., r or t). We analyze the payoff based on Table 5.2, and employ the same game formulation to answer the fundamental questions as: 1) What does a strategy s_r gain as a payoff when it meets another same strategy s_r or s_t? 2) How does the equilibrium solution make the player satisfy and respect the other's choices? As we consider the players in our game are rational, all players would maximize their payoff by minimizing the cost of energy consumption and all players' interest not to end up selecting the same strategy.

Table 5.2: Strategies Competition form of Evolutionary Routing Game (i.e., strategies s_r and s_t)

	s_r	s_t
s_r	0 , 0	$\frac{1}{C_{Ar}}$, $\frac{1}{C_{Bt}}$
s_t	$\frac{1}{C_{At}}$, $\frac{1}{C_{Br}}$	0 , 0

5.3.3.2 Pure Strategy Nash Equilibrium and Evolutionary Stability for the Game

In this subsection, we derive the PSNE as first potential solutions for our evolutionary anti-coordination routing game. Then, we analyze its evolutionary stability.

5.3.3.2.1 Pure Strategy Nash Equilibrium

According to definition 1, we prove that our evolutionary routing game has two pure Nash Equilibrium strategies.

Definition 1: A Pure Nash Equilibrium [70] of the routing game is a strategy profile $s^* \in \mathcal{S}$

of actions, such that:

$$u(s^*{}_i, s^*\_i) \geq u(s_i, s^*\_i), \forall i \in \mathcal{N} \tag{5.5}$$

In other words, the strategy $s^*{}_i$, to be pure NE it must satisfy the above condition. This condition means that no player i has an incentive to deviate to another strategy to gain a higher payoff than the one who is playing $s^*{}_i$, given that the other players' strategies remain the same s^*_i.

Lemma 1: In the evolutionary routing congestion anti-coordination game, strategy pairs (s_r, s_t) and (s_t, s_r) are pure strategy NE.

Proof. Suppose two nodes are picked randomly from two large populations of sensor nodes in the network. These nodes are supposed to select one of the two strategies, each competes against the other, in order to transmit the packet. In Table 5.2, assume the row and the column are the two players from populations $\mathcal{A}$ and $\mathcal{B}$, respectively. These players select strategy pairs (s_r, s_t) and (s_t, s_r). The payoffs of the selection are $\frac{1}{C_{\mathcal{A}r}}, \frac{1}{C_{\mathcal{B}t}}$ and $\frac{1}{C_{\mathcal{A}t}}, \frac{1}{C_{\mathcal{B}r}}$, respectively. Let us say that the players select strategy pairs (s_r, s_r) and (s_t, s_t) instead. Thus, the payoffs for those strategy pairs will be zero. This means that the player who is playing strategy s_r does not have an incentive to change the strategy to s_t because of the penalty of reducing the payoff according to equation 5.4. As a result, we can say that strategy pairs (s_r, s_r) and (s_t, s_t) are not profitable deviations. According to the PSNE definition 1, the strategy pairs (s_r, s_t) and (s_t, s_r) are a pure strategy NE for this game. □

5.3.3.2.2 Evolutionary Stability of the Game's PSNE

We examine the PSNE evolutionary stability of the routing game according to definition 2 as follows:

Definition 2: In a symmetric game, the strategy s is evolutionary stable ESS in pure strategies if:

1. $u(s, s)$ is NE; $u(s, s) > u(\acute{s}, s)$ for all $\acute{s}$ and

2. if $u(s,s) = u(\acute{s},s)$, then $u(s,\acute{s}) > u(\acute{s},\acute{s})$

That means the players will play (s,s), which is a symmetric NE. The strategy s is called evolutionary stable if a small group playing a different strategy, $\acute{s}$, which is referred to as the mutant strategy, would disappear with time. The ESS [70] defined above as any evolutionary stable strategy must be a symmetric pure NE, where the performance of strategy s against itself is better than it does against a mutant strategy. However, if the strategy is not strictly Nash, it should satisfy the second condition of the evolutionary stability. The second condition defined as that the incumbent s must do strictly better against the mutant $\acute{s}$ than a mutant strategy does against another mutant strategy. Consider a group of two populations playing the same strategy s, which is referred to as the incumbent strategy. In this game, the pure strategies are not symmetric pure NE where the payoff of strategy s_r is different from the payoff of strategy s_t (i.e., $u(s,s) < u(\acute{s},s)$). According to the definition 2 of ESS, the pure strategy NE in our game is not evolutionary stable, and it is impractical solution for the long term strategy of routs selection in WSNs, where it is always unfair for the player that select the higher cost of energy consumption path.

5.3.3.3 Mixed Strategy Nash Equilibrium and Evolutionary Stability for the Game

In this subsection, we derive the MSNE as second potential solutions for our evolutionary anti-coordination routing game, and we analyze its evolutionary stability.

5.3.3.3.1 Mixed Strategy Nash Equilibrium

Definition 3: The Mixed Strategy Nash Equilibrium [75] of the routing game is a probability distribution $\hat{P}$ (collection of weights) over the set of pure strategies $\mathcal{S}$ for any player such that:

$$\hat{P} = (p_1, p_2, p_3, ..., p_r) \in \mathbb{R}^{\mathcal{R}} \geq 0, \quad \text{and} \quad \sum_{t=1}^{\mathcal{R}} p_t = 1 \tag{5.6}$$

The pure strategy will be available with certain probabilities where the payoffs from all opponents of their strategies are eventually equal. Thus, the expected payoffs given to strategies in a Mixed Nash Equilibrium are equal.

In our game, let $\grave{p} = \{p, 1-p\}$ denote the proportions of the population $\mathcal{A}$ adopting s_r and s_t strategies, respectively, and $\grave{q} = \{q, 1-q\}$ denote the proportion of the population $\mathcal{B}$ adopting s_r and s_t strategies, respectively. In a 2-hop scenario, player 1, who belongs to population $\mathcal{A}$, plays strategy s_r with probability p and strategy s_t with $1-p$ probability. Player 2, who belongs to population $\mathcal{B}$, plays strategy s_r with probability q and strategy s_t with $1-q$ probability. We calculate those probabilities using the mixed strategy algorithm and the payoff in Table 5.3.

Table 5.3: Strategies Competition form of Evolutionary Routing Game with Probability Distribution $\hat{p}$ over the Pure Strategies (i.e., strategies s_r and s_t).

	Prob.$(s_r) = p$	Prob.$(s_t) = 1-p$
Prob.$(s_r) = q$	0 , 0	$\frac{1}{C_{Ar}}$, $\frac{1}{C_{Bt}}$
Prob.$(s_t) = 1-q$	$\frac{1}{C_{At}}$, $\frac{1}{C_{Br}}$	0 , 0

According to Mixed Nash definition 3, the expected utility from playing strategy s_r is equal to the expected utility for playing strategy s_t for any player as follows:

$$EU_{\upsilon}(s_r) = EU_{\upsilon}(s_t), \quad \upsilon \in \{\mathcal{A}, \mathcal{B}\} \tag{5.7}$$

The expected utility for playing strategy s_r for the player who belongs to $\mathcal{A}$ population and the player who belongs to population $\mathcal{B}$, respectively, are:

$$EU_{\mathcal{A}}(s_r) = q \cdot 0 + (1-q)\frac{1}{C_{\mathcal{A}r}} \tag{5.8}$$

$$EU_{\mathcal{B}}(s_r) = p \cdot 0 + (1-p)\frac{1}{C_{\mathcal{B}r}} \tag{5.9}$$

The expected utilities for playing strategy s_t for the players in the two populations are:

$$EU_{\mathcal{A}}(s_t) = q\frac{1}{C_{\mathcal{A}t}} + (1-q)\cdot 0 \tag{5.10}$$

$$EU_{\mathcal{B}}(s_t) = p\frac{1}{C_{\mathcal{B}t}} + (1-p)\cdot 0 \tag{5.11}$$

Setting (5.8) and (5.10) equal as in (5.7), then solve it to find the probability distribution $\grave{p} = \{p, 1-p\}$. Similarly, setting (5.9) and (5.11) equal as in (5.7), then solve it to find the probability distribution $\grave{q} = \{q, 1-q\}$ such as:

$$p = \frac{C_{\mathcal{A}t}}{C_{\mathcal{A}t} + C_{\mathcal{A}r}}, \quad 1-p = \frac{C_{\mathcal{A}r}}{C_{\mathcal{A}t} + C_{\mathcal{A}r}} \tag{5.12}$$

$$q = \frac{C_{\mathcal{B}t}}{C_{\mathcal{B}t} + C_{\mathcal{B}r}}, \quad 1-q = \frac{C_{\mathcal{B}r}}{C_{\mathcal{B}t} + C_{\mathcal{B}r}} \tag{5.13}$$

The players from $\mathcal{A}$ and $\mathcal{B}$ populations adopt the strategy s_r with probabilities (p, q), respectively, and the strategy s_t with probabilities $(1-p, 1-q)$, respectively. The players in the routing game mix their selections of the next hop to transmit the data packet with (p, q) and $(1-p, 1-q)$ probabilities. In addition, none of the players would change the strategy with an expectation of gaining a better payoff. The reason behind this behavior is that adopting the strategies in that manner will represent the same outcome.

5.3.3.3.2 Analysis Evolutionary Stability of the Game's MSNE

Previously, we already proved that the game solution is a Mixed Strategy Nash Equilibrium $(\grave{p}, \grave{q})$. Here, we analyze the evolutionary stability of Mixed Strategy Nash Equlibrium (MSNE) (i.e., $(\grave{p}, \grave{q})$) in our asymmetric routing game according to definition 4 of asymmetric evolutionary stable strategy [76] such as:

Definition 4**:** Define $(\grave{p}, \grave{q})$ as a two-species evolutionary stable strategy [76]. if it is asymptotically stable under the two-dimensional equation whenever it is based on the strategy pair $(\grave{p}, \grave{q})$ and $(\hat{p}, \hat{q})$, when $(\grave{p}, \grave{q}) \neq (\hat{p}, \hat{q})$.

In other words, the two-species ESS with strategy pair $(\grave{p}, \grave{q})$ cannot be invaded by a mutant subsystem, which uses a different strategy pair $(\hat{p}, \hat{q})$.

Lemma 2: Our mixed strategy Nash equilibrium $(\grave{p}, \grave{q})$ is a two-species evolutionary stable strategy.

Proof. First, we define the replicator equations, which is ruling the behavior of the system over time [77], based on the strategy pair $(\grave{p}, \grave{q})$. In our routing game, we define the replicator equation such that the fraction of strategy s_r grows at a rate equal to its fitness minus the average fitness of the player. We have the following replicator equations:

$$\begin{aligned} \dot{p} &= p[(\frac{1-q}{C_{\mathcal{A}r}}) - (\frac{p(1-q)}{C_{\mathcal{A}r}} + \frac{(1-p)q}{C_{\mathcal{A}t}})] \\ &= p(1-p)(\frac{1-q}{C_{\mathcal{A}r}} - \frac{q}{C_{\mathcal{A}t}}) \end{aligned} \tag{5.14}$$

$$\begin{aligned} \dot{q} &= q[(\frac{1-p}{C_{\mathcal{B}r}}) - (\frac{q(1-p)}{C_{\mathcal{B}r}} + \frac{(1-q)p}{C_{\mathcal{B}t}})] \\ &= q(1-q)(\frac{1-p}{C_{\mathcal{B}r}} - \frac{p}{C_{\mathcal{B}t}}) \end{aligned} \tag{5.15}$$

Second, we need to find the stable fixed point for the two replicator equations. We have the MSNE point, which we calculated in 5.3.3.3.1. We proved how this point is a fixed point under the two replicator equations (5.14) and (5.15).

Since we already have a stable point $(\grave{p}, \grave{q})$ in our model, we need to show that the point is fixed under the replicator equations. Therefore, we need to satisfy that the last part (i.e., $(\frac{1-q}{C_{\mathcal{A}r}} - \frac{q}{C_{\mathcal{A}t}})$ and $(\frac{1-p}{C_{Br}} - \frac{p}{C_{\mathcal{B}t}})$) in equations (5.14) and (5.15), respectively, should equal zero. Therefore, if we substitute the values of p and q from equations (5.12) and (5.13) with these last parts, we

will get zero. As a result, $(\grave{p}, \grave{q})$ is a asymptotically stable fixed point for the replicator dynamic. Based on asymmetric ESS [76], our mixed strategy NE $(\grave{p}, \grave{q})$ is a two-species evolutionary stable strategy. □

5.3.3.3.3 Numerical Analysis of Evolutionary Stability for the Game's MSNE

For the sake of certainty, we will analyze the ESS for the proposed MSNE solution by satisfying the condition of the following theorem [76] numerically in this part.

Theorem[76]: $(\grave{p}, \grave{q})$ is a two-species ESS if and only if

either $\grave{p} \cdot (D\hat{p} + E\hat{q}) > \hat{p} \cdot (D\hat{p} + E\hat{q})$

or $\grave{q} \cdot (F\hat{p} + G\hat{q}) > \hat{q} \cdot (F\hat{p} + G\hat{q})$

for all strategy pairs $(\hat{p}, \hat{q})$ that are sufficiently close (not equal) to $(\grave{p}, \grave{q})$. D, E, F, and G are the payoff matrices for interspecies interaction.

In our routing game, suppose two sensor nodes are picked randomly from two population (i.e., $\mathcal{A}$ and $\mathcal{B}$), and these nodes are supposed to select one of the two strategies (i.e., s_r and s_t), which compete against each other in order to transmit the data packet. Assume that we have the payoff matrix values for Table 5.2 as: $C_{\mathcal{A}t} = 4, C_{\mathcal{A}r} = 2, C_{\mathcal{B}t} = 8$, and $C_{\mathcal{B}r} = 6$. Based on those values, we calculate the MSNE and the rest of the elements as: $(\grave{p}, \grave{q}) = \begin{pmatrix} \frac{4}{7} & \frac{2}{3} \\ \frac{3}{7} & \frac{1}{3} \end{pmatrix}$, $D = \begin{pmatrix} 0 & \frac{1}{2} \\ \frac{1}{4} & 0 \end{pmatrix}$, and $E = \begin{pmatrix} 0 & \frac{1}{6} \\ \frac{1}{8} & 0 \end{pmatrix}$. D and E are the payoff matrices for interspecies interactions. Suppose there are small groups adopting a mutant strategy $(\hat{p}, \hat{q})$ instead, which is greedier than the incumbent strategy $(\grave{p}, \grave{q})$. Furthermore, assume that the mutant strategy selects the near hop r with higher probability (i.e., $p + \delta$, $q + \delta$) and selects the farther hop t with lower probability (i.e.,$(1 - p) - \delta$, $(1 - q) - \delta$), where δ is a small positive number (i.e., $\delta = 0.1$). Thus, $(\hat{p}, \hat{q}) = \begin{pmatrix} \frac{4}{7} + \delta & \frac{2}{3} + \delta \\ \frac{3}{7} - \delta & \frac{1}{3} - \delta \end{pmatrix}$. Then, by substituting those values in the first condition of the theorem [76], we have $\grave{p}.(D\hat{p}+E\hat{q}) >$

$\hat{p}.(D\hat{p} + E\hat{q})$ (i.e., $0.23 > 0.22$). Accordingly, $(\grave{p}, \grave{q})$ cannot be invaded by the greedier mutation and is ESS.

5.3.3.4 R-Hop Scenario and Replicator Dynamics

In this subsection, we provide a dynamic way to achieve the equilibria and extend our analysis to the R-Hop scenario for our evolutionary routing game according to the concept of replicator dynamics. We introduce the replicator dynamic model in order to show how the players, who repeatedly play the routing game, evolve their behavior in every stage of the game. The populations learn with each strategy's interaction until they reach a stable state. Replicator dynamics describe the populations' behavior of sharing associated with different strategies, that evolve over time [77]. In the following equations, we derive the replicator dynamics of our routing game framework with r hops.

In the following, we introduce fitness defined by our replicator dynamic equations. From the above sections 5.3.3.3, let consider two populations of interacting nodes. Each time nodes from one population (row players $\mathcal{A}$) are randomly paired with nodes from the other population (column players $\mathcal{B}$). All players have a set of hops $\mathcal{R}$, and strategy $s_r \in \mathcal{S}$ are adopted. Let $\grave{p} = \{p_1, p_2, p_3, ..., p_r\}$ and $\grave{q} = \{q_1, q_2, q_3, ..., q_r\}$ denote the proportion of the two-population adopting $s_1, s_2, s_3, ..., s_r$ strategies, respectively, where summation of the proportions equals to 1 (i.e., $\sum_{i=1}^{r} p_i = 1$ and $\sum_{i=1}^{r} q_i = 1$) as described in section 5.3.3.3. Let $(\grave{p}, \grave{q})$ represent the incumbent strategy of selecting hop r with probability p_r,q_r. In addition, let the set of $\mathcal{U} = \{u_1, u_2, u_3, ...u_r\}$ represent the average payoff of the players selecting hop r at a given stage of our game. Furthermore, let u_r denote the utility function of adopting strategy s_r. The payoff of selecting hop r strategy s_r for row player ($\mathcal{A}$) is given by:

$$u_r = u_0 + \sum_{x=1}^{R} q_r u(s_r, s_t), \quad \forall r, t \in \mathcal{R} \tag{5.16}$$

The payoff of selecting hop r strategy s_r for column player ($\mathcal{B}$) is given by:

$$u_r = u_0 + \sum_{x=1}^{R} p_r u(s_r, s_t), \quad \forall r, t \in \mathcal{R} \tag{5.17}$$

where u_0 is the initial fitness of every player, and $u(s_r, s_t)$ is the fitness of selecting hop r in pairwise competition against adopting hop t.

Let $\overline{u_{\mathcal{A}}}$ and $\overline{u_{\mathcal{B}}}$ denote the average fitness for entire population $\mathcal{A}$, and $\mathcal{B}$, respectively, which are given by:

$$\overline{u_{\mathcal{A}}} = \sum_{y=1}^{r} p_y (q_y u_y), \quad \forall y \in \mathcal{R} \tag{5.18}$$

$$\overline{u_{\mathcal{B}}} = \sum_{y=1}^{r} q_y (p_y u_y), \quad \forall y \in \mathcal{R} \tag{5.19}$$

For each next time slot, the probability $(\check{p_r}, \check{q_r})$, of selecting next hop r of the game is calculated by:

$$\check{p_r} = p_r + \frac{q_r(u_r - \overline{u_{\mathcal{B}}})}{\overline{u_{\mathcal{B}}}} \tag{5.20}$$

$$\check{q_r} = q_r + \frac{p_r(u_r - \overline{u_{\mathcal{A}}})}{\overline{u_{\mathcal{A}}}} \tag{5.21}$$

The proportion of sensors selecting hop r in the next time slot will be either increased or decreased according to the comparison of the average fitness of selecting that hop to the overall fitness of the entire sensor population in the current time slot. According to our evolutionary replicator equations, the next particular hop will be selected more frequently in a subsequent time slot if the payoff of selecting that hop is higher than the average overall fitness of the entire sensor network. Algorithm 5.1 shows the summary of the proposed replicator dynamics algorithm.

Algorithm 5.1: Replicator dynamics

```
Results: Converge the startegy of selection hops to ESS;
Initialization: Set the available hops R and their related utilities
 (payoffs) U, intial fitness u_0, population distribution p_r and q_r, hop
 utilities u_r ;
begin
    for evrey time slot of the game do
        At current time caculate:
          1. average payoff of selecting hop r for sensors
             population (i.e.,A and B) at current time
             (equations (5.16-5.17))
          2. Calculate average fitness ū for entire sensor
             nodes population (equations (5.18-5.19))
        Caclulate hop selection startegies for next time slot (equations
         (5.20-5.21)) ;
    end
end
```

5.3.3.5 Fairness Analysis

Fairness is an important performance criteria in routing protocols for resource sharing. Janin's fairness index [78] is one of the efficient measurements to determine the fair share of the system's resources. In our proposed game, we analyze the fairness of both pure and mixed solutions of the Nash Equilibria, and consider the case of 2-hop scenario of the routing sharing game for the sake of clarity. Furthermore, the same concept will be applied in the case of R-hop scenario. Measuring of the fairness of the derived Nash equlibria, and the guaranteeing of the provision of the same utilities to all users, is achieved by following Jain's equation:

$$\mathcal{J}(u_1, u_2, u_3, ..., u_N) = \frac{(\sum_{i=1}^{\mathcal{N}} u_i)^2}{\mathcal{N} \cdot \sum_{i=1}^{\mathcal{N}} {u_i}^2} \tag{5.22}$$

where $\mathcal{N}$ is the number of sensor nodes and the utility of allocating the hops is given by u_i. The index of the equation are bounded between 0 (worst case and totally unfair system) and 1 (best case and perfectly fair system). We analyze the fairness of the solutions of the game as follows:

1. As we proved earlier that the Pure Strategy Nash Equilibrium (PSNE) for the evolutionary routing anti-coordination game is the pair of strategy (s_r, s_t) and (s_t, s_r). According to our previously named assumption for 2-hop scenario, transmitting the packet through hop r will provide a higher payoff than transmitting the packet through hop t. This means that $u_r \neq u_t$ and the distribution of payoffs for the ratio in equation (5.22) are unequal and less than 1. Also, one player in the game always gets a smaller payoff than the other. Thus, PSNE is not a fair solution because it does not result in equal payoff for all nodes.

2. Another finding for the game is that a Mixed Strategy Nash Equilibrium (MSNE) is the probability distribution $\grave{p}, \grave{q}$ (collection of weights) computed by equations (5.12) and (5.13). Based on definition 3 of MSNE, the expected utility of the strategies for all players are equal even though the cost of transmitting the packet through the hops are different, and that makes the opponents indifferent about their choice of strategy. Having equal payoffs u_i will maximize the value of the equation (5.22) which equals 1. As a result, the MSNE's resource distribution is fair.

5.4 A Game Theoretic Approach for Energy-Efficient Clustering Algorithm in Sensor Networks

One of the important issue in WSNs is selection of clusterheads using energy efficient clustering algorithm. In this section of the chapter, we take a game theoretic approach to devise a clustering algorithm for WSNs. Analyzing and predicting the rational and selfish behaviors of various entities– the decisions of which determine the outcome of the game using game theory that have been applied to numerous areas of wireless communications [19], a powerful mathematical tool.

The rest of the section is organized as follows:We highlights the motivation for the idea in section 5.4.1. The network model is presented in Section 5.4.2. The clustering game is presented in Section 5.4.3. We propose the clustering technique in Section 5.4.4.

5.4.1 Motivation for the Proposed Idea

In our approach, the nodes are the players who play the clustering game. We propose a Cost and Payment-based clustering Algorithm (CoPA) where we formalize the profits and losses for each node. In order to balance the energy consumption, CoPA has the provision to alternate the responsibility of a clusterhead among the nodes, and it uses a weighted metric that combines the transmission power and energy of each node. We formulate an anti-coordination clustering game for 2 players as well as N players using only local information. The Correlated Equilibrium (CE) for the clustering game is derived by solving the linear optimization, and the adaptive regret matching (no-regret) algorithm is utilized to guarantee convergence of the probability distribution to the CE. Furthermore, in terms of the efficiency and fairness among the nodes, we prove and discuss the optimality of CE solution for the clustering game, and compare it to the pure and Mixed Strategy Nash Equilibrium (MSNE) solutions. Finally, we demonstrate that CoPA has superior performance in terms of network lifetime and system throughput by evaluating the performance of our clustering algorithm with two popular clustering techniques.

5.4.2 Network Model

We consider a network with N sensor nodes represented by the set $N = \{1, 2, 3, ..n\}$, and divide the entire network into non-overlapping clusters. Each cluster has one clusterhead that receives/transmits data packets from its cluster members and also communicates with the base station in order to deliver those data packets. Furthermore, we consider that the base station is located outside the sensing field. Apart from the communications, the clusterhead has additional responsibilities compared with the cluster members, which include aggregating (i.e., multiplexing and demultiplexing) the data of its members, packet forwarding, and sometimes scheduling. Therefore, the energy consumption rate of a clusterhead is significantly higher than the energy consumption rate of a cluster member. This leads to the situation where each node prefers not to

be a clusterhead as long as there are other nodes willing to serve as clusterheads. In case all the nodes decide to be cluster members (i.e., no clusterheads), then the data of all cluster members cannot be relayed to the BS, resulting significant data loss to the sensor network. Thus, to keep the network operating in a fair manner [79], the nodes must find a way to efficiently rotate their roles between clusterheads and cluster members. Following, we use a game theoretical approach to present a Cost and Payment-based clustering Algorithm (CoPA).

5.4.3 Clustering Game

Let us formally define the game and the cost functions of the nodes. Then we will analyze the equilibria and the no-regret learning for the correlated equilibria.

5.4.3.1 Game Framework

We formulate an anti-coordination $N-$player and $2-$ strategy symmetric game. The game is presented as $\mathcal{G} =< \mathcal{N}, \mathcal{S}, \mathcal{U} >$. The players are represented by $\mathcal{N}$; each player has the same action/strategy space represented by $\mathcal{S}$, and their utility is given by $\mathcal{U}$.

The set of strategies available to a sensor node is to decide between being a clusterhead (CH) or a cluster member (CM), and is represented as $\mathcal{S} = \{CH, CM\}$. The structure of network is described as a cost and payment model: the nodes gain a specific payoff when they select one of these strategies. Each node behaves selfishly in order to maximize its own payoff (minimize the cost) and stay alive as long as possible. A player may choose to serve as the clusterhead and carry out the additional responsibilities for its members, or refuse to be a clusterhead (e.g., prefer to be a cluster member) in order to maximize its payoff. If more than one player in close physical proximity opt to become a clusterhead, then smaller clusters emerge. As a result, unnecessary control overhead and power consumption would be incurred. However, if none of the nodes opt to be a clusterhead, all the nodes will suffer and all will obtain a payoff of 0 as the nodes will not be able to send their data to the base station.

The set of utility functions of the nodes denoted by $\mathcal{U}(s_i)$, is given by:

$$\mathcal{U}(s_i) = \begin{cases} 0 & \text{when } s_i = CM, \forall i \in N \\ \frac{1}{C_{ch}} & \text{when } s_i = CH \\ \frac{1}{C_{cm}} & \text{when } s_i = CM \end{cases} \tag{5.23}$$

where C_{ch} represents the cost of being a clusterhead, and C_{cm} represents the cost of being a cluster member. For the sake of simplicity, let us first provide the possible equilibria in the case of 2 players and their payoffs as presented in Table 5.4. Based on this payoff matrix, the best outcome occurs when one of the nodes selects to be a clusterhead and the other selects to be a cluster member.

Table 5.4: Strategic form of 2-player clustering game with strategies CH and CM.

	CH	CM
CH	$\frac{1}{C_{ch}}$, $\frac{1}{C_{ch}}$	$\frac{1}{C_{ch}}$, $\frac{1}{C_{cm}}$
CM	$\frac{1}{C_{cm}}$, $\frac{1}{C_{ch}}$	0 , 0

5.4.3.2 Cost Model

The total cost of being a clusterhead, C_{ch}, consists of two parts: i) the energy spent to transmit packets to the base station and ii) the energy consumed for aggregating the packets received from the cluster members. Thus,

$$C_{ch} = C_{tx(ch,BS)} + C_{rx,aggr} \tag{5.24}$$

where $C_{tx(ch,BS)}$ is the cost of transmission from the clusterhead to the base station, and $C_{rx,aggr}$ is the cost of receiving and aggregating the packets from the cluster members. We define $C_{tx(ch,BS)}$

as:

$$C_{tx(ch,BS)} = d_{ch,BS}^2 \cdot e_{amp} + e_{elec} \tag{5.25}$$

where $d_{ch,BS}$ is the distance between the clusterhead and the base station, e_{amp} the transmit amplifier dissipation in order to achieve the required signal level, and e_{elec} is the transmission circuitry dissipation.

As for the cost of receiving and aggregating data from cluster members, it is proportional to the cluster size (i.e., $\bar{k}$ average number of neighbors), i.e.,$C_{rx,aggr} \propto \bar{k}$.

It is to be noted that the cluster members will be at varying distances from the clusterhead and therefore the clusterhead uses different power levels to transmit to its members. (We assume that there is some power control algorithm is place– the specifics of which is beyond the scope of this chapter.) Thus,

$$C_{rx,aggr} = \sum_{i=1}^{\bar{k}} d_i^2 \cdot e_{elec} + \bar{k} \cdot e_{aggr} + e_{lis} \tag{5.26}$$

where d_i is the distance of the ith cluster member from its clusterhead and e_{aggr} is the cost of aggregation for one cluster member. e_{lis} is the cost of listening to the wireless medium even though no packets are being to transmitted.

The cost of i-th node being a cluster member is the cost of transmission from this node to its clusterhead ch_i considering the distance (d_{i,ch_i}) is calculated by:

$$C_{cm} = C_{tx(i,ch_i)} = e_{amp}.d_{i,ch_i}^2 + e_{elec} \tag{5.27}$$

According to above mentioned energy model and assuming the base station is located outside the sensing region, the cost of being a clusterhead is expected to be larger than the cost of being a cluster member, i.e.,

$$C_{ch} > C_{cm} \tag{5.28}$$

5.4.3.3 Analysis and Equilibrium

5.4.3.3.1 Pure and Mixed Nash Equilibrium

For the clustering game, we derive the solution concepts in the form of Pure and Mixed Nash Equilibrium for 2-players and N-players.

Lemma 1: Strategy pairs (CH, CM) and (CM, CH) are pure strategy NE for 2-player clustering game.

Proof. In Table 5.4, assume the row and the column are the two players from the cluster. If these players select strategy pairs (CH, CM) and (CM, CH), the payoffs of the selection will be $(\frac{1}{C_{ch}}, \frac{1}{C_{cm}})$ and $(\frac{1}{C_{cm}}, \frac{1}{C_{ch}})$, respectively. On the other hand, if these players select strategy pairs (CH, CH) and (CM, CM) instead, the payoffs will be $(\frac{1}{C_{ch}}, \frac{1}{C_{ch}})$ and zero, respectively. This means that the player who is playing strategy CH does not have an incentive to change the strategy to CM because of receiving less payoffs (i.e., zero). Furthermore, the player who is playing strategy CM does not have an incentive to change the strategy to CH because of receiving less payoffs too (i.e., $\frac{1}{C_{ch}}$). Thus, the strategy pairs (CH, CM) and (CM, CH) are a pure NE for this game according to the definition [70]. □

Proposition 1: For the anti-coordination clustering game for N players, there are N pure NE where the strategy of a single player is to select CH and all the rest of the nodes are to select CM.

The mixed strategy Nash Equilibrium of the clustering game is a probability distribution $\hat{p}$ over the pure NE where each player will have equal expected payoff. Each node will take a random selection conformity with the probability distribution. Let α be the probability of playing CH and $\beta = 1 - \alpha$ be the probability of playing CM. In order to compute these probabilities, we calculate the expected utility function of playing CH as:

$$EU_{CH} = \frac{1}{C_{ch}} \tag{5.29}$$

The expected utility of playing CM is obtained by:

$$EU_{CM} = \frac{1}{C_{cm}} \cdot [1 - (1 - \alpha)^{N-1}] \tag{5.30}$$

According to the definition of mixed NE [75], the expected utilities of playing strategies CH and CM are equal and no player has incentive to change her strategy. Thus,

$$EU_{CH} = EU_{CM} \tag{5.31}$$

Substituting (5.29) and (5.30) in (5.31) and solving the expression in order to calculate the probability α that corresponds to the equilibrium, we get:

$$\alpha = 1 - (\frac{C_{ch} - C_{cm}}{C_{ch}})^{\frac{1}{N-1}} \tag{5.32}$$

The distribution of the mixed strategy NE for the clustering game is $\hat{p} = \{\alpha, \beta\}$ which means that the players will mix their choice for selecting clusterhead strategy and cluster member without incurious about the outcome. However, MSNE is not efficient enough where we could end up with (CH, CH) or (CM, CM) strategies, which is not desirable for the system and could lead to performance degradation of the network.

5.4.3.3.2 Correlated Equilibrium (CE)

We propose a new solution concept, Correlated Equilibrium, for the clustering game that maximizes the outcome and prevents undesirable action. The correlated equilibrium concept is more general than NE and was first proposed by Nobel Laureate Robbert J. Aummann [80].

Thus far, the players' strategies are independent where each player chooses her mixed strategy independently without any communication with each other. According to MSNE solution, all players will gain equal payoffs. However, if the players can avoid ending up with the same

strategies by following an agreement/external signal for the coordination of actions between the nodes, the outcome will be maximized, and efficiency of the system will be higher. The strategy profile is selection according to joint distribution. This results distribution strategy profile called Correlate Equilibrium, where it is best interest for each player to follow the external signal and conform with the recommended strategy. Thereby, the players have no incentive to deviate to gain higher payoff.

The essence of a correlated equilibrium [70] is that when all players follow the external recommendation signal, no player has a unilateral incentive to deviate from the trusted authority's recommendation to achieve higher payoff. Moreover, that signal could be generated by an arbitrator which is seen as a virtual entity and does not depend on the system. The correlated equilibrium is defined as:

Definition of correlated equilibrium [70]: A probability distribution π is a correlated equilibrium of the game G if and only if, for all $i \in N$, $s_i \in S_i$ and $s_{-i} \in S_{-i}$:

$$\sum_{s_{-i} \in S_{-i}} \pi(s_i, s_{-i})[u_i(s'_i, s_{-i}) - u_i(s_i, s_{-i})] \leq 0 \tag{5.33}$$

where $\pi(s_i, s_{-i})$ denotes the joint probability distribution of players. The action for user i and its opponents are s_i and s_{-i}. The inequality (5.33) implies that the expected payoff of player i playing the recommendation strategy s_i at the CE is greater than or equal to the expected payoff that could be received for choosing any other strategy s'_i. In other words, choosing action s'_i instead of s_i cannot obtain a higher expected payoff for user i.

5.4.3.3.3 Linear Programming Solution

For the proposed game, we investigate a linear optimization method to calculate the optimal CE [70],[75],[80]. We drive the CE linear system for 2-player game as shown in Table 5.4, then we implement the same mechanism for N players. A correlated strategy pair in the game

is given by the CE joint probability distribution, which is represented as a 4-dimensional vector $\pi = (p_1, p_2, p_3, p_4)$, where $p_1 + p_2 + p_3 + p_4 = 1$. A correlated strategy pair means that the strategy pair (CH, CH) is played with probability p_1, strategy pair (CH, CM) is played with probability p_2, strategy pair (CM, CH) is played with probability p_3, and strategy pair (CM, CM) is played with probability p_4.

In order to find the egalitarian equilibrium for the game, we formulate the game as linear programming and define the objective function f to find the optimal strategy CE as:

$$f = \max_{p} \sum_{i \in N} \mathbb{E}_p(u_i) \tag{5.34}$$

$$\text{such that} \begin{cases} \forall s_i, s_i' \in S_i, \text{and}, i \in N, \\ p(s_i, s_{-i})[u_i(s_i', s_{-i}) - u_i(s_i, s_{-i})] \leq 0 \end{cases}$$

where $\mathbb{E}_p(.)$ is the expectation over p. Then, the constrains for CE for 2-player game are:

$$u_1(CH,CH)p_1 + u_1(CH,CM)p_2 \geq u_1(CH,CH)p_1 + u_1(CM,CM)p_2 \tag{5.35}$$

$$u_1(CM,CH)p_3 + u_1(CM,CM)p_4 \geq u_1(CH,CH)p_3 + u_1(CH,CM)p_4 \tag{5.36}$$

$$u_2(CH,CH)p_1 + u_2(CM,CH)p_3 \geq u_2(CH,CM)p_1 + u_2(CM,CM)p_3 \tag{5.37}$$

$$u_2(CH,CM)p_2 + u_2(CM,CH)p_4 \geq u_2(CH,CH)p_1 + u_2(CM,CH)p_4 \tag{5.38}$$

By solving the above inequalities, the obvious solution for the CE probability distribution is: $p_1 = p_4 = 0$ and $p_2 = p_3$ which maximizes the sum of the expected payoffs for all players. Thus, the CE joint probability distribution $\pi = (0, p, 1-p, 0)$. Thereby, we have eliminated the

possibility of selecting the same strategy for the players.

For N-player and 2-strategies clustering game, we can derive the linear system and CE constrains according to (5.34) in the same manner for obtaining polynomial time algorithms for optimizing over CE. The number of inequality constraints grow exponentially with the number of players [81]. This result proves that following the external signal is self-enforcing, since cooperation arises naturally from the rules of the game. In addition, it must be considered that the external signal is not binding and players can ignore it. Thus, we guarantee the convergence of equilibrium to CE by utilizing the no-regret learning algorithm discussed in section 5.4.3.5.

5.4.3.4 Fairness and Efficiency (Pareto Optimality)

In this section, we will discuss the fairness and efficiency of all the proposed solution game (i.e., pure and mixed strategy NE compared with CE), as well as evaluate the proposed CE solution by using a concrete example and applying the Pareto optimality concept. Pareto Optimality is the objective measurement of efficiency in game theory.

Table 5.5: An Example of Payoffs Matrix for 2-player

	CH	CM
CH	$\frac{1}{6}$, $\frac{1}{6}$	$\frac{1}{6}$, $\frac{1}{2}$
CM	$\frac{1}{2}$, $\frac{1}{6}$	0 , 0

The two pure strategy NE in the clustering game (i.e., (CH, CM) and (CM, CH)) are unfair where one node always gets higher payoff than the other. However, the MSNE for the game achieves the fairness where the expected utility of the players are equal.

For sake of clarity, let us assume the example of payoffs matrix for 2-players as shown in Table 5.5. The MSNE for the clustering game is the distribution $(\alpha = 1/3, \beta = 2/3)$ over the set of pure strategies. The expected utility for both players will be equal when they mix their strategies

according to MSNE. As per equations (5.29)-(5.32), the expected utility is 0.16. Additionally, the chance of none of the players being a clusterhead $(2/3 \times 2/3 = 44.4\%)$, and the chance of ending up with more than one clusterhead at the same time is $(1/3 \times 1/3 = 11.1\%)$. This means that there is always a high chance of an undesirable action occurring with MSNE (i.e., 55%) either for lossing communication with the base station in the case of absence the clusterhead, or energy wastage in case of more than one clusterhead in the cluster. Accordingly, the MSNE is an inefficient equilibrium to the game. In the same manner, the joint probability distribution of CE for the game shown in Table 5.5 is $(\pi = \{0, 1/2, 1/2, 0\})$ which is calculated by the linear programming (5.34-5.38). The expected utility for the players is $1/2 \times (1/6 + 1/2) = 0.33$, which is greater than the expected utility of MSNE as well as the payoffs of always be a clusterhead.

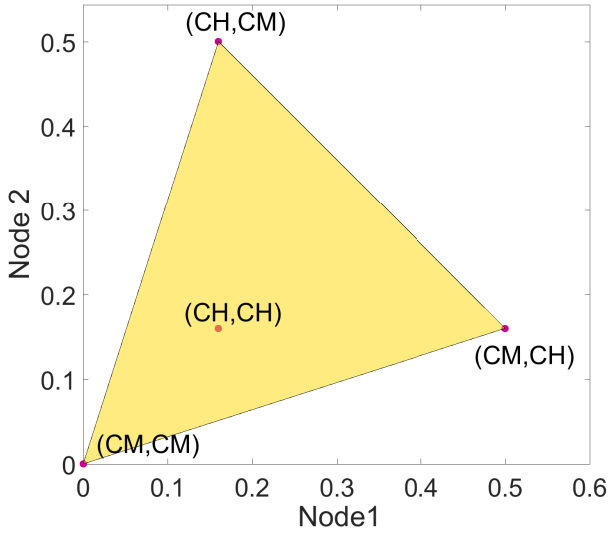


Figure 5.1: Geometrical representation of the set of attainable payoffs under CE for Table 5.5 .

Furthermore, another way to prove the efficiency of the CE is to analyze the Pareto optimality of the solution. The main idea of Pareto optimality is to maximize the outcome of the game where no player can be better off without making some other players worse off. In other words,

an outcome of the game is Pareto efficient if there is no other outcome where a player's utility can be increased without making some other player's utility worse [82]. Figure 5.1 is the convex hull graphical presentation of the game considered in Table 5.5. The 4 points in the figure represent all four possible payoffs. The maximum payoffs attainable by a node must occur at one of the vectors of the convex hull (i.e., (CH, CM) and (CM, CH)), which are the pure Nash equilibria. It can be noticed that the set of Pareto optimal solution is the line between these two payoff vectors, whereas a mixed between these two vectors of expected payoffs is the proposed CE solution for the game.

Therefore, the CE is Pareto optimal (i.e., efficient) solution, where it maximizes the expected utility besides achieving fairness. Moreover, it must be noticed that the CE is less expensive than NE computationally, where computing CE only requires solving a linear program. In contrast, NE requires finding its fixed point completely to solve it.

5.4.3.5 No-Regret Learning Algorithm for CE N-player game

We provide how the strategies of the players reach an equilibrium without needing the trust arbitrators, where the recommended signal is not binding and the players are free to ignore it. In order for the convergence to occur to the set of correlated equilibria in the long run, we use the learning process called regret matching (no-regret) algorithm [83]. The goal of the algorithm is to minimize the regret of each player and reach 0 as time $t \to \infty$. Adjustment of the probability distribution is guided by the average difference (i.e., regret measures) based on the history of the actions that have been played by all players from past periods.

In particular, assume that the game is played repeatedly through time $t \in \{1, 2, 3, 4, ...T\}$, and player i selects the distribution $(p_t)_i$ over action S. Each player in each period decides either to continue playing the same probability distribution $(p_t)_i$ for the next time tick $(t + 1)$ or switch to other probabilities $(p'_t)_i$ that are proportional to the difference "regrets" relative to the current probability. Precisely, for any two distinct actions $s, s' \in S^i$ of player i selecting according to a probability distribution $p^i_t(s)$ and $p^i_t(s')$, respectively, the regret of the player i at time T for not

playing s'_i is calculated as:

$$R^i_t(s_i, s'_i) = max\{D^i_t(s_i, s'_i), 0\} \tag{5.39}$$

where the average difference is given by:

$$D^i_t(s_i, s'_i) = \frac{1}{t}\sum_{\tau=1}^{t}[u_i(s'_i, s^{-i}_\tau) - u^i(s_\tau)] \tag{5.40}$$

For the next period $(t+1)$, the probabilities $p^i_{t+1}(s_i)$ and $p^i_{t+1}(s'_i)$ for player i to take action s_i and s'_i, respectively, are computed as:

$$\begin{cases} p^i_{t+1}(s'_i) = \frac{1}{\mu}R^i_t(s_i, s'_i), \\ p^i_{t+1}(s_i) = 1 - p^i_{t+1}(s'_i). \end{cases} \tag{5.41}$$

where the probability $p^i_{t+1}(s_i)$ is a linear function of regret, and μ is an independent parameter of time and history, and is sufficiently large. Choice of $\mu > 2M^i$ guarantees that the probability of playing the same strategy as in the last period is positive, where M^i is an upper bound on $|u^i(.)|$. In each period, the player selects an action and observes the loss/gain to adjust the probability of choosing an alternative action for higher payoff until the strategies converge to CE. Algorithm 5.2 shows the summary of no-regret learning algorithm.

5.4.4 Strategy Space Reduction for CoPA

Though all the nodes in the network should contribute to the network by serving as clusterhead from time to time, there would always be some nodes that are less suitable to take on the added responsibility. At any point of time, there would be better suited nodes and 'weaker' nodes. The weaker nodes might have less energy remaining, lower transmission capabilities, or lower computing power. Therefore, instead of having *all* nodes participate in the game and exploring the

entire strategy space for finding the equilibrium solution, we argue that certain weaker nodes can safely be excluded for clusterhead consideration.

Algorithm 5.2: Regret-matching (no-regret) learning algorithm

```
Initialization: Set the probability for taking action s_i, ∀s ∈ S_i for the
 node i arbitrarily, p^i_{t=1}(s_i);
begin
    for t = 1, 2, 3, 4... do
        for each node i do
            Calculate payoff u^i_t for playing with probability p^i_t(s_i);
            Find the regret R^i_t(s_i, s'_i) of the player i for not playing s'_i up
             to time t (equations (5.39-5.40));
            Find the probability distribution action for t + 1 (equation
             (5.41)) as;
              1. Update p^i_{t+1}(s_i) to take action s_i
              2. Calculate p^i_{t+1}(s'_i) to take action s'_i
        end
    end
end
```

In order to select the group of nodes that will contribute into the game at any time instance, we consider two system parameters– transmission energy consumption and residual energy, and combine them using a weighed average. If ω_n represents the weighted average of node n, then

$$\omega_n = w_1 D_n + w_2 E_n \tag{5.42}$$

where D_n is the summation of the distances of all neighbours of node n (i.e., $D_n = \sum_{n \in N} \{dist(n, n')\}$), and E_n denotes how much energy the node consumed until the current time. w_1 and w_2 are the weighting factors.

Based on ω_n, it is relatively easy to categorize a node as 'suitable' or 'unsuitable' just by comparing ω_n to some $threshold$ value. As for determining the $threshold$, a simple way would be to use some local cluster parameters, like the mean of the weighted average of all the nodes. Additionally, the $threshold$ is updated periodically and sent to all cluster members by the same

arbitrator (i.e., virtual entity) responsible for generating the external signal for CE solution.

The suitable nodes participate in the repeated clustering game by playing the game in rounds. After each round, all nodes update ω_n and compare with the new $threshold$ for the next round. This exclusion policy has two main features: i) the weighted metric is generic enough and can accommodate any number of node parameters, ii) prohibits unsuitable nodes to participate in the game, thereby reducing the strategy space and speeding up the equilibrium convergence.

5.5 Summary

Designing routing and clustering algorithms that alleviate congestion, and achieve a high energy efficient clustering technique in wireless sensor networks is a challenging problem. Absence of a centralized mechanism to select among available paths unavoidably introduces extra collisions, resulting in reduction of the sensor network lifetime. This chapter formulates the congestion routing issue in WSNs to seek equilibrium solutions and approaches the issue with an evolutionary game theoretical framework. The proposed approach enables independent sensor nodes to evolve a strategy that would ensure long term in distributed manner.

Furthermore, we proposed a cost and payment clustering techniques (CoPA) for wireless sensor networks. CoPA determines the cost of being a clusterhead or a cluster member and provides the probability distribution for the correlated equilibrium. We also proposed a flexible weighted function in order to determine a node's eligibility to participate in the clustering game. In addition, we proved that the correlated equilibrium achieves better performance than the pure and mixed strategy Nash equlibria in term of efficiency and fairness.

CHAPTER 6: A GAME THEPRITIC AOOROACH FOR ATTACKS AND DEFENSE STARTEGIES

6.1 Overview

Most of the network security research focus on either presenting a specific vulnerability or hacking technique, or proposing a specific defense algorithm to defend against a well-defined attack scheme. Although such wireless sensor networks security research is important, few have paid attention to the dynamic interactions between attackers and defenders, where both sides are intelligent and will dynamically change their attack or defense strategies in order to gain the upper hand over their opponents. A secure and trustworthy network system with considering the limitation of the resources constraints is significantly important in WSNs design, where some information is highly sensitive. Therefore, the design of a good defense system must integrate the security features along with the computational aspects. Moreover, it must also consider the resource constraints of networks such that the network is not over-burdened.

In this chapter, we design a network-warfare framework, rooted in game theory, which involves a dynamic interaction between attackers and defenders. A novel approach for a defense mechanism against several types of attacks/threats on WSNs are proposed– a hyper defense approach that considers the limitation of the resources as well as the security value in the network.

In addition, we attain optimal strategies for the defender and the attacker considering that they can dynamically choose their strategies in order to maximize their own payoff based on cost minimization. Generally speaking, we classify the actions of either attacking or defending into three categories: level zero, level one, and level two. The attacker can alternate between these three strategies, where level zero represents no attack, level one represents a low intensity of attack, and level two represents a high intensity of attack. Likewise, we classify the defenders actions into three corresponding defense levels. For level zero, the defender decides to not defend

at all. The second one is a low level of defense, which could cost some of the resources (i.e., energy, or memory space, etc.). The third one is a high level of defense, which requires more computational, battery power, or memory, but gains strong countermeasures against the threats. In practice, the strategies of attackers and defenders for any network security problems could be categorized into more fine-grained levels, but for the sake of clarity and modeling purposes, we believe such a three-level classification of attack or defense is generalized enough and can well represent attack and defense activities in real practice. We emphasize the often-neglected research of the dynamic interactions and evolution among network security attackers and defenders. We present a non-cooperative zero-sum game in modeling the network-warfare between attackers and defenders based on the generalized three-level attack/defense strategies game. We present the case study of three different types of WSNs attacks to demonstrate how the proposed game theoretic framework can be applied in a broad range of network security problems.

The rest of this section is organized as follows: Non-cooperative attack-defense security game is proposed in Section 6.2. We propose various case studies of the attack-defense security game in Section 6.3. Summary are drawn in the last section 6.4.

6.2 Non-Cooperative An Attack-Defense Security Game

This section discusses how an attacker-defender security game is formulated as a non-cooperative zero-sum game. In addition, we describe attacker and defender strategies and derive their solutions. Being rational players in the game, an attacker competes for the best action and his objective is to maximize his own utility. Therefore, the opponents are not bound to cooperate with each other where the malicious attacker would want to play a suitable strategy to maximize his chances of being successful and waste the resources of the system. In contrast, the defender would also like to play a suitable strategy to maximize his chances of protection against the opponents without overspending energy or computation on defending.

As discussed in the literature review, most previous game theory research [58] [6] [60] model attackers and defenders with only two strategies, no attack/defense, or with attack/defense. In order to provide a broader modeling of attackers/defenders where they can adjust their attack-/defense strategies with different intensities, in this chapter, we model each player with three levels of strategies: no attack/defense, low level of intensity, and high level of intensity.

Attackers and defenders experience different cost to benefit affects in order to achieve their success in either attack or defense. Therefore, in our game, each attacker and defender have different levels of strategies instead of having just two levels, as suggested by most of the previous research. In our model, each of the players adopts zero level of intensity, low level of intensity, or high level of intensity.

6.2.1 Game Model

We consider a two-player non-coordination zero-sum security game represented by $\mathcal{G} =< (\mathcal{N}), (\mathcal{S}), (\mathcal{U}) >$, where $\mathcal{N} = \{A, D\}$ represents the two players: Player A is a malicious-node/attacker and the other player D is a defender. $\mathcal{S} = \{a_r, d_r | r \in \{0, 1, 2\}\}$ is the strategy space, which is the set of actions that are available for each player, and their utilities are given by $\mathcal{U}$.

As we mentioned above, the attacker and the defender can use one of the three levels of the available strategies during the game. For the attacker, level zero means that he decides not to attack, denoted by a_0 =*No-Attack*, level one is low intensity of attack, denoted by a_1 =*Attack-1*; and level two is a high intensity of attack, denoted by a_2 =*Attack-2*. Generally speaking, from the attacker's perspective, compared with the strategy *Attack-1*, the strategy *Attack-2* is more effective in generating successful attack, but takes more resources or cost more for the attacker to implement. Correspondingly, level zero for the defender means that he decides not to implement any defense, denoted by d_0 =*No-Defend*; level one is a low intensity of defense, denoted by d_1 =*Defend-1*; and level two is a high intensity of defense, denoted by d_2 =*Defend-2*.

Therefore, the attacker A has three strategies: a_0=*No-Attack*, a_1=*Attack-1*, and a_2=*Attack-2*. The defender D has three strategies as well: d_0=*No-Defend*, d_1=*Defend-1*, and d_2=*Defend-2*. Both players choose their strategies simultaneously without any collaboration, assuming common knowledge about the game (i.e., $\mathcal{U}$)/(gain and lost).

We assume that the value of the protected assets by the defender D is worth of ω_n, where $\omega_n > 0$ and $n \in \{1, 2\}$. ω_1 is the value of assets compromised by *Attack-1* strategy deployed by the attacker successfully; ω_2 is the value of assets compromised by *Attack-2* strategy deployed by the attacker successfully. According to zero-sum game, we assume that the gain of one player is equal to the loss of the opponent. Therefore, ω_n is the gain by the attacker if his strategy Attack-n is successful and $-\omega_n$ denotes the loss/damage by the defender. The value of this loss by defender refers to the degree/amount of damage such as, wasting energy, number of compromised/disabled nodes, loss of data integrity, etc.

Meanwhile, the attacker/defender also needs to make some effort (i.e., pay certain cost) to implement their attack/defense strategies. For the attacker, we denote the cost of attack as c_{an} where $n \in \{1, 2\}$: c_{a1} is the cost to deploy *Attack-1* strategy, and c_{a2} is the cost to deploy *Attack-2* strategy. Likewise, for the defender, we denote the cost of defense as c_{dn} where $n \in \{1, 2\}$: c_{d1} is the cost to deploy *Defend-1* strategy, and c_{d2} is the cost to deploy *Defend-2* strategy.

6.2.2 Model Assumptions

We make the following assumptions for our proposed three-level attack/defense strategy model:

- Value of security assets is always greater than the cost to defend or attack against them since otherwise the defender or the attacker does not have any incentive to defend or attack, respectively; i.e, $\omega_n > c_{an}, c_{dn}, n \in \{1, 2\}$.

- Cost of attack strategy a_1=*Attack-1* is less than the cost of attack strategy a_2 =*Attack-2* for

the attacker. Since *Attack-2* is a more aggressive and effective attack strategy than *Attack-1*, *Attack-2* takes more attacking efforts or cost to deploy. (i.e., $c_{a1} < c_{a2}$).

- Cost of defense strategy d_1 =*Defend-1* is less than the cost of strategy d_2 =*Defend-2* for the defender. Again, this is because *Defend-2* is a more aggressive and effective defense strategy than *Defend-1*. (i.e., $c_{d1} < c_{d2}$).

- Generally speaking, a more aggressive/effective attack will cause more damage to a target if the attack succeeds. Thus based on the definition of ω_n in previous subsection, it is safe to assume that ($\omega_2 \geq \omega_1$).

Table 6.1: Strategic form of Attack-Defense security game.

		Defender (D)		
		d_0	d_1	d_2
Attacker (A)	a_0	0 , 0	c_{d1} , $-c_{d1}$	c_{d2} , $-c_{d2}$
	a_1	$\omega_1 - c_{a1}$, $c_{a1} - \omega_1$	$c_{d1} - c_{a1}$, $c_{a1} - c_{d1}$	$c_{d2} - c_{a1}$, $c_{a1} - c_{d2}$
	a_2	$\omega_2 - c_{a2}$, $c_{a2} - \omega_2$	$\omega_2 + c_{d1} - c_{a2}$, $c_{a2} - c_{d1} - \omega_2$	$c_{d2} - c_{a2}$, $c_{a2} - c_{d2}$

In addition, the game model requires us to define what is the outcome when the attacker deploys one specific attack strategy and the defender implements one specific defense strategy. We make the following assumptions on the game outcomes:

- Attack is successful under these scenarios: *Attack-1* vs. *No-Defend*; *Attack-2* vs. *Defend-1* or *No-Defend*.

- Defense is successful under these scenarios: *Defend-1* vs. *Attack-1* or *No-Attack*; *Defend-2* vs. *Attack-2* or *Attack-1* or *No-Attack*.

- Zero gain or loss when there is no attack and no defense deployed, i.e., *No-Attack* vs. *No-Defend*.

The above assumptions mean that the more aggressive defense strategy, *Defend-2*, is secure against all attacks. However, the low-level defense strategy, *Defend-1*, is good to defend the low-level attack, *Attack-1*, but is still vulnerable to deal with the aggressive attack, *Attack-2*. Table 6.1 illustrates the payoff matrix of the game in a strategic form.

6.2.3 Nash Equilibria Analysis for Non-cooperation Game

For the proposed security game, there is no Pure Strategy Nash Equilibrium (PSNE) where each player in the game always has the incentive to deviate to another strategy in order to gain higher payoff. We can argue that there is no pair of deterministic strategy that works for both players. Therefore, we derive Mixed Strategy Nash Equilibrium (MSNE) for our model. Figure 6.1 illustrates the extensive form of the game.

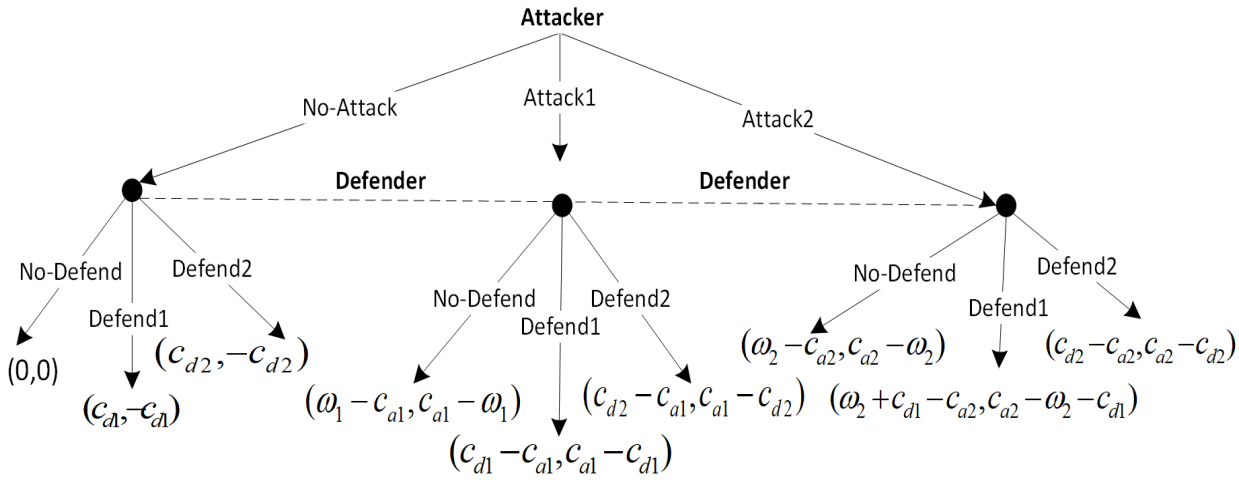


Figure 6.1: Extensive form of the Attack-Defense game.

6.2.3.1 MSNE for Security Game with Three-level Strategies

Definition 1: The Mixed Strategy Nash Equilibrium [75] of the security game is a probability distribution $\hat{P}$ over the set of pure strategies $\mathcal{S}$ for any player such that:

$$\hat{P} = (p_1, p_2, p_3, ..., p_r) \in \mathbb{R}^{\mathcal{R}} \geq 0, \quad \text{and} \quad \sum_{t=1}^{\mathcal{R}} p_t = 1 \tag{6.1}$$

For the attacker, let p_{a_0} be the probability of playing strategy a_0, p_{a_1} be the probability of playing strategy a_1, and $p_{a_2} = 1 - p_{a_0} - p_{a_1}$ be the probability for playing strategy a_2 for the attacker. In the same manner, for the defender let p_{d_0} be the probability of playing strategy d_0, p_{d_1} be the probability of playing strategy d_1, and $p_{d_2} = 1 - p_{d_1} - p_{d_2}$ be the probability for playing strategy d_2.

According to the MSNE definition, the opponents become indifferent about the choice of their strategies by making the expected payoffs equal. Therefore, in our proposed game, the mixed strategy makes each player indifferent among all three of their strategies when the expected utilities from playing strategies a_0, a_1, and a_2 are equal for the attacker, and the expected utilities from playing strategies d_0, d_1, and d_2 are equal for the defender, i.e.,

$$EU(p_{a_0}) = EU(p_{a_1}) = EU(p_{a_2}) \tag{6.2}$$

$$EU(p_{d_0}) = EU(p_{d_1}) = EU(p_{d_2}) \tag{6.3}$$

Then, from Table 6.1, we find the expected utility of the attacker for playing strategy a_0, a_1, and a_2 as function of the mixed strategy which are given by:

$$EU(p_{a_0}) = (p_{d_0})(0) + p_{d_1}(-c_{d1}) + p_{d_2}(-c_{d2}) \tag{6.4}$$

$$EU(p_{a_1}) = (p_{d_0})(\omega_1 - c_{a1}) + p_{d_1}(c_{d1} - c_{a1}) + p_{s_{d2}}(-c_{d2}) \tag{6.5}$$

$$EU(p_{a_2}) = (p_{d_0})(\omega_2 - c_{a2}) + p_{d_1}(\omega_2 + c_{d2} - c_{a2}) + p_{d_2}(c_{d2} - c_{a2}) \tag{6.6}$$

Substituting (6.4), (6.5), and (6.6) in (6.2), we have the probability distribution p_{a_0}, p_{a_1}, and p_{a_2} for the attacker such as:

$$p_{a_0} = \frac{c_{a1}}{\omega_1}, p_{a_1} = \frac{c_{a2}}{\omega_2} - \frac{c_{a1}}{\omega_1}, p_{a_2} = 1 - \frac{c_{a2}}{\omega_2} \tag{6.7}$$

Similarly, the expected utility of the defender for playing strategy d_0, d_1, and d_2 are a function of the mixed strategy which are given by:

$$EU(p_{d_0}) = (p_{d_0})(0) + p_{a_1}(c_{a1} - \omega_1) + p_{a_2}(c_{a2} - \omega_2) \tag{6.8}$$

$$EU(p_{d_1}) = (p_{a_0})(c_{d1}) + p_{a_1}(c_{a1} - c_{d1}) + p_{a_2}(c_{a2} - \omega_2 - c_{d1}) \tag{6.9}$$

$$EU(p_{d_2}) = (p_{a0})(-c_{d2}) + p_{a_1}(c_{a1} - c_{d2}) + p_{a_2}(c_{a2} - c_{d2}) \tag{6.10}$$

Substituting (6.8), (6.9), and (6.10) in (6.3), we have the probability distribution p_{d_0}, p_{d_1}, and p_{d_2} for the defender such as:

$$p_{d_0} = 1 - (\frac{c_{d2} - c_{d1}}{\omega_2} + \frac{c_{d1}}{\omega_1}), p_{d_1} = \frac{c_{d1}}{\omega_1}, p_{d_2} = \frac{c_{d2} - c_{d1}}{\omega_2} \tag{6.11}$$

The mixed strategy NE for the non-cooperation security game is given by the distribution $\{p_{a_0}, p_{a_1}, p_{a_2}\}$, and $\{p_{d_0}, p_{d_1}, p_{d_2}\}$ of equations (6.7) and (6.11) which means that each player will randomize his selection conformity with the probability distribution. Consequently, the opponents

in the game will be indifferent about the outcomes of the play.

6.2.3.2 MSNE for Security Game with Two-level Strategies

In case $c_{a2} << \omega_2$, we could have $p_{a_1} < 0$ according to Equation 6.7, which means that the attacker would need to be putting a negative weight on a_1 strategy to make other player indifferent between his three strategies, and that impossible. On the other hand, this negative probability implies that the attacker has no incentive to deploy the a_1 strategy at all, and has strong incentive to always play a_2 strategy (level 2 of attack) instead of a_1 strategy (level 1 of attack) when he attempts to attack the system in order to maximize his payoff. In contrast, the defender does not have any incentive to play d_1 strategy (level 1 of defense) which will minimize his payoff and cost him more due to the increasing of ω_2. Thus, the two strategies a_1 and d_1 could be eliminated completely from the strategy space. As a result, the game will reduce to 2-strategy for each player with new MSNE.

In case the system is under aggressive attack with very small cost of attacking, the non-coordination zero-sum security game will be reformulated with the new strategy space $\mathcal{S} = \{a_r, d_r | r \in \{0, 2\}\}$. The attacker has two pure strategies: a_0 = *No-Attack*, and a_2 = *Attack-2*. Also, the defender has two pure strategies: d_0 = *No-Defend*, and d_2 = *Defend-2*. Table 6.2 illustrates the payoff matrix of the game with two strategies form.

Table 6.2: Strategic form of the Attack-Defense game with two strategies.

		Defender (D)	
		d_0	d_2
Attacker (A)	a_0	0 , 0	c_{d2} , $-c_{d2}$
	a_2	$\omega_2 - c_{a2}$, $c_{a2} - \omega_2$	$c_{d2} - c_{a2}$, $c_{a2} - c_{d2}$

The distribution $\{p_{a_0},\, p_{a_2} = 1 - p_{a_0}\}$ for the attacker, and $\{p_{d_0},\, p_{d_2} = 1 - p_{d_0}\}$ for the defender are mixed strategy NE for the non-cooperation security game. In this case, each player will randomize his selection of two strategies conformity with the probability distribution and he will be indifferent about the outcomes of the play as well.

In order to compute these probabilities for the attacker, we calculate the expected utility as function of the mixed strategy which are given by:

$$EU(p_{d_0}) = (p_{a_0})(0) + p_{a_2}(c_{a2} - \omega_2) \tag{6.12}$$

$$EU(p_{d_2}) = (p_{a_0})(-c_{d2}) + p_{a_2}(c_{a2} - \omega_2) \tag{6.13}$$

The expected utility of the defender for playing strategy d_0, and d_2 are a function of the mixed strategy which are given by:

$$EU(p_{a_0}) = (p_{d_0})(0) + p_{d_2}(c_{d2}) \tag{6.14}$$

$$EU(p_{a_2}) = (p_{d_0})(\omega_2 - c_{a2}) + p_{d_2}(c_{d2} - c_{a2}) \tag{6.15}$$

As we mentioned above, the expected utilities of playing the two strategies of each player are equal and no player has incentive to change his strategy. Thus,

$$EU(p_{d_0}) = EU(p_{d_2}) \tag{6.16}$$

$$EU(p_{a_0}) = EU(p_{a_2}) \tag{6.17}$$

Then, substituting (6.12), and (6.13) in (6.16), and (6.14), and (6.15) in (6.17) and solving

the expression in order to find the probabilities that correspond to the equilibrium, we get:

$$p_{a_0} = \frac{\omega_2 - c_{d2}}{\omega_2}, p_{a_2} = 1 - \frac{\omega_2 - c_{d2}}{\omega_2} \tag{6.18}$$

$$p_{d_0} = \frac{c_{d2}}{\omega_2}, p_{d_2} = 1 - \frac{c_{d2}}{\omega_2} \tag{6.19}$$

6.3 Case Study of the Attack-Defense Security Game

In this section, we study several types of network attacks and discuss what strategies attackers or defenders can take with minimum resource consumption. Basically, we provide how our proposed game approach can model specific network security problems. According to our attack-defense game model, the attacker can take three different attacking actions. In addition, the defender against the attacker will have three levels of defense strategies as well. In the following subsections, we introduce three concrete attack defense scenarios to illustrate how attack-defense strategies and their dynamic interactions can be modeled via our game theoretic framework.

6.3.1 Defense System Against Hello Flood Attack

Hello flood attack [84] is one of the common attacks in the network layer that a wireless sensor network (WSN) could face, where the attacker will be able to create an illusion of being a neighbor to other nodes or a base station. The hello flood attack can be implemented by an attacking node by sending or replying the hello packets , which are used for neighbor discovery, with significantly high transmission power. This action will convince the nodes in the network that the adversary node is their neighbor.

Attack Strategies

In our security game, the hello flood attacker will play the game by employing one of the two levels of attack in case he decides to attack the system as we mentioned above (i.e., Level one or two). In the low intensity level-one attack, the adversary node sends hello message to sensor nodes and convince them that the adversary is one of their neighbors. Thus, the attacker will behave as a false neighbor node [85].

On the other hand, in the high intensity level-two attack, the adversary node rebroadcasts the received Route Request Packet (RREQ) with high power to a large number of nodes and convinces the nodes that the attacker node is their base station. More specifically, the communication of the sensor nodes with the base station usually occurs through their neighbors. Thus, when the attacker succeeds in creating a false node as base station, and broadcasts a message to all nodes with a high power transmission, the regular node will be confused, convinced that the message came from its neighbor, and assume that this is shortest path from the base station. The adversary in this case can control the entire network through being a false base station [86] [87].

Defense Strategies

In contrast, the defender has one of the two levels of defense against this type of attack. The level-one defense, which is suitable for dealing with the level-one attack, does not require high computational power or battery power to implement. This low level of defense is based on response timing, which is correlated with the transmission distance. There is a predefined time threshold and a normal node should reply a hello message within that time interval. In case the reply message sent by a node is not received in that time by the hello message requesting node then the responding node will be treated as a malicious node [85] [86].

The second level defense strategy is a more advanced detection technique against the aggressive hello flood attack and requires more computational power and battery power than the

level-one defense strategy. The level-two defense strategy could be Signal Strength plus hello message based client puzzles scheme (MBCP) [87]. In this scheme, the nodes are classified as friends according to the signal strength, where each node checks the signal strength of the received hello message with respect to a known reference signal strength. Therefore, if the received signal strength of hello message is the same as the predefined fixed signal strength in the radio range, then the requesting node is a legal node. Otherwise, the node will be classified as a stranger and needs to be further validated. In order to check the validity of a suspicious node, short client puzzles will be used; and with the increasing number of hello messages sent, the difficulty of solving the puzzle will rise as well [87]. Another technique could be applied as a level-two defense for WSN is location verification scheme, which verifies the locations of abnormal nodes by filtering the nodes into normal node or malicious node. The detection of the attack utilizes the greedy filtering by matrix location verification scheme [88]. In summary, the game theoretic strategies of this attack and defense game are as follows:

- **Attacker** a_1 : Behave as a false neighbor
- **Attacker** a_2 : Behave as a false base station
- **Defender** d_1 : Response timing scheme
- **Defender** d_2 : Signal strength and hello message based client puzzles scheme (MBCP); or location verification scheme

6.3.2 Defense System Against Malware Attack

Malware is one of the major threats faced by our cyberworld. It is powerful enough to cause a substantial damage. Throughout the cyber warfare between malware attackers and defenders, malware has evolved with more advanced propagation, compromising, and stealthy techniques, and has be widely used by various attackers to disrupt business operation, steal sensitive information,

gain unauthorized access or any other targeted behavior [89]. In practice, because of the resources limitation in sensor nodes, that restrict their ability to protect their self and own systems, it is easier for the nodes to be compromised by the malware attacks [90].

Attack Strategies

The attacker in the proposed game model can alternate between two different intensities of malware attacks according to his effort and cost of the attacks. The first level attack (i.e., level-one attack) is to generate malware by reusing existing malicious code. Such a malware is easy to produce without requiring significant skill from the attacker, but at the same time it is easy to be detected by signature-based security systems as well.

The second level malware attack (i.e., level-two attack) is more destructive and harder to defend, where the malware is generated by using zero-day vulnerability, or advanced attacking techniques such as polymorphism or metamorphism. Polymorphic malware changes its appearance and creates a countless number of distinct decryptors, and metamorphic malware can automatically re-code itself each time it spreads out by making the best use of obfuscation techniques [91] [92]. By dynamically changing the code format and signature, these advanced attacking techniques make it much harder for defenders to detect a malware.

Defense Strategies

The level-one defense against malware attacks, which is suitable to protect a security system against the level-one malware attack, utilizes the signature-based security system known as static analysis. It relies on its own signature dataset to detect and block recognized malware [91]. Existing signature-based security systems, such as various anti-virus software, as long as they have updated signature database, are fast and effective for fending off level-one malware described above. However, this type of defense will be insufficient against level-two malware attacks where the attacker uses new variants of malware to avoid signature based detection.

Therefore, the level-two defense is the more advanced strategy that has higher requirements on computation power, Internet connectivities, detection response time, and security staff skill/knowledge, etc. This level of defense utilizes dynamic malware analysis techniques, such as Sandbox, to diagnose malware by utilizing a virtual system to analyze the suspected files. The operating principle of this virtual system is to monitor the real running status of a suspicious file, and determine whether or not the file is malicious based on its observed behavior [93] [94] [95]. In summary, the game theoretic strategies of this attack and defense game are as follows:

- **Attacker** a_1**:** Malware generated using existing malicious code
- **Attacker** a_2**:** Malware generated by using zero-day vulnerability, or polymorphic or metamorphic coding techniques
- **Defender** d_1**:** Static Analysis (i.e.,signature-based security system)
- **Defender** d_2**:** Dynamic malware analysis techniques (Sandbox)

6.3.3 Defense System Against Password Guessing Attack

Authentication is an essential element of any security model. Most real-world network systems rely on password for authentication. Authentication of the users in resource constrained network (i.e., WSNs) is one of the major security concerns. A common threat that is used for the authentication of the network for verifying users to get the systems resources, is a password guessing attack, which is a brute force attack that attempts to discover a user password by systematically trying every possible combination of the password [96].

Attack Strategies

The first level attack is a low intensity of password guessing trials that require no skill from an attacker. The attacker will behave as a normal user and send one login attempt one at a time.

This type of password guessing attack is slow in password trial, and hence, could take a very long time for an attacker to discover the correct password.

The second level attack is a high intensity of password guessing trials by utilizing more advanced techniques, such as using the multiple virtual clients scheme [97]. Using such a scheme, an attacker could create many virtual clients from one computing device. These virtual clients behave as completely independent normal users. In this way, an attacker could try many passwords concurrently and thus dramatically speed up the password guessing process.

Defense Strategies

The low level of defense is login throttling scheme. Basically speaking, this scheme limits the frequency of failed login attempts. It can simply put an upper limit on the number of failed login attempts within a given time period, or ask the client to compute the response for a given challenge in order to ensure that the client is not able to launch a large number of password trials in a small amount of time. A large number of password guesses in a small time interval will be eliminated by making password guessing action a time consuming and costly for an adversary [98].

The high level of defense against the level-two password guessing attack described above is intrusion detection system that has efficient detection mechanism and high speed of detection. The defender will be able to determine the true source of attacker's requests by extracting the device fingerprint. "Device fingerprinting is the process of gathering device information to generate device-specific signatures and using them to identify individual devices" [99]. These fingerprints can be extracted from the traffic (transmitted signal) by utilizing an advanced analysis across the protocol stack in order to identity spoofing [99] [100] [101]. In summary, the game theoretic strategies of this attack and defense game are as follows:

- **Attacker** a_1**:** Behave as one normal user and sends one login request at a time.
- **Attacker** a_2**:** Utilize virtual client techniques in order to send many login requests concur-

rently at a time.

- **Defender d_1:** Throttling authentication attempts scheme
- **Defender d_2:** An advanced intrusion detection system that can identify login request real sources (device fingerprint)

6.4 Summary

In this chapter, we proposed a non-cooperative attack-defense security game formulation under different attack situations. In this game, the attacker seeks to inflict the most damage in the network without being detected, while the defender tries to maximize his defending capabilities with a constraint on the limits of the resources. We have proposed a novel hyper defense system which uses the dynamic interaction game model between the attacker and defender to derive equilibrium strategies.

CHAPTER 7: PERFORMANCE EVALUATION

In this chapter, we discuss the performance evaluation of our proposed work. We present the simulation models, experiments, and corresponding results. In order to evaluate the performance of the proposed mechanisms, we conducted extensive simulation experiments in C++ and MATLAB on Windows based platform, and compared with the state-of-the-art. Our intention is to generate and examine various situations that represent the real world scenarios as realistically as possible.

Our simulation study is broadly divided into four parts. Section 7.1 represents the results of the proposed EE-MAC: An Energy Efficient sensor MAC layer protocol. In section 7.2, we propose the results of using ADP: An ADaPtive energy efficient approach in any layer of the networking stack. In section 7.3, we show the results of the proposed routing and clustering mechanisms under game theory frameworks. The results for the proposed dynamic hyper defense technique is discussed in section 7.4.

7.1 EE-MAC Experiment and Results

7.1.1 Simulation Setup

We evaluate EE-MAC: An Energy Efficient sensor MAC layer protocol and compare it with S-MAC in terms of energy consumption and delay. In the simulations, 700 nodes are scattered over a square area, where they remain active for a certain duration $\overline{t_a}$. The sleep times are varied as per exponential distribution with a mean $\overline{t_s}$.

We simulate for both fixed and varying t_s values. Although the sleep times are exponentially distributed in theory, there is an upper bound d_{max} on the time a node can sleep after which it has to wake up irrespective of any triggers in real-life applications. For the combined metric, we use $w_1 = w_2 = 0.5$, i.e., both energy and delay are equally important. As for the energy consump-

tion in active and sleep states, we assume $W_a = 36$ and $W_s = 0.015$ as specified in [9]. Table 7.1 summarizes the simulation parameters.

Table 7.1: Simulation Parameters

Number of Nodes	$100 - 700$
W_a	36
W_s	0.015
w_1	$0.5; 0.1$
w_2	$0.5; 0.9$

7.1.2 Simulation Results

The performance of the proposed protocol is presented in Figures 7.1-7.3. In Figure 7.1, we show how the energy consumption varies with increasing sleep times for a fixed active time ($t_a = 100, 200$, and 300). As expected, the more a node sleeps the less would be the energy consumption. Additionally, with lower active times, energy consumption is also reduced. As shown in Figure. 7.2, the savings in energy due to increased sleep times is offset by the delay degradations. We used two different values for the maximum delay allowed for a node to sleep i.e., $d_{max} = 300$ and $d_{max} = 400$. In Figure. 7.3, the combined utility is given for $t_a = 100, 200$, and 300.

In Figures 7.4-7.7, we compare the performance of EE-MAC with S-MAC. Figure. 7.4 illustrates the energy consumption for EE-MAC and S-MAC for 100 to 700 nodes with $w_1 = w_2 = 0.5$ (same weights for energy and delay). We can see that EE-MAC performs better in energy consumption for smaller number of nodes. However, as the number of nodes increase, the energy savings of EE-MAC also increases accordingly. In Fig. 7.5, we set $w_1 = 0.9$ and $w_2 = 0.1$ to show the effect of varied importance of delay and energy. The results show that the energy consumption in EE-MAC with the new weight values is also less than the energy consumption in S-MAC.

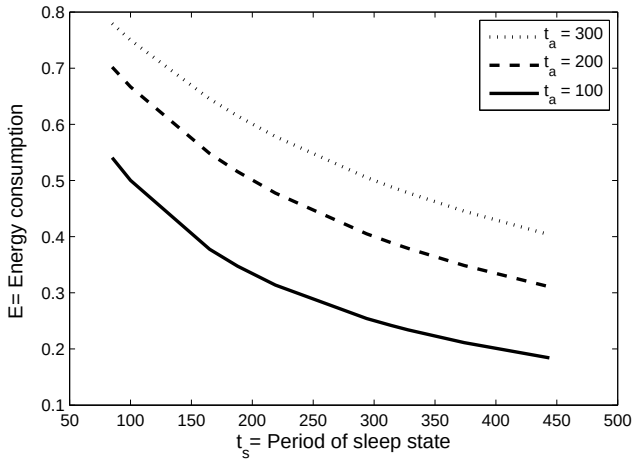


Figure 7.1: Energy consumption vs. sleep times

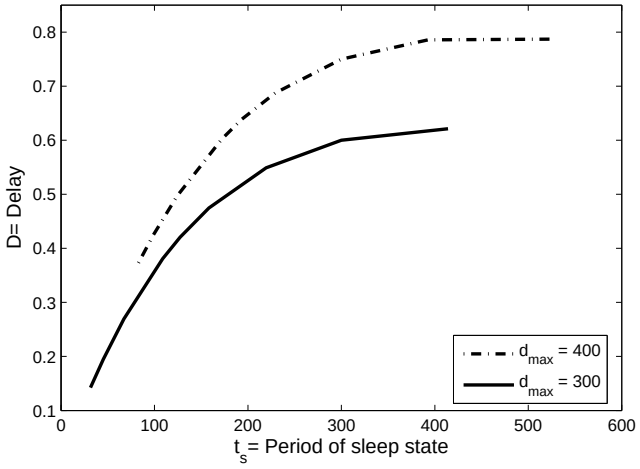


Figure 7.2: Delay vs. sleep times

Figure. 7.6 shows the performances of EE-MAC and S-MAC in terms of delay, for a fixed number of nodes and $t_s = 100$. With high sleep times, EE-MAC performs better as S-MAC is expected to have an inefficient delay performance. The delay performances improve when the average sleep time is reduced. Figure. 7.7 presents the delay performances for $t_s = 20$. Further reduction of ts shows better delay performance for EE-MAC than S-MAC, but with compromised energy savings.

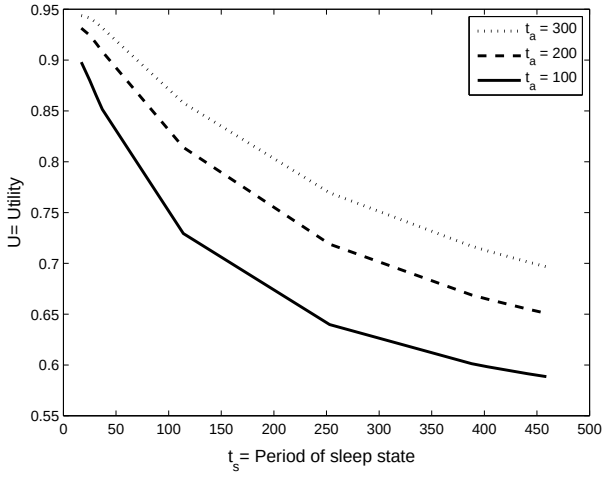


Figure 7.3: Combined utility when $t_a = 100$, $t_a = 200$, $t_a = 300$.

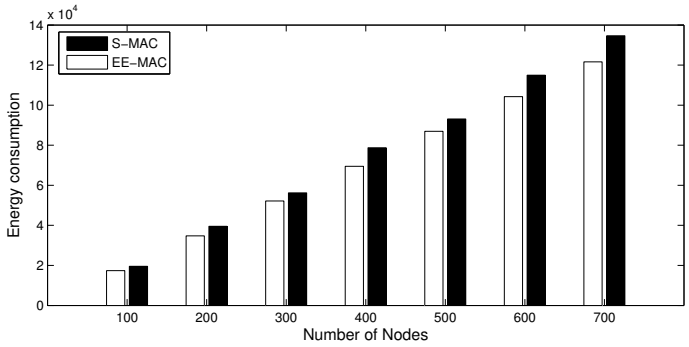


Figure 7.4: Energy consumption for $w_1 = w_2 = 0.5$

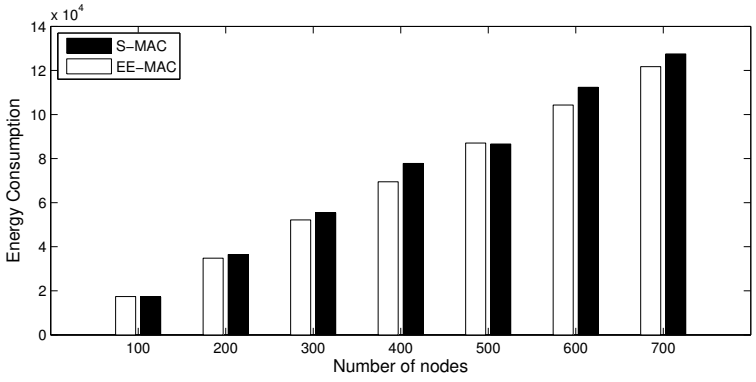


Figure 7.5: Energy consumption for $w_1 = 0.9$ and $w_2 = 0.1$

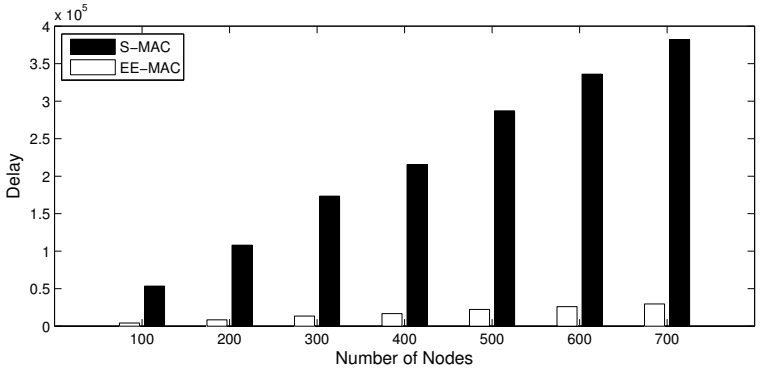


Figure 7.6: Delay with $t_s = 100$

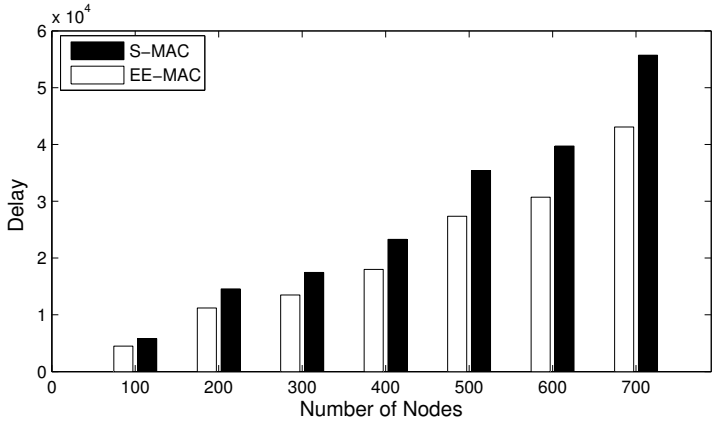


Figure 7.7: Delay with $t_s = 20$

Figure. 7.6 and Figure. 7.7 illustrate that the delay performance of EE-MAC is better than S-MAC for variable sleep times. The results reveal that it is best to have variable sleep times that can be tuned based on the sensing activity and the desired tradeoff between energy and delay. The performance difference between the protocols is more significant for delay than the energy consumption.

7.2 ADP Experiments and Results

7.2.1 Simulation Setup

In this section, we evaluate our ADP approach by comparing it with the base approach which applies the same technique but without adaptation, i.e., the sensor nodes in the base approach have a fixed period of sleeping time. We conduct the experiments that test ADP and the base approach over three different underlying sensing event densities in order to illustrate the impact of underlying dynamic sensing event load on the sensor nodes' behavior. In the first scenario, sensing event occurrence follows a constant rate of Poisson process all the time. In the second scenario, the sensing event Poisson arrival rate λ is increased from λ_{low} to λ_{high} rate in the middle of the simulation. The last scenario is the reverse of the second scenario, where λ starts from λ_{high} and decreases to λ_{low} rate in the middle of the simulation.

We simulate our sensing approach and base approach in Matlab. To be realistic, we use the parameter values of TelosB Mote, a low-power wireless sensor module, as battery energy model as specified in [102]. That is to say, the value of power consumption in the wake-up state is $1.8mA$, and power consumption in sleep state is $5.1\mu A$. We set up 10 nodes and classify the nodes into three groups, i.e., nodes in each group have the same settings and observe/report the same sequence of sensing events. As explained at Chapter 4, sensors in each group will achieve exactly the same scheduling by running ADP independently, as if they synchronize with each other. Node 1 to 3 are in Group 1; Node 4 to 6 are in Group 2; and Node 7 to 10 are in Group 3. In order to get an accurate results, we average the simulation results over 100 runs.

The proposed ADP approach tries to achieve a balanced trade-off between energy saving and data report latency. As shown in the cost function (4.6), the network operator can adjust the relative values of the two weight factors w_1 and w_2 to achieve energy saving while maintaining an acceptable sensing data report latency. For example, by increasing the value of w_2/w_1, the operator can reduce data report latency at the cost of saving less amount of energy. We define the

latency as the time interval between occurrence of sensing event and node wake up to report the event. Furthermore, there is no universal amount of delay that can be defined as acceptable latency because it purely depends on the application and should define by the operator. We assume the maximum acceptable latency range for our simulation is $[4 - 6s]$.

Based on the two performance metrics of remaining battery energy and data report latency, we evaluate system performance from two perspectives: first, by the end performance, which illustrates the behavior of sensor nodes at the end of the simulation time; second, by the temporal performance that shows the behavior of the sensor nodes along their lifetime in three experiments. In Experiment I, the value of sleeping time for the base approach is $t_s = 1/\lambda_{avg}$. In Experiment II, we set different values of sleeping time for the base approach, which is $t_s = 1/\lambda_{high}$, which will make the nodes wake up more frequently. In order to show the nodes' behavior in terms of latency and energy saving while assuming a lower acceptable amount of latency, in Experiment III, we keep the value of sleeping time for the base approach as $t_s = 1/\lambda_{avg}$ but with a higher value of w_2 in order to reduce the latency. The performance of ADP is tested using the following metrics:

- Energy Efficiency: ratio of summation of remaining energy for all nodes divided by the summation of initial energy of all nodes.
- Average remaining energy for all nodes along the simulation time and remaining energy for each node at the end of simulation.
- Average latency for all nodes along the simulation time and average latency for each node at the end of simulation.
- Percentage number of nodes that have less than 20% of energy remaining as compared with their initial energy.

7.2.2 Simulation Results

Impact of Sensing Event Density

For the three different sensing event loads as mentioned above, we measure the energy efficiency and percentage number of nodes that have less than 20% of energy at the end of the simulation for both our proposed ADP approach and the base approach. In the base approach, the value of sleeping time is set to be $1/\lambda_{avg}$. Figures 7.8 and 7.9 show the performance of ADP compared with the base approach at the end of time. Figure 7.8 illustrates the percentage ratio of the number of nodes that have less than 20% of energy level over the three scenarios. This figure shows that all nodes consume more than 80% of their energy at the end of the simulation in the base approach, while 44% of the nodes in ADP still has more than 20% of their energy in the first scenario and 60%, 87% of the nodes still has more than 20% in the second and third scenarios, receptively. Figure 7.9 illustrates the measurement of energy efficiency for ADP and the base approach. Compared with the base approach, ADP achieves a higher energy efficiency for all cases. As the sensing event load changes in the second and third scenarios, ADP still has a higher energy efficiency than the base approach.

Energy Saving and Latency

This section represents the results of three experiments , as mentioned above, over the second scenario of underlying sensing events; in this second scenario the sensing event Poisson arrival rate λ is increased from λ_{low} to λ_{high} rate at the middle of the simulation. The results show the performance of ADP and the base approach. The following figures 7.10, 7.11, and 7.12 demonstrate the behavior of sensor nodes for saving energy and latency over a dynamic changing underlying sensing event load. Figure 7.10 represents the performance of ADP and the base approach when the value of fixed sleeping time in the base approach is $1/\lambda_{avg}$. The percentage of remaining energy and the amount of latency for each node at the end of simulation are showed in figure 7.10(a).

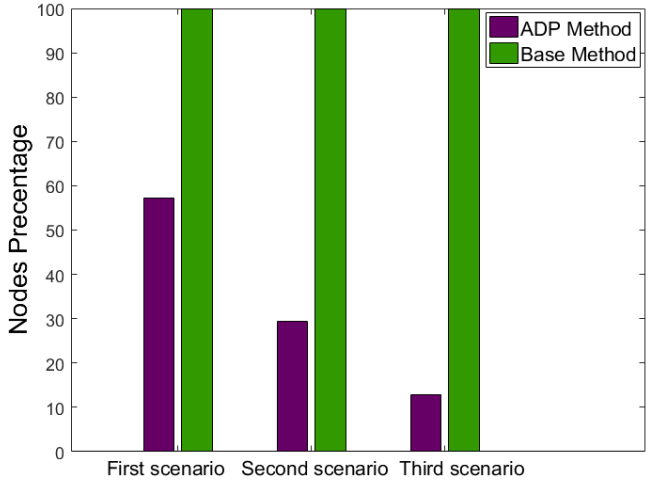


Figure 7.8: Percentage number of nodes that have less than 20% of energy.

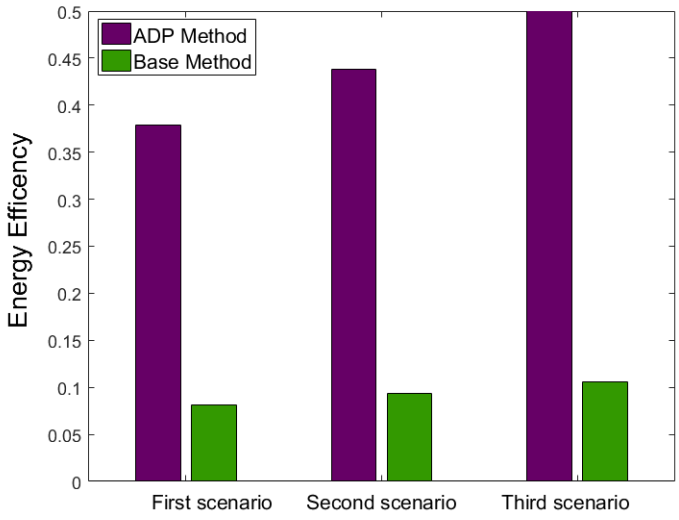
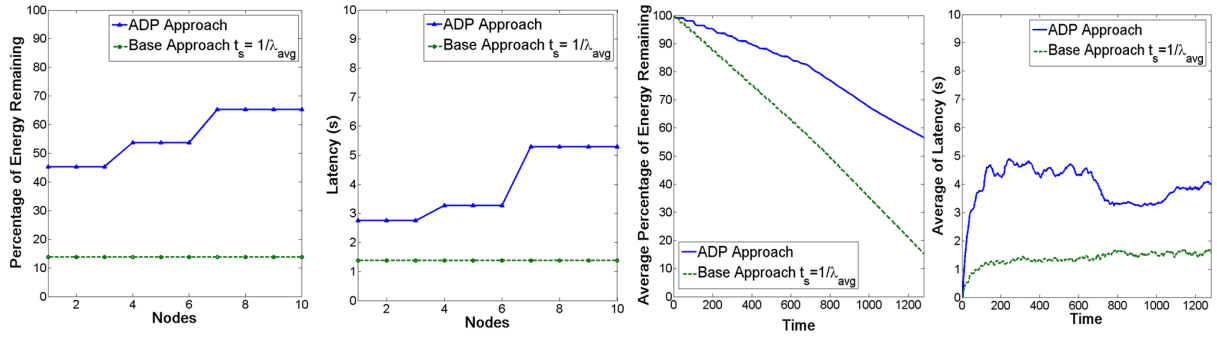


Figure 7.9: Energy Efficiency = ratio of the sum of nodes remaining energy to the sum of nodes initial energy.

To study the impact of factor c, the importance of reporting data, the first group of Node 1 to 3 has the largest value of c, while the third group of Node 7 to 10 has the smallest value of c. The proposed ADP approach makes nodes consume less energy than the nodes in the base ap-

proach, and the latency stays low within the acceptable range set. Figure 7.10(b) demonstrates the measurement of the average of percentage of remaining energy and latency for nodes throughout the simulation. The graph of average percentage of remaining energy shows the amount of energy saved in ADP is more than that in the base approach by 40%. When the sensing event density changes in the middle of simulation, energy consumption rate in the ADP approach also changes correspondingly. In addition, the average of latency in our approach decreases in the middle of simulation according to the density change of sensing events.



(a) Energy and latency at the end of simulation time.

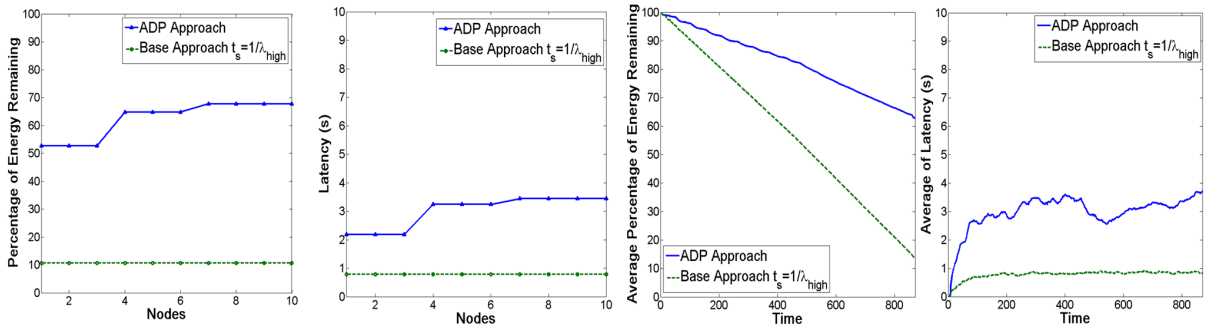
(b) Average of energy and latency during the simulation time.

Figure 7.10: Performance of ADP and the base approach in Experiment I.(when the value of fixed sleeping time t_s for base approach is $1/\lambda_{avg}$.) ADP gains a high amount of energy saving and keeps latency well below the acceptable latency. In (b), change the middle of the curves refer to the density change of sensing event.

Figure 7.11 illustrates the experiment's results when the value of t_s in the base approach is $1/\lambda_{high}$. Figure 7.11(a) and 7.11(b) represent the percentage of remaining energy and latency for each node at the end of the simulation, and the average of remaining energy and latency throughout the simulation, respectively. In this instance, we notice that our approach also achieves high performance for saving energy by 45% and keeps latency under the maximum acceptable latency.

To show the nodes' behavior in term of energy saving and latency with more emphasis on reducing the latency than on energy saving, we test ADP and the base approach by changing the

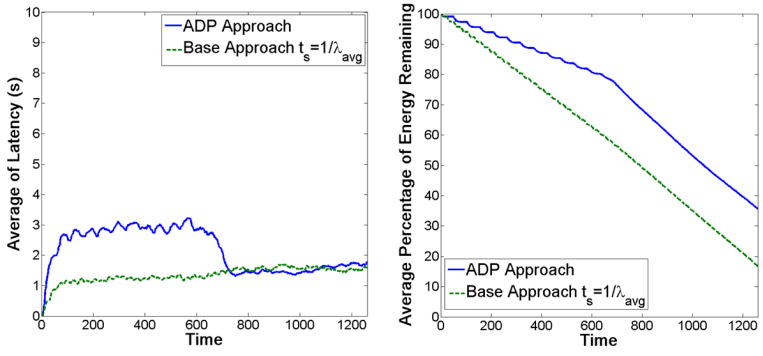
parameters of weighted factors in the experiment III. The results of the average energy remaining and latency are represented in Figure 7.12, and follow the same trend as the previous experiment for saving energy. The figure shows that our approach saves 15% energy when compared to the base approach. It achieves a good improvement in latency, and it could even achieve less latency when compared with the base approach.



(a) Energy and latency at the end of simulation time.

(b) Average of energy and latency during the simulation time.

Figure 7.11: Performance of ADP in Experiment II. (The results similar to Fig. 7.10 when the value of fixed sleeping time t_s for base approach is $1/\lambda_{high}$ instead of $1/\lambda_{avg}$ (waking up more frequently).) ADP also gains a higher amount of energy saving.



(a) Average of energy during time.

(b) Average of latency during time.

Figure 7.12: Performance of ADP in Experiment II. (The results similar to Fig. 7.11 when the value of fixed sleeping time t_s for base approach is $1/\lambda_{high}$ instead of $1/\lambda_{avg}$ (waking up more frequently).) ADP also gains a higher amount of energy saving.

7.3 Routing and Clustering Experiments and Results

7.3.1 An Evolutionary Routing Game Algorithm

In order to analyze and study the effects of applying the proposed evolutionary routing game model for multiple routes in a wireless sensor network, we have conducted simulation experiments. We study the behavior of selecting strategies when sensor nodes do not cooperate with each other, and how the hop selection strategies converge into evolutionary stable states. The empirical analysis of our evolutionary routing game consists of three aspects: First, we will demonstrate the results of our experiments in which sensor nodes have only two available hops to transmit the data packets, show the impact of implementing Replicator Dynamics, and how the strategies converge to an evolutionarily stable state. Second, we will present the results of simulation under dynamic network conditions, and show that the evolutionary game is able to converge to a new ESS. A diversity of wireless network conditions will result in different transmitting costs. Node failure due to changing condition can occur for various reasons, such as uncontrolled environment, battery depletion, or a communication failure. Node failure will in turn result in the changes of the cost of routing paths. Also, the mobility of the nodes in a WSN is another possible cause for the dynamic changes of the cost of routing paths. Finally, we will provide several experimental results with multiple hops available (i.e., 3 and 4 heterogeneous hops) as well.

Figures 7.13 and 7.14 represent the scenario of having 2 hops available to forward the data packet. Figure 7.13(a) shows the behavior of selecting one of two available hops with some probability where a transmission through hop 1 produces a lower cost than a transmission through hop 2. The probabilities of selecting the hops are modified depending on average fitness, which is gained from strategic interaction in subsequent time slots as shown in Figure 7.14(a). Moreover, in our simulation, any positive value for the utility function would be commutable and feasible. In Figures 7.13 and 7.14, the cost function of selecting the hops are assumed to be ($u_{1\mathcal{A}} = 0.5$ & $u_{2\mathcal{A}} = 0.25$) and ($u_{1\mathcal{B}} = 0.166$ & $u_{2\mathcal{B}} = 0.125$) for hops 1 and 2, respectively. MSNE is

$\grave{p} = \{0.57, 0.43\}$ and $\grave{q} = \{0.66, 0.33\}$ for population $\mathcal{A}$ and $\mathcal{B}$, respectively.

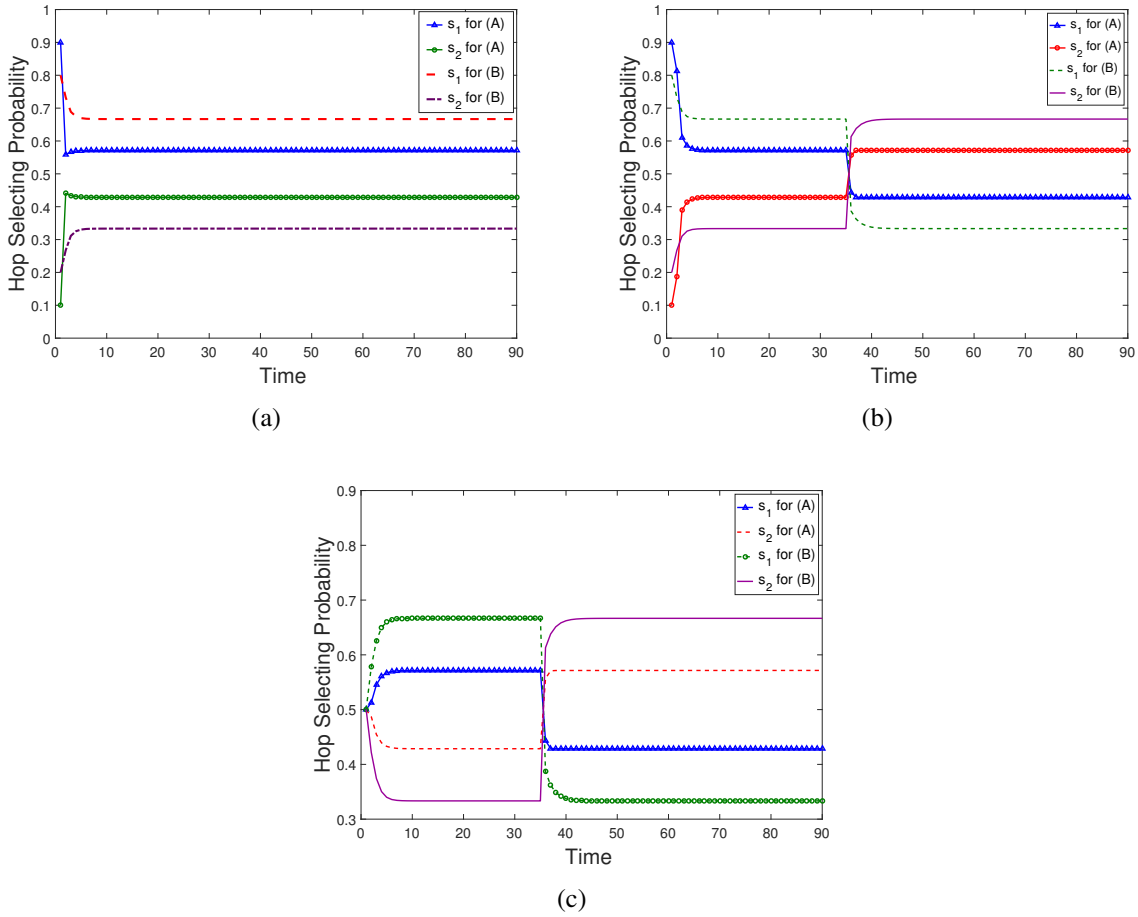


Figure 7.13: Proportion of selecting strategies for both population (i.e.,$\mathcal{A}$ & $\mathcal{B}$) when number of available hops for forwarding is $R = 2$. (a) Hop selecting probability when the initial probabilities are unequal. (b) Hop selecting probability when the initial probabilities under changing conditions, (i.e., cost of forwarding through hop 1 higher than hop 2 at t=350, when initial probabilities are unequal, and (c) when initial probabilities are equal.

First, let us consider the scenario where some sensor nodes become greedier and transmit the packet with a lower cost through hop 1. Thus, the payoff for those nodes who adopt strategy s_1 at $time = 1$ is less than the payoff for selecting hop 2, as demonstrated in Figures 7.13(a) and 7.14(a). This is because forwarding through the lower cost hop by more nodes results in collisions

and thus gain a zero payoff. As a result, the hop selecting probability of greedy nodes decreases in $time = 2$ (as shown in Figure 7.13(a) and their payoff increases at that time, which is still less than the average payoffs of the entire population as shown in Figure 7.14(a)). In a similar yet opposite scenario, the nodes that are less greedy and transmit through hop 2, which costs more for transmitting, receive a higher payoff at $time = 1$ than the nodes transmitting through hop 1.

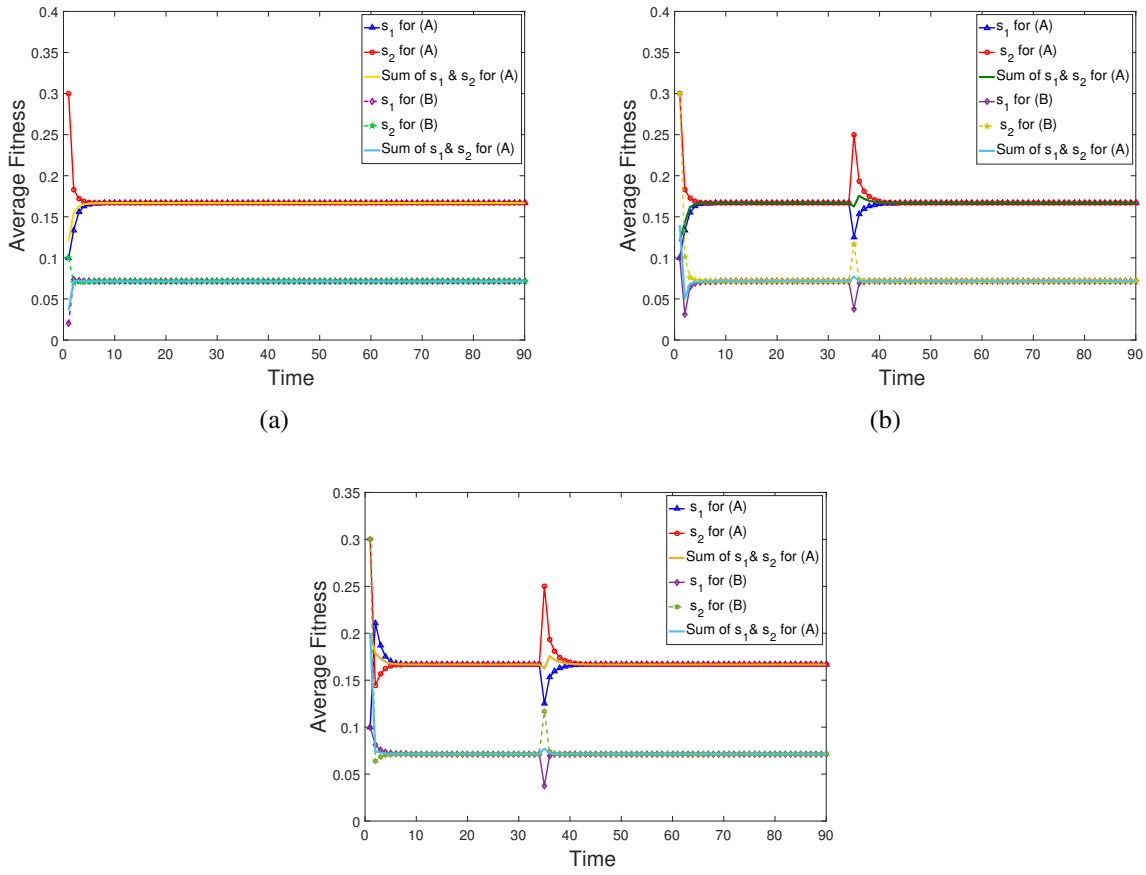


Figure 7.14: Related Average fitness of selecting strategies in Fig. 7.13 for both population (i.e.,$\mathcal{A}$ & $\mathcal{B}$) when number of available hops for forwarding is $R = 2$. (a) Average and weighted sum of fitness when the initial probabilities are unequal. (b) Related average fitness under changing conditions (i.e., cost of forwarding through hop 1 higher than hop 2 at t=35) when initial probabilities are unequal, and (c) when initial probabilities are equal.

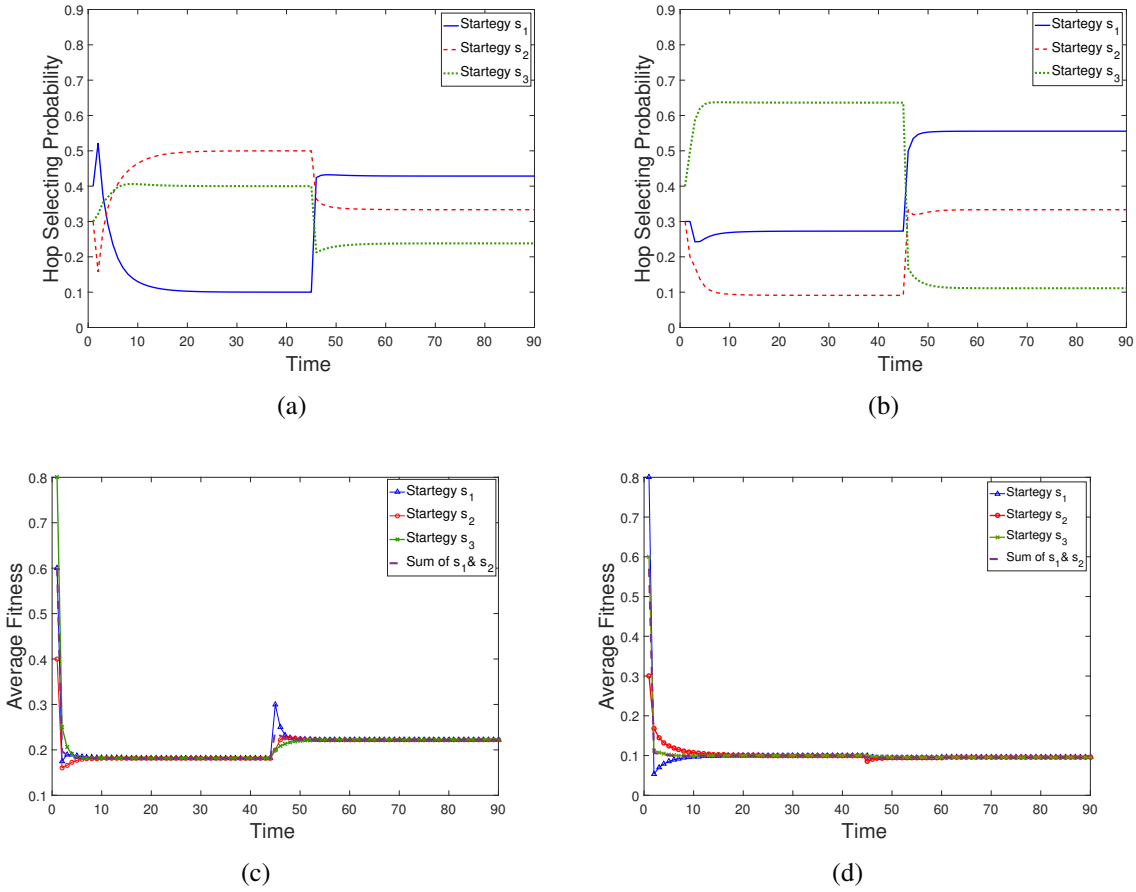


Figure 7.15: Proportion of selecting strategies and related average fitness for both population (i.e.,$\mathcal{A}$ & $\mathcal{B}$) when number of available hops for forwarding is $R = 3$. (a) and (b) Hop selecting probability when under changing conditions and initial probabilities are unequal for population $\mathcal{A}$ and $\mathcal{B}$, respectively. (b) and (d) Related average.

Moreover, this causes the hop selecting probability to increase in the following time for the less greedy nodes and decreases their payoffs. In a similar manner, the hop selecting probability is modified until the system becomes stable and reaches the ESS, (i.e., time=10 in the case of figure 7.13(a)). The amount of time taken to converge to ESS is important in determining energy wastage in sensor networks due to the collision and lost the data. Figures 7.13(b) and 7.14(b) demonstrate the case of changing network conditions, where the cost of transmitting through hop 2 becomes less

than through hop 1, and hop 2 becomes more preferable to be selected from the nodes at $t = 35$. The hop selecting probability still converges to a new ESS. Similar observations of convergence to ESS can be found in the case where initial hop selection probabilities are equal and the network conditions are changed as shown in Figures 7.13(c) and 7.14(c).

The previous experiment (i.e., figures 7.13 and 7.14) demonstrated that the fairness of probability distribution of selecting the two hops are achieved only when the probability of selecting the two hops equals $p_1 = 0.57, p_2 = 0.43, q_1 = 0.66$, and $q_2 = 0.33$ as in figure 7.13(a), for both population, respectively, which is the game's MSNE as well as the ESS. Next, in order to present the robustness of our game, we conduct the experiment under changing network condition and with equal and unequal initial probabilities for the player as well. The results show that the strategies are still able to converge to ESS as shown in figures 7.13(b),7.14(b),7.13(c) and 7.14(c).

Figures 7.15 and 7.16 exhibit the performance of the system and the convergence of hop selection probabilities to ESS in case of multi-hops (i.e., 3 hops), where each hop has a different transmitting cost for each population ($\mathcal{A}$ and $\mathcal{B}$). Moreover, Figures 7.15 and 7.16 show the behavior of nodes when the network conditions changed (i.e., changed at the time $t = 45$) in our proposed evolutionary game, and when the initial probabilities are unequal and equal, receptively. Figures 7.15(a), 7.15(c), 7.16(a), and 7.16(c) show the convergence probabilities of selecting 3 hops to ESS and related average fitness by population $\mathcal{A}$. Figures 7.15(b), 7.15(d), 7.16(b), and 7.16(d) show the convergence probabilities of selecting 3 hops to ESS and related average fitness by population $\mathcal{B}$. For example, at the beginning in figure 7.15(a), the game converges to ESS for population $\mathcal{A}$ when hop 2 is more preferable to be selected from the nodes and the initial values for utility of selecting s_1, s_2, and s_3 are 0.2, 0.9 and 0.5, respectively. At $time = 45$, the network conditions are changed: Hop 1 becomes more attractive for the sensors and adopting s_1 will produce higher payoff than selecting s_2 or s_3. The initial values for utility of selecting s_1, s_2, and s_3 are changed to 0.5, 0.3 and 0.2, respectively. Similarly in Figure 7.15(b), the network conditions are changed with different utility values for each strategy selection. The system reaches stability

under new network conditions and converges to a different ESS for all populations.

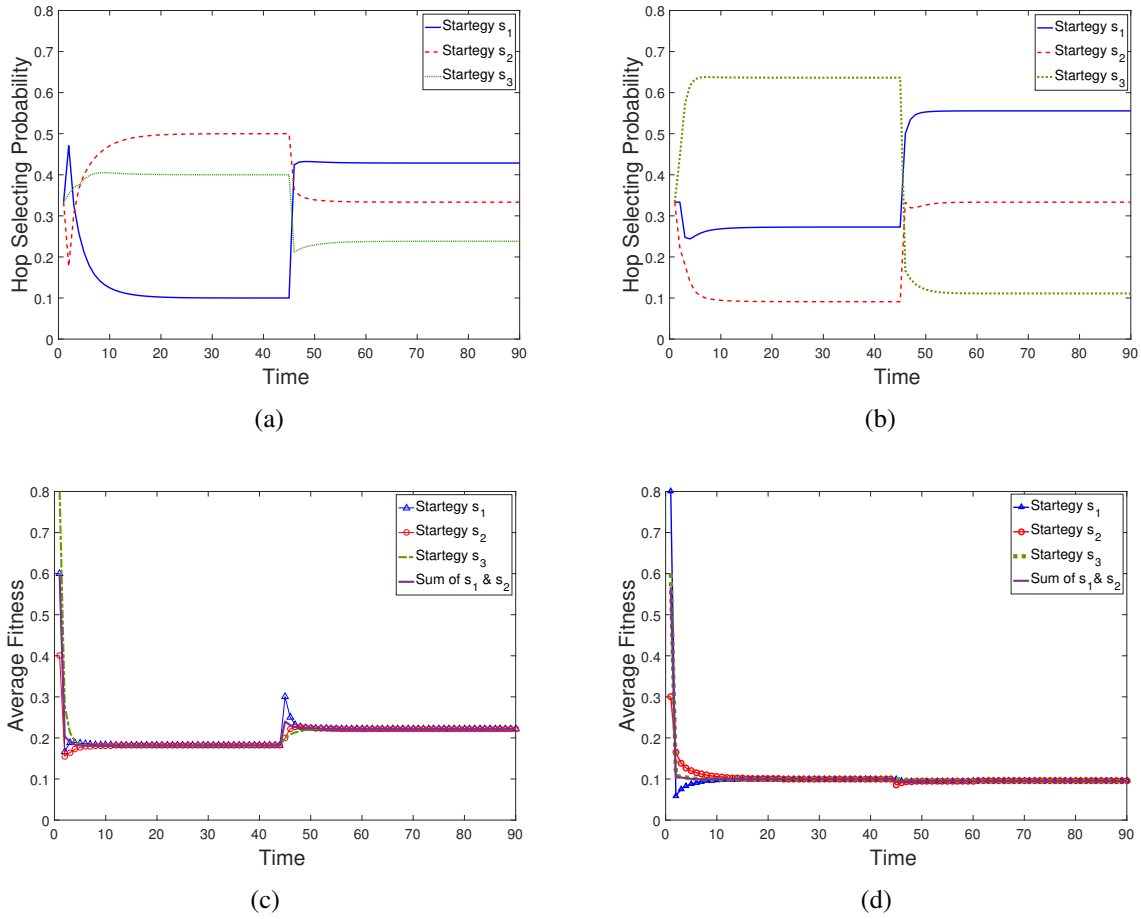


Figure 7.16: Proportion of selecting strategies and related Average fitness for both population (i.e.,$\mathcal{A}$ & $\mathcal{B}$) when number of available hops for forwarding is $R = 3$. (a) and (b) Hop selecting probability when the initial probabilities under changing conditions and initial probabilities are equal for population $\mathcal{A}$ and $\mathcal{B}$, respectively. (b) and (d) Related average.

Figure 7.17 shows the convergence of hop selection probabilities to ESS in case of having 4 hops available, and their utilities are varied according to transmitting cost. Figures 7.17(a) and 7.17(b) illustrate the converges to the ESS for population $\mathcal{A}$ and $\mathcal{B}$, respectively. We notice that the rate of convergence to ESS is affected by the number of hops, variety of the transmitting cost, and the initial access probabilities of players, where the convergence rate to ESS decreases when the

number of hops increases. Figures 7.17(c) and 7.17(d) illustrate the converges to the ESS under new network conditions for population $\mathcal{A}$ and $\mathcal{B}$, respectively. As a result, the system will be able to reach stability with 2 and multi-hop of different transmitting costs, even under the changing of network conditions and with varied values of initial access probabilities.

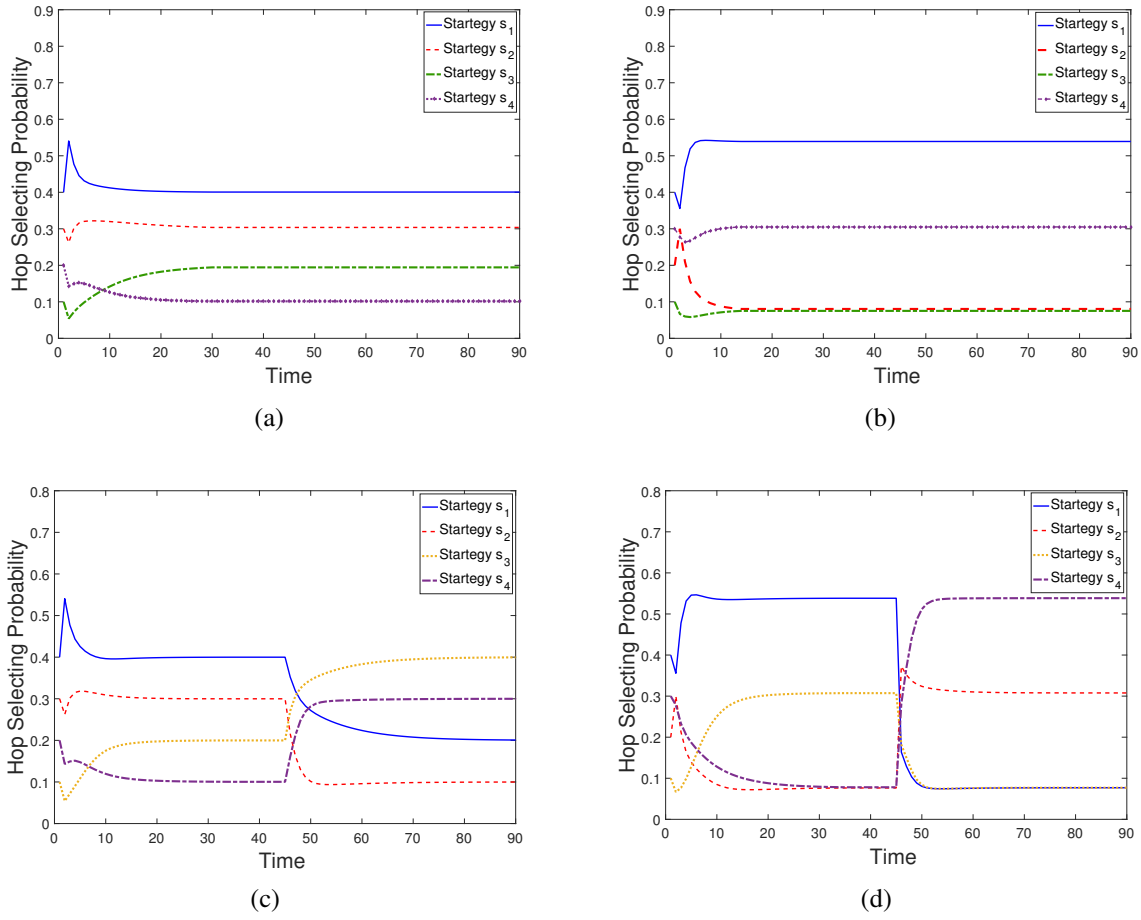


Figure 7.17: Proportion of selecting strategies for both population (i.e.,$\mathcal{A}$ & $\mathcal{B}$) when number of available hops for forwarding is $R = 4$. (a) and (b) Hop selecting probability for population $\mathcal{A}$ and $\mathcal{B}$, respectively. (b) and (d) under changing condition of network (i.e., $t = 45$).

7.3.2 Energy Efficient Clustering Algorithm

In order to test the veracity of the CE in determining the clusterhead set in the proposed COPA for WSN, we resort to simulation experiments. We simulate a system of N sensor nodes using MATLAB. In order to measure the performance of our clustering algorithm, we compare it with two other well-known clustering techniques: probability-based [103] and CROSS [13]. The probability of being a clusterhead is fixed in the probability-based, and is set to 0.05 as in [103]. The probability of being a clusterhead in CROSS is defined as $p = 1 - \omega^{\frac{1}{N-1}}, 0 < \omega < 1$; where the value of ω is set as per [13]. For CoPA, the probability of being a clusterhead or a cluster member depends on the CE probability distribution for the clustering game as presented in Chapter 5 (Section 5.4.3). We assume that the base station is located outside the sensing field. The sensor nodes form a connected network i.e., we get a single component graph.

The rest of the simulation parameters are presented in Table 7.2. Furthermore, we identify three metrics that reveal the performance of any clustering technique: network lifetime, average residual energy, and amount of data sent to the base station (throughput).

Table 7.2: Simulation Parameters

Parameters	**Value**
Initial energy	0.5 J
Transmit and receive energy	50 nJ
Transmit to the base station	100 nJ
Data aggregation energy	5 nJ

In order to show the relative performance of exclusion policy of CoPA, Figure 7.18 presents the number of nodes that contribute to the game for a various number of sensor nodes (i.e., $N = 20, ..., 140$). Because of the exclusion policy (Section 5.4.4), we notice that the average number of participated nodes is less than the total number of sensor nodes (i.e., $55\% - 65\%$). Therefore, the strategy space will significantly reduce and the equilibrium convergence will speed up.

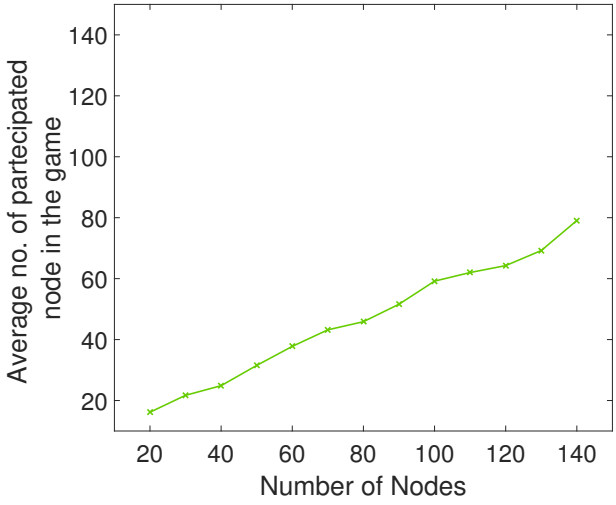


Figure 7.18: Number of nodes that participate in our proposed clustering game (CoPA).

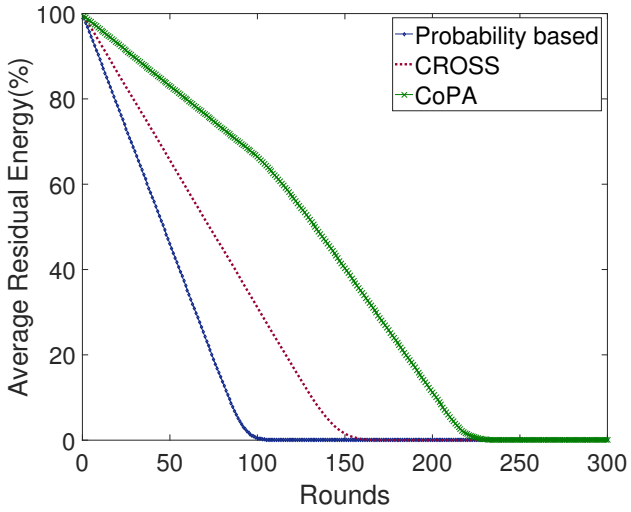


Figure 7.19: Average residual energy.

Figure 7.19 shows the average residual energy of the sensor nodes for the three clustering methods. The number of nodes considered in this experiment was 50. For the probability-based and CROSS clustering, the average residual energy for the nodes drops to almost 0 in 100 and

150 rounds, respectively. CoPA on the other hand has a steadier energy degradation. Figure 7.20 exhibits the network lifetime for various number of sensor nodes (i.e., $N = \{40, ...140\}$) for the probability-based, CROSS, and CoPA. We define network lifetime as 'the lifespan of the first node in all sensor nodes that depletes its energy' [13]. We consider a node's energy is exhausted when 99% of the sensor's initial energy has been consumed. CoPA achieves a longer lifetime than the other two for any numbers of nodes.

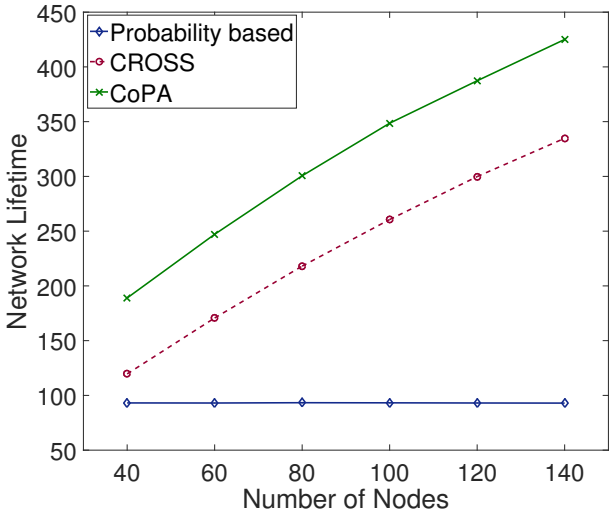


Figure 7.20: Network Lifetime.

We also measure the throughput of the system according to the amount of data sent to the base station, where the only way to reach the base station is through the clusterheads. In the absence of any clusterhead, the data cannot be relayed to the base station. In Figure 7.21, we present the amount of data that was sent to the base station. The simulation results show that CoPA has the highest value, which is 5% and 20% more than the probability-based and CROSS, respectively. Consequently, CoPA ensures of determination of clusterheads in each round and guarantees a pathway for the sensed data to be sent to the base station. As a final comment, the absence of clusterhead could occur continuously in the probability-based and CROSS because of

their dependence on the node's probability for playing as a clusterhead, whereas CoPA guarantees of the existence of clusterheads in every round till the network dies.

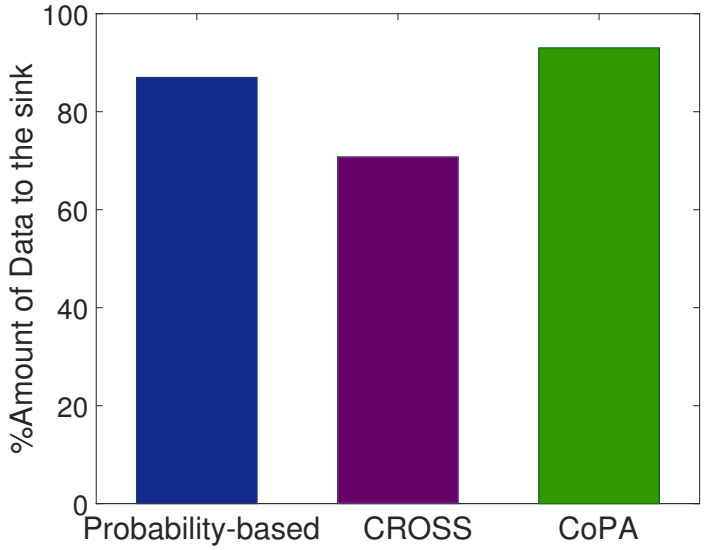


Figure 7.21: Amount of data sent to Base Station.

7.4 Attack and Defense Experiments and Results

7.4.1 Simulation Setup

In this section, we have simulated our proposed defense approach (i.e., hyper defense) for wireless sensor network scenario and compared it with two "always-on" constant defending systems in order to validate the performance of our model. The first constant defending system employs the low intensity (level-one) defense all the time, and the second constant defending system employs the high intensity (level-two) defense all the time as well. We assume that all the nodes in the network have the same initial battery energy in the beginning of the game. We also take into account a network where the nodes consider the battery life as the priority requirement, and

where the nodes defend against resource consumption attacks. The attacker aims at attacking the network and destroying/reducing the lifetime of the network. In such circumstances, the security value ω may be represented by the conserved energy by success defense action. The attacker and defender will play the game according to the equations in Chapter 6 (section 6.2.3).

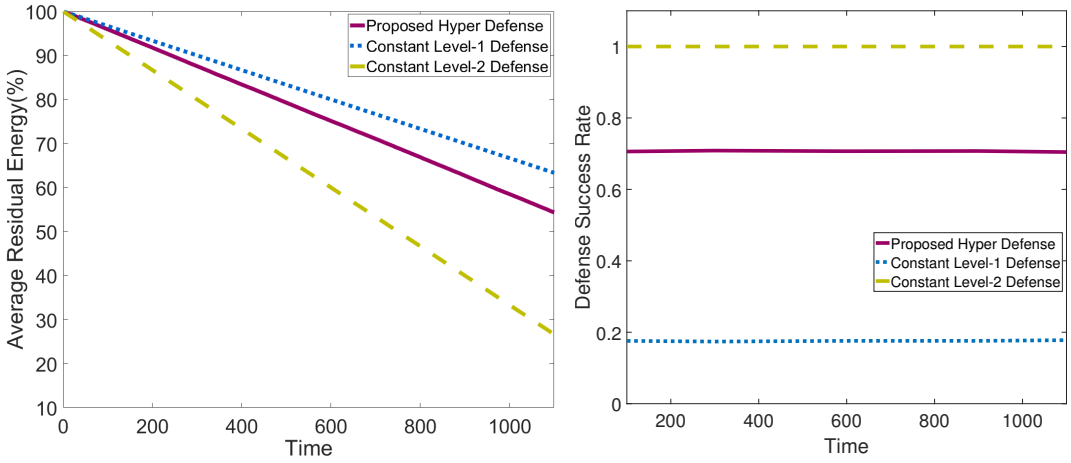
The proposed attack-defense model (i.e., hyper defense) tries to achieve a suitable defense strategy for the system as well as to consider the limitation of the resources. We evaluate system performance by identifying two metrics: average residual energy, and defense success rate. Furthermore, we consider the variety of security value ω_n compared to the cost of attack c_{an} and cost of defense c_{dn} in order to show the impact of this variable ω_n on the performance of the model in two experiments.

7.4.2 simulation Results

In the first experiment, we assume that the security value ω_n is higher than the attacking and defending cost (i.e., $\omega_n > c_{an}$ and $\omega_n > c_{dn}$) while considering the variety of the attack and defense cost as illustrated in Figures 7.22, 7.23, and 7.24. In Figure 7.22, the cost of attack and defense are assumed to be equal (i.e., $c_{an} = c_{dn}$). In Figure 7.23, the attacking cost is assumed to be less than the defending cost (i.e., $c_{an} < c_{dn}$). Inversely, the cost of attack is assumed to be higher than the cost of defense (i.e., $c_{an} > c_{dn}$) in Figure 7.24. The proposed hyper defense achieves a higher percentage of average residual energy than the constant level-2 defense. In the proposed hyper defense, the defender still has $55\%, 40\%$, and 58% of the energy in the three scenarios (i.e., Figures 7.22(a), 7.23(a), and 7.24(a)) of different defending/attacking cost, respectively. However, the defender has $29\%, 18\%$, and 28% of the energy in the constant the constant level-2 defense as shown in Figures 7.22(a), 7.23(a), and 7.24(a), respectively.

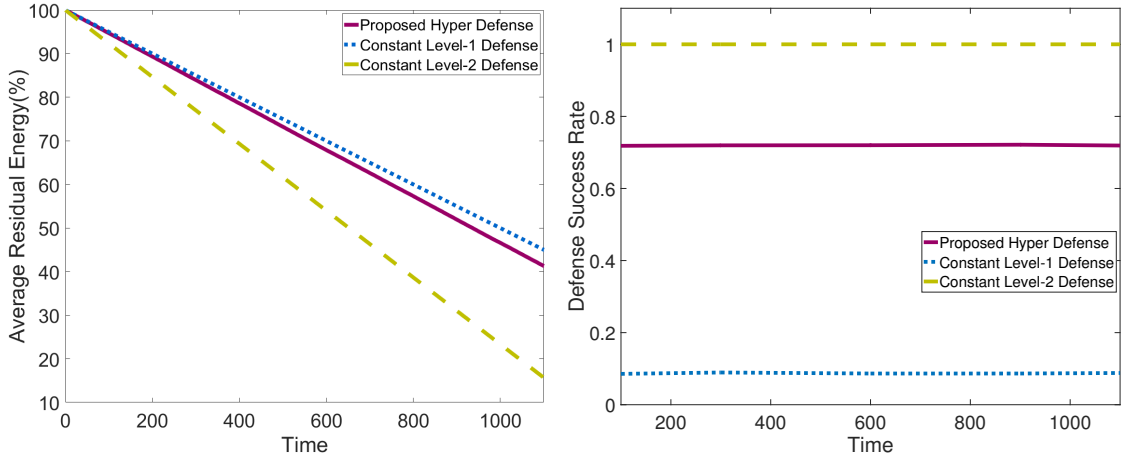
In addition, the constant level-1 defense consumes less amount of energy, but we notice that the defense success rate is too low compared with our proposed model. The hyper defense produces a good defense success rate (i.e., 0.7, 0.7, and 0.8) as illustrated in Figures 7.22(b),

7.23(b), and 7.24(b), respectively, compared with the constant level-1 defense as well as achieving a higher residual energy compared with the constant level-2 defense approach.



(a) (b)

Figure 7.22: Average residual energy and defense success rate when ω is higher than c_{an} and c_{dn} (i.e, $\omega_n > c_{an}, c_{dn}, n \in \{1, 2\}$), and the cost of attack and defense are equal (i.e., $c_{an} = c_{dn}$), respectively.

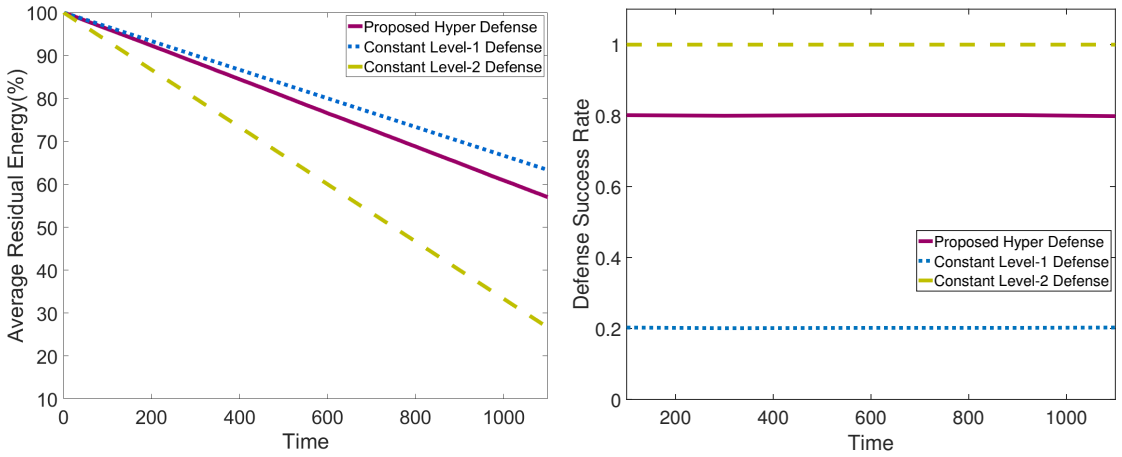


(a) (b)

Figure 7.23: Average residual energy and defense success rate when ω is higher than c_{an} and c_{dn} (i.e, $\omega_n > c_{an}, c_{dn}, n \in \{1, 2\}$, and $c_{an} < c_{dn}$)

In the second experiment, we consider the diversity of security value compared with at-

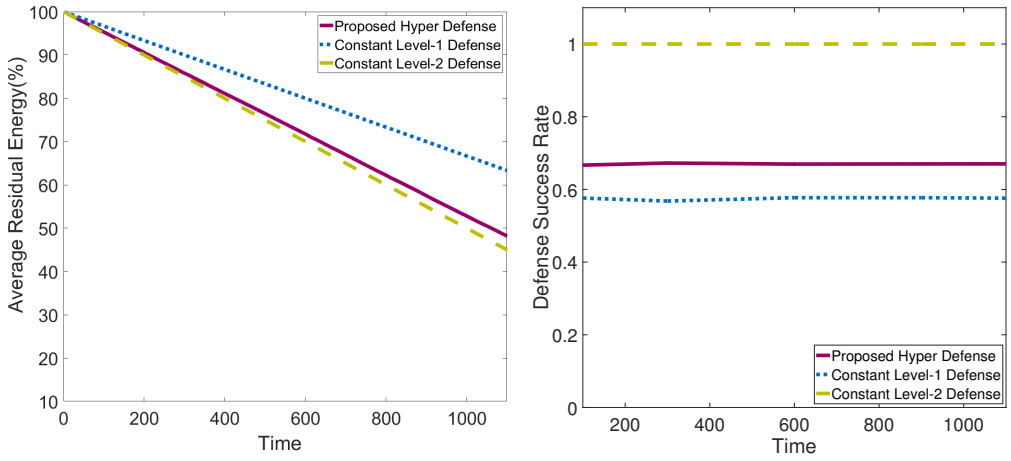
tacking and defending cost. We assume that security value ω_n is significantly higher than c_{an} and c_{dn}, and assume that $c_{an} = c_{dn}$. This means that if the attacker succeeds, the system will be at a very high risk and suffer a big loss. Figure 7.25 presents the average residual energy and defense success rate when the security value ω_n is significantly higher than the cost of attack c_{an} and cost of defense c_{dn}. It is interesting to observe that hyper defense still has a higher average residual energy than the constant level-2 defense approach as shown in Figure 7.25(a).



(a) (b)

Figure 7.24: Average residual energy and defense success rate when ω is higher than c_{an} and c_{dn} (i.e, $\omega_n > c_{an}, c_{dn}, n \in \{1, 2\}$, and $c_{an} > c_{dn}$).

Moreover, from Figure 7.25(b), we see that the proposed hyper defense achieves a higher defense success rate than the constant level-1 defense. Because of the high security value, the defender's chances of activating/utilizing the Defend-2 strategy also increase, and the chance of utilizing each strategy will be dynamically adjusted according to the variable cost in our proposed model. This implies that the equilibrium of the proposed security game is fairly robust on the performance of the hyper defense system. As a final comment, the proposed hyper defense system saves energy and achieves a high rate of success concurrently instead of turning on the defense system 100% of the time, especially for a network that emphasizes on energy efficiency.



(a) (b)

Figure 7.25: Average residual energy and defense success rate when ω is significantly higher than c_{an} and c_{dn} (i.e, $\omega_n >> c_{an}, c_{dn}, n \in \{1, 2\}$).

7.5 Summary

In this chapter, we have evaluated our proposed mechanisms through extensive simulation experiments and compared with the state-of-the-art. Through the results of EE-MAC, we observe reduced energy consumption at the cost of increased delay. EE-MAC also improves the delay performance for fixed number of nodes compared to S-MAC. In the simulation using varying traffic loads, ADP has been shown to improve energy efficiency while keeping latency low. In addition, our experimental results under dynamic network conditions show that the proposed an evolutionary routing game model is converging to strategy choices to ESS successfully. CoPA achieves better performance in terms of network lifetime and throughput compared to other popular clustering techniques as shown in the results. The proposed hyper defense system achieves a high performance in terms of residual energy and defense success rate compared to two other constant defending systems as well.

CHAPTER 8: CONCLUSIONS AND FUTURE WORK

8.1 Conclusions

As sensor nodes have limited power resources, achieving energy efficiency and security in wireless sensor network design is of the utmost fundamental issues. This dissertation provides various techniques that would satisfy the variety of requirements of real-world WSN applications, and sheds light on five proposed approaches to investigate the energy efficiency and security issues in wireless sensor network design. In this dissertation, Chapter 3 dealt with a MAC layer, while Chapter 4 dealt with dynamic feedback approach for energy efficiency in any layers of the protocol stack in WSNs. Chapter 5 presented two models of energy efficiency in routing and clustering under a game theoretic framework. A dynamic hyper approach for a defense mechanism against several types of WSNs attacks is proposed. Specifically, the major contributions of this dissertation are summarized as follows:

- Since sensor nodes consume more power while sensing and transmitting compared to idle time, achieving a low duty-cycle improves the performance in terms of energy consumption. Chapter 3 presents our novel approach by putting nodes to sleep at the cost of degraded delay performance. To that end, we propose an Energy Efficient MAC layer protocol, called EE-MAC, and derive the energy consumption, and the incurred delay when the node switches between active state and sleep state. We also propose a combined metric, which is a linear sum of the energy consumption and the incurred delay to find the optimal sleep time.

- In Chapter 4, we propose a novel adaptive energy saving approach called ADP for WSNs. The goal of ADP is to extend the network lifetime without introducing much data sensing report latency. To achieve this goal, we dynamically adjust the optimal sleep time and adapt the behavior of the sensor nodes depending on a fluctuating underlying sensing event load, remaining battery levels, and the importance of sensing data.

- In Chapter 5, we propose two new mechanisms for routing and clustering in WSNs under game theoretic frameworks. Our first approach is to design an evolutionary routing game to reduce the load and avoid collision on the most used routes in a distributed manner. We derive the equilibrium strategies of selecting the next hop in the routing game, and have proved that the mixed strategy Nash Equilibrium derived in the game is an Evolutionary Stable Strategy (ESS). Moreover, we present the replicator dynamic model to show how the populations improve their performance and converge their strategy selections to ESS over time based on payoff comparison as demonstrated by the experiment results. The second theoretic approach is based on the concept of Correlated Equilibrium (CE), called A Cost and Payment-based clustering Algorithm (CoPA). The proposed Correlated Equilibrium ensures the efficiency and fairness in the long term. Linear optimization and machine learning techniques are utilized for the solutions. We have also proposed a simple way to determine a node's eligibility to participate in the clustering game based on a flexible weighted function. The unsuitable nodes are prohibited, thereby reducing the strategy space and speeding up convergence to the equilibrium.

- Chapter 6 presents our proposed novel non-cooperative attack-defense security game formulation under different attack situations. In this game, the attacker seeks to inflict the most damage in the network without being detected, while the defender tries to maximize his defending capabilities with a constraint on the limits of the resources. We have proposed a novel hyper defense system which uses the dynamic interaction game model between the attacker and defender to derive equilibrium strategies.

We have extensively evaluated the energy efficiency of the proposed mechanisms and compared them with the state-of-the-art mechanisms in the specific target domain. Our finding shows that EE-MAC has improved performance as compared to S-MAC, ADP achieves a significant gain in energy saving, a high energy efficiency, and has a desirable effect on latency. Furthermore,

the replicator design model for WSNs routing shows that the sensor nodes in a WSN improve their performance in the long term, and the proposed CoPA achieves a superior performance over the performance of probability-based and CROSS clustering based approaches. Finally, we have shown the good performance of the proposed hyper defense model when compared with two different constant defense systems as demonstrated in the experiment results.

8.2 Future Work

Although this dissertation has made significant progress on energy efficient and security on wireless sensor networks design, there are many open research issues. A number of mechanisms proposed in this dissertation can be extended, and applied in a variety of ways. In this section, we shed a light on some of the interesting topics that are worth pursuing for future research.

One of these mechanisms can be applied as follows. In a WSN, sensors have two major operations: sensing and forwarding data [38]. In part of our dissertation, we focus on producing an energy-efficient way to sense an event based on the feedback. Other researches, such as PW-MAC [39], focus on the forwarding and transmission of sensed data. PW-MAC is an energy-efficient predictive wakeup MAC protocol that enables senders to accurately predict receivers wakeup times. The protocol minimizes idle listing and overhearing by enabling a sender to rendezvous with a receiver quickly according to the predicted receiver wake-up time. It could be beneficial to combine PW-MAC technique and our proposed ADP approach together to have a complete energy efficient scheduling system.

Another promising direction is the deeper study of the application of employing game theory in wireless communication. For example, our proposed evolutionary routing game can be extended. Although our proposed MSNE is fair and optimal, the collision may still occur. Thus, one of the coordination equilibrium in game theory may apply a solution for the routing game in order to completely avoid the collision issues in routing protocol. In addition, the interest of

the study of the security issues of the networks, and awareness of the application's requirements with all the related factors, is one of the significant directions worthy to pursue further. Therefore, employing game theory in WSNs will continue to mature and will open new possibilities for designing robust routing algorithms and security system.

LIST OF REFERENCES

[1] M. Bansal *et al.*, "An ant colony optimization algorithm to solve the broken link problem in wireless sensor network.-a review," *International Journal of Scientific Research*, vol. 4, no. 11, 2016.

[2] J. A. Stankovic, A. D. Wood, and T. He, "Realistic applications for wireless sensor networks," in *Theoretical Aspects of Distributed Computing in Sensor Networks*, pp. 835–863, Springer, 2011.

[3] T. Rault, A. Bouabdallah, and Y. Challal, "Energy efficiency in wireless sensor networks: A top-down survey," *Computer Networks*, vol. 67, pp. 104–122, 2014.

[4] K. Bok, Y. Lee, J. Park, and J. Yoo, "An energy-efficient secure scheme in wireless sensor networks," *Journal of Sensors*, vol. 2016, 2016.

[5] H. Lu, J. Li, and M. Guizani, "Secure and efficient data transmission for cluster-based wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, pp. 750–761, March 2014.

[6] S. Roy, C. Ellis, S. Shiva, D. Dasgupta, V. Shandilya, and Q. Wu, "A survey of game theory as applied to network security," in *2010 43rd Hawaii International Conference on System Sciences*, pp. 1–10, Jan 2010.

[7] S. De, C. Qiao, D. Pados, M. Chatterjee, and S. Philip, "An integrated cross-layer study of wireless cdma sensor networks," in *IEEE Journal on Selected Areas on Communications (JSAC), Special Issue on Quality of Service Delivery in Variable Topology Networks*, vol. 22, pp. 193–205, Sept 2004.

[8] P. Ji, C. Wu, Y. Zhang, and Z. ha, "Research of an energy-aware MAC protocol in Wireless Sensor Network," in *Control and Decision Conference*, CCDC, pp. 4686–4690, july 2008.

[9] W. Ye, J. Heidemann, and D. Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 12, pp. 493–506, June 2004.

[10] A. Munir and A. Gordon-Ross, "Optimization approaches in wireless sensor networks," *Sustainable Wireless Sensor Networks*, pp. 313–338, 2010.

[11] S. K. Singh, M. Singh, D. Singh, *et al.*, "Routing protocols in wireless sensor networks–a survey," *International Journal of Computer Science & Engineering Survey (IJCSES) Vol*, vol. 1, pp. 63–83, 2010.

[12] S. Sharma and A. Nayyar, "Mint-route to avoid congestion in wireless sensor network," *International Journal of Emerging Trends and Technology in Computer Science*, vol. 3, pp. 91–94, March-April 2014.

[13] G. Koltsidas and F.-N. Pavlidou, "A game theoretical approach to clustering of ad-hoc and sensor networks," *Telecommunication Systems*, vol. 47, no. 1-2, pp. 81–93, 2011.

[14] A. Attiah, M. I. Akbas, M. Chatterjee, and D. Turgut, "Ee-mac: Energy efficient sensor mac layer protocol," in *Local Computer Networks Workshops (LCN Workshops),38th Conference on*, pp. 116–119, IEEE, 2013.

[15] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell System Technical Journal*, vol. 39, pp. 1253–1265, Sept. 1960.

[16] A. Attiah, M. F. Amjad, O. Nakhila, and C. Zou, "Adp: An adaptive feedback approach for energy-efficient wireless sensor networks," in *2016 International Conference on Computing, Networking and Communications (ICNC)*, pp. 1–7, IEEE, 2016.

[17] R. Muraleedharan, I. Demirkol, O. Yang, H. Ba, S. Ray, and W. Heinzelman, "Sleeping techniques for reducing energy dissipation," in *The Art of Wireless Sensor Networks*, pp. 163–197, Springer, 2014.

[18] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, pp. 1567–1576 vol.3, 2002.

[19] W. Wang, M. Chatterjee, and K. Kwiat, "Coexistence with malicious nodes: A game theoretic approach," in *Game Theory for Networks. GameNets. International Conference on*, pp. 277–286, IEEE, 2009.

[20] A. Attiah, M. F. Amjad, M. Chatterjee, and C. C. Zou, "An evolutionary game for efficient routing in wireless sensor networks," in *Global Communications Conference (GLOBECOM),*, pp. 1–7, IEEE, 2016.

[21] P. Huang, L. Xiao, S. Soltani, M. Mutka, and N. Xi, "The evolution of mac protocols in wireless sensor networks: A survey," *Communications Surveys Tutorials, IEEE*, vol. 15, no. 1, pp. 101–120, 2013.

[22] T. van Dam and K. Langendoen, "An adaptive energy-efficient mac protocol for wireless sensor networks," in *International Conference on Embedded Networked Sensor Systems*, SenSys '03, pp. 171–180, 2003.

[23] H. Hu, J. Min, X. Wang, and Y. Zhou, "The improvement of s-mac based on dynamic duty cycle in wireless sensor network," in *Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference on*, vol. 1, pp. 341–345, June 2011.

[24] L. li Gao, "A Energy Consumption Improvements of S-MAC in WSN," in *International Conference on Internet Technology and Applications*, iTAP, pp. 1–3, Aug. 2011.

[25] G. Wang, D. Turgut, L. Bölöni, Y. Ji, and D. C. Marinescu, "A mac layer protocol for wireless networks with asymmetric links," *Ad Hoc Networks*, vol. 6, no. 3, pp. 424–440, 2008.

[26] G. Wang, D. Turgut, L. Bölöni, Y. Ji, and D. C. Marinescu, "A simulation study of a mac layer protocol for wireless networks with asymmetric links," in *Proceedings of the 2006 international conference on Wireless communications and mobile computing*, pp. 929–936, ACM, 2006.

[27] J. Ai, J. Kong, and D. Turgut, "An adaptive coordinated medium access control for wireless sensor networks," in *Computers and Communications, 2004. Proceedings. ISCC 2004. Ninth International Symposium on*, vol. 1, pp. 214–219 Vol.1, 2004.

[28] G. Zheng, J. Fu, S. Tang, Y. Li, and Z. Dong, "A dual channel-based energy efficient and low latency mac protocol for wsns," in *Networks Security Wireless Communications and Trusted Computing (NSWCTC), 2010 Second International Conference on*, vol. 1, pp. 466–469, IEEE, 2010.

[29] K.-T. Cho and S. Bahk, "Optimal hop extended mac protocol for wireless sensor networks," *Computer Networks*, vol. 56, no. 4, pp. 1458–1469, 2012.

[30] S. Dash, A. R. Swain, and A. Ajay, "Reliable Energy Aware Multi-token Based MAC Protocol for WSN," in *IEEE International Conference on Advanced Information Networking and Applications*, AINA, pp. 144–151, 2012.

[31] A. A. G. M. Shafiullah, S. A. Azad, "Energy-efficient wireless mac protocols for railway monitoring applications," in *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, Nov 2012.

[32] Y. Liu, F. Jiang, H. Liu, and J. Wu, "Sc-mac: A sender-centric asynchronous mac protocol for burst traffic in wireless sensor networks," in *2012 18th Asia-Pacific Conference on Communications (APCC)*, pp. 848–853, IEEE, 2012.

[33] H. Liu, F. Jiang, and G. Yao, "An appointment based mac protocol for wireless sensor networks," in *Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2012 International Conference on*, pp. 368–372, IEEE, 2012.

[34] O. Yang and W. Heinzelman, "Modeling and performance analysis for duty-cycled mac protocols with applications to s-mac and x-mac," *Mobile Computing, IEEE Transactions on*, vol. 11, pp. 905–921, June 2012.

[35] R. Kannan, R. Kalidindi, S. S. Iyengar, and V. Kumar, "Energy and rate based mac protocol for wireless sensor networks," *ACM Sigmod Record*, vol. 32, no. 4, pp. 60–65, 2003.

[36] O. Yang and W. Heinzelman, "Sleeping multipath routing: A trade-off between reliability and lifetime in wireless sensor networks," in *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, pp. 1–5, IEEE, 2011.

[37] L. Aslanyan, H. Aslanyan, and H. Khosravi, "Optimal node scheduling for integrated connected-coverage in wireless sensor networks," in *Computer Science and Information Technologies (CSIT), 2013*, pp. 1–13, Sept 2013.

[38] A. Erdogan, V. Coskun, and A. Kavak, "The sectoral sweeper scheme for wireless sensor networks: Adaptive," 2006.

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

[40] H.-Y. Shi, W.-L. Wang, N.-M. Kwok, and S.-Y. Chen, "Game theory for wireless sensor networks: a survey," *Sensors*, vol. 12, no. 7, pp. 9055–9097, 2012.

[41] W. Wang, M. Chatterjee, and K. Kwiat, "Coexistence with malicious nodes: A game theoretic approach," in *Game Theory for Networks. GameNets. International Conference on*, pp. 277–286, May 2009.

[42] B. Arisian and K. Eshghi, "A game theory approach for optimal routing: In wireless sensor networks," in *WiCOM*, pp. 1–7, IEEE, 2010.

[43] L. Buttyn and J.-P. Hubaux, "Nuglets: a virtual currency to stimulate cooperation in self-organized mobile ad hoc networks," tech. rep., 2001.

[44] R. Kannan and S. Iyengar, "Game-theoretic models for reliable path-length and energy-constrained routing with data aggregation in wireless sensor networks," *Selected Areas in Communications, IEEE Journal on*, vol. 22, pp. 1141–1150, Aug 2004.

[45] M. Maskery and V. Krishnamurthy, "Decentralized adaptation in sensor networks: Analysis and application of regret-based algorithms," in *IEEE Decision and Control*, pp. 951–956, Dec 2007.

[46] M. Abd, S. Majed Ai Rubeaai, K. Tepe, and R. Benlamri, "Game theoretic energy balancing routing in three dimensional wireless sensor networks," in *IEEE WCNC*, pp. 1596–1601, March 2015.

[47] M. Kordafshari, A. Movaghar, and M. Meybodi, "A joint duty cycle scheduling and energy aware routing approach based on evolutionary game for wireless sensor networks," *Journal Archive*, vol. 14, 2017.

[48] E. Altman, Y. Hayel, and H. Kameda, "Evolutionary dynamics and potential games in non-cooperative routing," in *Modeling and Optimization in Mobile, Ad Hoc and Wireless Net-*

works and Workshops, 2007. WiOpt 2007. 5th International Symposium on, pp. 1–5, IEEE, 2007.

[49] K. Komathy and P. Narayanasamy, "Trust-based evolutionary game model assisting aodv routing against selfishness," *Journal of Network and Computer Applications*, vol. 31, no. 4, pp. 446–471, 2008.

[50] E. Zeydan, D. Kivanc, C. Comaniciu, and U. Tureli, "Energy-efficient routing for correlated data in wireless sensor networks," *Ad Hoc Networks*, vol. 10, no. 6, pp. 962–975, 2012.

[51] R. Feng, T. Li, Y. Wu, and N. Yu, "Reliable routing in wireless sensor networks based on coalitional game theory," *IET Communications*, vol. 10, no. 9, pp. 1027–1034, 2016.

[52] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference on*, pp. 10–pp, IEEE, 2000.

[53] M. Chatterjee, S. K. Das, and D. Turgut, "Wca: A weighted clustering algorithm for mobile ad hoc networks," *Cluster computing*, vol. 5, no. 2, pp. 193–204, 2002.

[54] Z. Zeng-Wei, W. Zhao-Hui, and L. Huai-Zhong, "Clustering routing algorithm using game-theoretic techniques for wsns," in *Circuits and Systems. ISCAS. Proceedings of the International Symposium on*, vol. 4, pp. IV–904–7 Vol.4, May 2004.

[55] M. Esmaeeli and S. A. H. Ghahroudi, "Improving energy efficiency using a new game theory algorithm for wireless sensor networks," *International Journal of Computer Applications*, vol. 136, no. 12, 2016.

[56] M. Mishra, C. R. Panigrahi, J. L. Sarkar, and B. Pati, "Gecsa: A game theory based energy efficient cluster-head selection approach in wireless sensor networks," in *2015 International Conference on Man and Machine Interfacing (MAMI)*, pp. 1–5, Dec 2015.

[57] A. Attiah, M. Chatterjee, and C. C. Zou, "A game theoretic approach for energy-efficient clustering in wireless sensor networks," in *Wireless Communications and Networking Conference (WCNC), 2017 IEEE*, pp. 1–6, IEEE, 2017.

[58] Y. Wang, Y. Wang, J. Liu, Z. Huang, and P. Xie, "A survey of game theoretic methods for cyber security," in *2016 IEEE First International Conference on Data Science in Cyberspace (DSC)*, pp. 631–636, June 2016.

[59] A. E. Chukwudi and I. C. Eze Udoka, "Game theory basics and its application in cyber security," *Advances in Wireless Communications and Networks*, vol. 3, no. 4, pp. 45–49, 2017.

[60] M. H. Manshaei, Q. Zhu, T. Alpcan, T. Bacşar, and J.-P. Hubaux, "Game theory meets network security and privacy," *ACM Computing Surveys (CSUR)*, vol. 45, no. 3, p. 25, 2013.

[61] M. Fallah, "A puzzle-based defense strategy against flooding attacks using game theory," *IEEE Transactions on Dependable and Secure Computing*, vol. 7, pp. 5–19, Jan 2010.

[62] Y. Liu, C. Comaniciu, and H. Man, "A bayesian game approach for intrusion detection in wireless ad hoc networks," in *Proceeding from the 2006 workshop on Game theory for communications and networks*, p. 4, ACM, 2006.

[63] M. Mohi, A. Movaghar, and P. M. Zadeh, "A bayesian game approach for preventing dos attacks in wireless sensor networks," in *Communications and Mobile Computing, 2009. CMC'09. WRI International Conference on*, vol. 3, pp. 507–511, IEEE, 2009.

[64] E. O. Elliott, "Estimates of Error Rates for Codes on Burst-Noise Channels," *Bell System Technical Journal*, vol. 42, pp. 1977–1997, Sept. 1963.

[65] M. R. Ahmad, E. Dutkiewicz, and X. Huang, "A survey of low duty cycle mac protocols in wireless sensor networks," *Book Chapter in, Wireless Sensor Network*, 2009.

[66] V. Jacobson, "Congestion avoidance and control," in *ACM SIGCOMM Computer Communication Review*, vol. 18, pp. 314–329, ACM, 1988.

[67] M. R. Sheldon, "Introduction to probability models," 2010.

[68] D. Estep, "The bisection algorithm," *Practical Analysis in One Variable*, pp. 165–177, 2002.

[69] Q. Wang, "Traffic analysis & modeling in wireless sensor networks and their applications on network optimization and anomaly detection," *Network Protocols and Algorithms*, vol. 2, no. 1, pp. 74–92, 2010.

[70] Z. Han, *Game theory in wireless and communication networks: theory, models, and applications*. Cambridge University Press, 2012.

[71] E. Altman, R. ElAzouzi, Y. Hayel, and H. Tembine, "An evolutionary game approach for the design of congestion control protocols in wireless networks," in *WiOPT*, pp. 547–552, April 2008.

[72] L. Zhou and Q. Wen, "Energy efficient source location privacy protecting scheme in wireless sensor networks using ant colony optimization," *International Journal of Distributed Sensor Networks*, vol. 10, no. 3, p. 920510, 2014.

[73] D. Lin, Q. Wang, D. Lin, and Y. Deng, "An energy-efficient clustering routing protocol based on evolutionary game theory in wireless sensor networks," *International Journal of Distributed Sensor Networks*, 2015.

[74] Z. Chen, Y. Qiu, J. Liu, and L. Xu, "Incentive mechanism for selfish nodes in wireless sensor networks based on evolutionary game," *Computers & Mathematics with Applications*, vol. 62, no. 9, pp. 3378–3388, 2011.

[75] D. Fudenberg and J. Tirole, "Game theory. 1991," *Cambridge, Massachusetts*, vol. 393, 1991.

[76] K. Sigmund, *Evolutionary Game Dynamics: American Mathematical Society Short Course, January 4-5, 2011, New Orleans, Louisiana*. AMS Short Course Lecture Notes, American Mathematical Soc.

[77] P. D. Taylor and L. B. Jonker, "Evolutionary stable strategies and game dynamics," 1978.

[78] R. Jain, D.-M. Chiu, and W. R. Hawe, *A quantitative measure of fairness and discrimination for resource allocation in shared computer system*, vol. 38. Eastern Research Laboratory, Digital Equipment Corporation Hudson, MA, 1984.

[79] S. Brahma, M. Chatterjee, and K. Kwiat, "Congestion control and fairness in wireless sensor networks," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 8th IEEE International Conference on*, pp. 413–418, IEEE, 2010.

[80] R. J. Aumann *et al.*, "Subjectivity and correlation in randomized strategies," *Journal of mathematical Economics*, vol. 1, no. 1, pp. 67–96, 1974.

[81] C. H. Papadimitriou and T. Roughgarden, "Computing correlated equilibria in multi-player games," *Journal of the ACM (JACM)*, vol. 55, no. 3, p. 14, 2008.

[82] P. M. Pardalos, A. Migdalas, and L. Pitsoulis, *Pareto optimality, game theory and equilibria*, vol. 17. Springer Science & Business Media, 2008.

[83] S. Hart and A. Mas-Colell, "A simple adaptive procedure leading to correlated equilibrium," *Econometrica*, vol. 68, no. 5, pp. 1127–1150, 2000.

[84] V. P. Singh, S. Jain, and J. Singhai, "Hello flood attack and its countermeasures in wireless sensor networks," *IJCSI International Journal of Computer Science Issues*, vol. 7, no. 11, pp. 23–27, 2010.

[85] A. Dubey, D. Meena, and S. Gaur, "A survey in hello flood attack in wireless sensor networks," *Int. J. Eng. Res. Technol*, vol. 3, 2014.

[86] M. S. Haghighi and K. Mohamedpour, "Securing wireless sensor networks against broadcast attacks," in *Telecommunications, 2008. IST 2008. International Symposium on*, pp. 49–54, IEEE, 2008.

[87] R. Singh, D. J. Singh, and D. R. Singh, "Hello flood attack countermeasures in wireless sensor networks," 2016.

[88] R. S. Hassoubah, S. M. Solaiman, and M. A. Abdullah, "Intrusion detection of hello flood attack in wsns using location verification scheme," *International Journal of Computer and Communication Engineering*, vol. 4, no. 3, p. 156, 2015.

[89] R. Rehman, G. Hazarika, and G. Chetia, "Malware threats and mitigation strategies: a survey," *Journal of Theoretical and Applied Information Technology*, vol. 29, no. 2, pp. 69–73, 2011.

[90] S. Shen, H. Li, R. Han, A. V. Vasilakos, Y. Wang, and Q. Cao, "Differential game-based strategies for preventing malware propagation in wireless sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 9, pp. 1962–1973, Nov 2014.

[91] I. You and K. Yim, "Malware obfuscation techniques: A brief survey," in *2010 International Conference on Broadband, Wireless Computing, Communication and Applications*, pp. 297–300, Nov 2010.

[92] A. Sharma and S. K. Sahay, "Evolution and detection of polymorphic and metamorphic malwares: A survey," *arXiv preprint arXiv:1406.7061*, 2014.

[93] J. Kim, S. Lee, J. M. Youn, and H. Choi, "A study of simple classification of malware based on the dynamic api call counts," in *International Conference on Computer Science and its Applications*, pp. 944–949, Springer, 2016.

[94] M. Egele, T. Scholte, E. Kirda, and C. Kruegel, "A survey on automated dynamic malware-analysis techniques and tools," *ACM computing surveys (CSUR)*, vol. 44, no. 2, p. 6, 2012.

[95] A. Queiruga-Dios, A. H. Encinas, J. Martín-Vaquero, and L. H. Encinas, "Malware propagation models in wireless sensor networks: A review," in *International Conference on EUropean Transnational Education*, pp. 648–657, Springer, 2016.

[96] M. Park, H. Kim, and S. W. Lee, "User authentication for hiererchical wireless sensor networks," in *2013 14th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, pp. 203–208, July 2013.

[97] O. Nakhila and C. Zou, "Parallel active dictionary attack on ieee 802.11 enterprise networks," in *MILCOM 2016 - 2016 IEEE Military Communications Conference*, pp. 265–270, Nov 2016.

[98] V. Goyal, V. Kumar, M. Singh, A. Abraham, and S. Sanyal, "Compchall: addressing password guessing attacks," in *Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on*, vol. 1, pp. 739–744, IEEE, 2005.

[99] Q. Xu, R. Zheng, W. Saad, and Z. Han, "Device fingerprinting in wireless networks: Challenges and opportunities," *IEEE Communications Surveys Tutorials*, vol. 18, pp. 94–104, Firstquarter 2016.

[100] K. B. Rasmussen and S. Capkun, "Implications of radio fingerprinting on the security of sensor networks," in *2007 Third International Conference on Security and Privacy in Communications Networks and the Workshops - SecureComm 2007*, pp. 331–340, Sept 2007.

[101] N. T. Nguyen, G. Zheng, Z. Han, and R. Zheng, "Device fingerprinting to enhance wireless security using nonparametric bayesian method," in *2011 Proceedings IEEE INFOCOM*, pp. 1404–1412, April 2011.

[102] M. Inc, “Telmosb mote platform.”

[103] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless microsensor networks,” in *System Sciences. Proceedings of the 33rd Annual Hawaii International Conference on*, p. 10 pp. vol.2, Jan 2000.