5-27-2011

Modeling Security and Cooperation in Wireless Networks Using Game Theory

Charles A. K. Kamhoua
Florida International University, kkcharlesa@yahoo.fr

FLORIDA INTERNATIONAL UNIVERSITY

Miami, Florida

MODELING SECURITY AND COOPERATION IN WIRELESS NETWORKS USING GAME THEORY

A dissertation submitted in partial fulfillment of the
requirements for the degree of
DOCTOR OF PHILOSOPHY
in
ELECTRICAL ENGINEERING
by
Charles Alexandre Kenmogne Kamhoua
2011

To: Dean Amir Mirmiran
College of Engineering and Computing

This dissertation, written by Charles Alexandre Kenmogne Kamhoua, and entitled Modeling Security and Cooperation in Wireless Networks using Game Theory, having been approved in respect to style and intellectual content, is referred to you for judgment.

We have read this dissertation and recommend that it be approved.

Deng Pan

Kang K. Yen

Norman D. H. Munroe

Kia Makki, Co-Major Professor

Niki Pissinou, Co-Major Professor

Date of Defense: May 27, 2011

The dissertation of Charles Alexandre Kenmogne Kamhoua is approved.

Dean Amir Mirmiran
College of Engineering and Computing

Dean Lakshmi N. Reddi
University Graduate School

Florida International University, 2011

DEDICATION

To my late father Roger Kamhoua, my mother Solange Kamhoua, and my wife Francine Kamhoua.

ACKNOWLEDGMENTS

I would like to express my gratitude to my advisers Dr. Niki Pissinou and Dr. Kia Makki. I am indebted to Dr. Niki Pissinou who introduced me to the area of game theory. Besides game theoretic applications to Telecommunication Networks, this research empowered me to have a better understanding of rationality, human behavior, and social interactions. The advices and guidance she provided allowed me to overcome significant challenges during my dissertation. I have also been fortunate to have Dr. Kia Makki actively involved in my research. I thank him for his invaluable guidance and for pushing me to always publish high quality papers. I hope to be able to live up to his expectations in the future.

I am thankful to Dr. Deng Pan, Dr. Kang Yen, and Dr. Norman Munroe for serving on my dissertation committee. Their constructive comments have improved the quality of this work. They have been helpful during the entire Ph.D. process.

I also feel deep gratitude towards my collaborator and co-authors. Colonel Jerry Miller (Ret.) at the Applied Research Center has always been available to answer any of my questions. My frequent exchanges with Dr. Kami Makki have been reflective.

I also thank the excellent faculty members of the Electrical and Computer Engineering Department. Their lectures and class projects have been influential. In particular, I received valuable support and encouragement from Dr. Jean Andrian.

Additionally, I show gratitude to Dr. Kevin Kwiat of Air Force Research Laboratory. Our initial conversation on fault tolerant networks and network security has been insightful.

Because of their constant motivation, I would like to thank my fellow lab mates at the Telecommunication and Information Technology Institute. Particularly, I thank Chanii Haley and Anthony Barreto for reading and removing the typos in my early writing. Likewise, I appreciate the comments of undergraduate students I have mentored over the last three years in the NSF/REU program.

I am grateful to the faculty members of ENSET at the University of Douala where I did my undergraduate studies. Especially, I thank my thesis advisors Dr. Martin Mbouenda and Mr. Felix Paune. Also, I show gratitude to Mr. Lazare Kamdem for his true support.

I am lucky to have close friends like Merlin Ngachin and Serge Feuze. I am glad of the stimulation and support they have provided. As graduate students themselves, they clearly understood all the challenges I was going through.

Furthermore, words are not enough to express the love and blessings from my family members. They have always been patient and have motivated my desire to pursuit graduate studies. Especially, my communication with Olivier Tsemogne at the beginning of my dissertation deepened my motivation to undertake a research topic related to game theory. Also, I would like to appreciate all the devotion and sacrifice made by Veronique Maga for making this dissertation possible.

Finally, I would like to acknowledge that my graduate studies have been partially funded by the US National Science Foundation and Florida International University through the Applied Research Center, the Electrical Engineering Department, and the Telecommunication and Information Technology Institute.

ABSTRACT OF THE DISSERTATION

MODELING SECURITY AND COOPERATION IN WIRELESS NETWORKS USING GAME THEORY

by

Charles Alexandre Kenmogne Kamhoua

Florida International University, 2011

Miami, Florida

Professor Niki Pissinou, Co-Major Professor

Professor Kia Makki, Co-Major Professor

This research involves the design, development, and theoretical demonstration of models resulting in integrated misbehavior resolution protocols for ad hoc networked devices. Game theory was used to analyze strategic interaction among independent devices with conflicting interests. Packet forwarding at the routing layer of autonomous ad hoc networks was investigated. Unlike existing reputation based or payment schemes, this model is based on repeated interactions. To enforce cooperation, a community enforcement mechanism was used, whereby selfish nodes that drop packets were punished not only by the victim, but also by all nodes in the network. Then, a stochastic packet forwarding game strategy was introduced. Our solution relaxed the uniform traffic demand that was pervasive in other works. To address the concerns of imperfect private monitoring in resource aware ad hoc networks, a belief-free equilibrium scheme was developed that reduces the impact of noise in cooperation. This scheme also eliminated the need to infer the private history of other nodes. Moreover, it simplified the computation of an optimal strategy. The belief-free approach reduced the node overhead and was easily tractable. Hence it made the system operation feasible. Motivated by the versatile nature of evolutionary game theory, the assumption of a rational node is relaxed, leading to the development of a framework for mitigating routing selfishness and misbehavior in Multi hop networks. This is accomplished by setting nodes to play a fixed strategy rather than independently choosing a rational strategy. A range of simulations was carried out that showed improved cooperation between selfish nodes when compared to older results. Cooperation among ad hoc nodes can also protect a network from malicious attacks. In the absence of a central trusted entity, many security mechanisms and privacy protections require cooperation among ad hoc nodes to protect a network from malicious attacks. Therefore, using game theory and evolutionary game theory, a mathematical framework has been developed that explores trust mechanisms to achieve security in the network. This framework is one of the first steps towards the synthesis of an integrated solution that demonstrates that security solely depends on the initial trust level that nodes have for each other.

TABLE OF CONTENTS

CHAPTER PAGE

LIST OF TABLES

LIST OF FIGURES

FIGURES PAGE

ABBREVIATION

AODV	Ad hoc On-demand Distance Vector
ARP	Address Resolution Protocol
BFE	Belief-Free Equilibrium
BFES	Belief-Free Equilibrium Strategy
DDOS	Distributed Denial of Service
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name Server
DoS	Denial of Service
DSR	Dynamic Source Routing
EGT	Evolutionary Game Theory
ESS	Evolutionary Stable Strategies
GTFT	Generous Tit For Tat
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IP	Internet Protocol
IPD	Iterated Prisoners’ Dilemma
IPS	Intrusion Prevention System
MAC	Medium Access Control
MANET	Mobile Ad Hoc Network
PDG	Prisoners’ Dilemma Game
PGS	Patient Grim Strategy

PONT	Punish Only *n* Times
P2P	Peer-to-Peer
QoS	Quality of Service
RWP	Random Waypoint
SGT	Stochastic Game Theory
SPD	Stochastic Prisoners’ Dilemma
SPE	Subgame Perfect Nash Equilibrium, Subgame Perfect Equilibrium
TCP	Transmission Control Protocol
TFT	Tit For Tat
UDP	User Datagram Protocol
VANET	Vehicular Ad Hoc Network
WMN	Wireless Mesh Networks
WSN	Wireless Sensor Networks

CHAPTER 1

INTRODUCTION

The widespread nature of wireless communication networks nurtures the dream of interconnection between any device, anytime, anywhere in the world. The fulfillment of this vision requires the participation of several nodes, from multiple network domains with different objectives and preferences. With the rapid advances in computer and wireless communications devices and associated techniques, mobile sensor networks are becoming feasible and attracting more and more research attention recently. This is because future wireless communication will entail autonomous devices, such as sensors, smart phones, and computers, to be interconnected in an ad hoc manner and without any underlying networked infrastructure. An ad hoc network is a temporary, on the fly, connection between typically autonomous devices where each device decides whether and to what extent it wishes to participate in the network. In the pursuit of their own interests, the participating devices could therefore misbehave – either by being selfish or by being malicious [1-20]. Selfish nodes attempt to save scarce resources such as battery power, bandwidth, memory spaces, and computational power. As a consequence, each node will strive to save its limited supply while competing to gain access to others' resources with the goal of maximizing their own capacity.

In such an autonomous ad hoc network, each node could misbehave. We can distinguish at least three types of misbehaving nodes: faulty, selfish and malicious nodes. Faulty nodes do not follow the recommendations of the protocol because they are damaged. Selfish nodes strive to maximize their individual outcome. A selfish node will misbehave if the protocol recommendation is not consistent with its self-interest. Malicious nodes attempt to destroy the network. They launch various attacks such as Denial of Service and selective forwarding.

The primary function of a network is to guarantee a secure connection between different nodes. Therefore, the primary issue that would arise in autonomous ad hoc networks would be about packet forwarding. In our targeted scenarios, nodes are selfish and there is no infrastructure. When the sender is not in the communication range of the destination, the packets go through multi hop communication. Intermediate nodes between the sender and the destination are requested to forward the packet for others. But, there is a cost in battery power and bandwidth associated with forwarding packets and therefore, it is not in the best interest of an intermediate node to forward packets. Yet, if all nodes refuse to forward packets for others, the network collapses. However, no node is interested in this outcome. This situation is similar to the *n* persons Prisoners' Dilemma Game (PDG) found in game theory. Therefore, in our work we use the repeated PDG to model the packet forwarding game in a network [1-4].

As we can see, selfish node behavior may result in, at the least, network performance degradation. Thus, efficient mechanisms need to be designed to enforce node cooperation at all layers. In fact, aside from the selfish behavior at the routing layers that we describe above, selfishness and conflicting interests can also cause several types of misbehaviors in other layers of networks without central managers. For instance, at the physical layer, a node can selfishly increase its power to successfully transmit its packets. The natural response of other selfish nodes would be to also increase their power. The result would be an unacceptable interference level that deteriorates the network operation. Looking at the MAC layer, some nodes may disregard the back off period in an attempt to send more packets. Instead of waiting for other nodes to finish sending their packet, each node may attempt to quickly send its own packets. As a consequence, there may be an increase in collision that could dramatically affect the network performance. Let's also consider the transport layer; the most widely used protocol in this layer is the Transmission Control Protocol (TCP). The flow allocation achieved by TCP is almost fair as a result of the additive increase and multiplicative decrease of the *window size* at each end point. The *window size* is the number of packets a node sends at once. A node using TCP must cut the window size in half after packet loss due to congestion. A selfish node may be reluctant to decrease its window size during congestion to maximize its own flow allocation. If all nodes adopt such behavior, congestion will be aggravated.

Alongside selfish behavior, the research in this dissertation examines autonomous network security. Security mechanisms in traditional networks rely on trusted systems like certificate authorities to operate. In the absence of such an authority, nodes are required to trust each other to secure the network and protect their privacy. Trust, in this context, can be understood as the belief or the confidence a node has about the appropriate participation of others in the security mechanism. Without trust, no intelligent node will participate in any security mechanism that involves the collaboration of several nodes to be successful. Absolutely, numerous security mechanisms require some level of trust in their design and implementation [21-26].

In addition to autonomous ad hoc networks, we looked into traditional wireless networks. We considered a service provider with a multitude of users. Upgrading the network infrastructure to accommodate the exponential increase in bandwidth demand observed in recent years is costly for the service providers. A provider would prefer to minimize their investment in upgrading the network while maximizing the number of customers using the product. On the other hand, without regular upgrades to the network from the provider, there may be more congestion, more delays, and generally, a low Quality of Service (QoS). One option available to unsatisfied users is to quit their provider. Surely, there is a conflict of interest between a service provider and its users. The optimum behavior of the provider will be connected to the users' strategies and organization.

We can see that independent nodes are involved in strategic interactions at all protocol layers to secure the network or when interacting with their provider. The success of one node behavior will depend on the behavior of other nodes. Game theory is the unifying mathematical framework able to model the conflict faced by the different nodes in each situation mentioned above and scrutinize the possible solutions with a precise characterization of their properties. This justifies the use of game theory as the main method employed in our investigation. In addition, when appropriate, we relaxed the assumption of node rationality used in the game theory model to advance a framework based on Evolutionary Game Theory (EGT) [4-5].

It becomes evident that the realization of the ideal of interconnection between any device, anytime, anywhere in the world and other computing paradigms is not possible if efficient mechanisms were not designed to stimulate nodes' participation or provide adequate compensation to all network entities. Clearly, a network protocol at all layers must take into account the self-interest of independent nodes performing separate optimization. Cooperation among ad hoc nodes can also protect a network from malicious attacks. In the absence of a central, trusted entity, new classes of distributed security mechanisms and privacy protections require cooperation among several ad hoc nodes to protect a network from malicious attacks [21-23]. This dissertation specifically addresses selfish and malicious nodes. The interconnection among selfishness, cooperation, and network security is investigated.

1.1 Research Objective

The main objective of this dissertation is to use game theory and EGT to model and analyze strategic interactions in wireless networks with selfish and malicious users. Our objective for this research is threefold:

1. Design distributed game theoretic and EGT algorithms using only local information to enforce nodes cooperation at the routing layer of the network and optimize network performance with autonomous nodes, incomplete information, and imperfect monitoring.

2. Use game theory and EGT to analyze new security mechanism for networks with autonomous nodes without a central manager or a trusted authority.

3. Investigate and propose adequate game theoretic solutions to the tremendous increase in bandwidth demand due to an expansion of the number of smart phones, iPhones, PDAs, and other mobile devices.

Specially, our work focuses on:

- The study of various techniques for incorporating imperfect monitoring, imperfect information, incomplete information into the game model for networks using both game theory and EGT.
- Comparing the performance of models developed using game theory and EGT and investigating the underlying assumptions of those theories in the context of ad hoc networks with mobile nodes and dynamic change in the network topology.
- Designing game theoretic and EGT algorithm for autonomous networks that achieve performance similar to those of cooperative networks with a central authority.
- Providing a detailed mathematical analysis of autonomous network security models to capture in a general contest the equilibrium conditions in the network.
- Predicting the user and provider behaviors under diverse noise levels and different monitoring schemes.

1.2 Significance

This research has significant impact in several areas. First, our work on enforcing cooperation at the routing layers provides significant insight into the problem of distributed decision-making, self-adaptation, self-management, random interactions, and the resource efficiency needed to cope with rapid changes in network topology. Our approach leads to a precise characterization of the properties of cooperation in ad hoc networks. Second, this research establishes one of the first solid frameworks for integrating realistic network conditions into autonomous network models. By realistic, we mean incorporating noise, private monitoring, and private information into the model. This work allows the self-interest of individual nodes to be in agreement with the common interest or better network performance. Further, as with traditional networks, autonomous networks need to be secured to authenticate the nodes, prevent misuse, detect anomalies, and protect user's privacy. Our research provides novel concepts and fundamental knowledge concerning the modeling, analysis, design, and robust control of complex real-time dynamic systems, including distributed autonomous network security. This work evaluates distributed security mechanism adequate for ad hoc networks. Clearly, this research facilitates the secure interconnection of autonomous devices in the cyberinfrastucture for the good of each user.

1.3 Organization and Contribution

This dissertation is divided in nine Chapters. Each Chapter presents specific research contributions and has been peer reviewed, presented at international conferences, and published. **Chapter 2** is about the related works that provide a critical analysis of the contribution of other researchers.

We start with the packet forwarding game. Initially, in **Chapter 3** [1], we assumed perfect monitoring and rational nodes. We analyzed the consequences of traffic load inequality on the packet forwarding game. Unequal traffic loads arise in the network because existing routing protocols in multi hop networks are designed to choose the shortest available path between the sender and the destination. As a consequence, the traffic load at the center of the network is significantly higher than at the border. A community enforcement mechanism is used to compel all nodes to cooperate. The defectors are punished not only by the victim, but also by any node that observes that defection. The set of individually rational and feasible payoffs is precisely characterized. This is the first work to derive the exact utility function of each node according to its position in a static network. Then, we show that if a load balancing algorithm is not used in a distributed static network, cooperation ultimately breaks down as the number of nodes increases.

Subsequently, **in Chapter 4** [2], the packet forwarding game model is a stochastic PDG. A stochastic dimension is introduced to eliminate the unrealistic assumption in other works that each node has packets to send in each time slot or that any two neighbors have uniform network traffic demands. A game theoretic algorithm, sequential equilibrium of the packet forwarding game able to enforce cooperation with a limited number of punishments is developed.

Thereafter, in **Chapter 5** [3], noise and imperfect monitoring are considered. Congestion, interference, and noise create inconsistency between observed actions and the true actions of a node in a distributed network. We use a *belief-free equilibrium* approach because it is easily tractable. The need to infer other nodes' private history is eliminated and the computation of optimal strategy is simplified. Nodes are indifferent between cooperation and defection at all histories. The simple strategy designed works with all types of monitoring technology: perfect monitoring, imperfect public monitoring, and imperfect private monitoring.

Ultimately, in **Chapter 6** [4], we relax the assumption of rationality used in game theoretic models and use the framework of EGT. Nodes do not choose strategies themselves but are pre-programmed to play a pure or mixed strategy. Pavlov and its variant *p*Pavlov are proposed to enforce node cooperation. Both Pavlov and *p*Pavlov are Evolutionary Stable Strategies (ESS) and are stable in the replicator dynamic. It is shown that *p*Pavlov is more appropriate than Pavlov if the discount factor is low.

The second part of my dissertation is presented in **Chapter 7** [21] and involves autonomous network security. As with traditional networks, these networks need to be secured to authenticate the nodes, prevent misuse, detect anomalies, and protect user privacy. Network security and privacy protection without a central manager is challenging. In autonomous networks, many security mechanisms and privacy protections require the cooperation of several nodes to defend the network from malicious attacks. For instance, new researches have presented a light weight cryptographic technique for distributed network based on hierarchical multi-variable symmetric functions. However, those algorithms use a polynomial of degree *k* and are therefore *k*-secure. This means that if *k* nodes do not protect their key, an attacker can break the security mechanism and have access to all encrypted communication. If that happens, there is no need for a node to encrypt the message because they can easily be decrypted. A rational node may even prefer not to encrypt because encrypting a message will have no other purpose than creating a useless and costly message overhead. Therefore, we particularly investigated when, for each node, it is cost-effective to freely participate in the security mechanism or protect its privacy depending on whether that node believes or trusts that all other nodes or at least a minimum number of other nodes will do the same. In this case, each node is involved in a trust dilemma that we also model using the mathematical framework of game theory and EGT. We demonstrate that the end result (secured or unsecured network) solely depends on the initial trust level that the nodes have for each other. Consequently, it is not possible for the nodes to move from the *low trust* equilibrium (unsecured network) to the *high trust* equilibrium (secured network). This is the first work to model the connection between security and trust in autonomous ad hoc network.

Chapter 8 [27] presents the last part of this work. The demand on mobile data usage has exponentially increased since the introduction of iPhones in 2007. The network became congested as millions of users tried to browse websites and social networks, send e-mails, stream multimedia, and transfer files simultaneously. An immediate solution for the providers would be to change their pricing strategy with the goal of slowing down heavy users and decreasing the bandwidth demand. From the users' standpoint, network providers must constantly upgrade their infrastructure to accommodate new applications and devices. However, upgrading the infrastructure will be costly for the provider. A provider goal is to maximize its revenues by attracting the maximum number of customers while minimizing the infrastructure cost. On the other hand, the users would prefer a regular upgrade of the network to have less congestion, less delays and a high QoS. Moreover, users that experience bad connection will be tempted to switch providers. The dynamic communication market and the users' and provider's interaction are analyzed in the framework of repeated game theory. Noise in users' monitoring is considered. Two scenarios are compared: individual and independent actions of users as opposed to the collective actions of users. The result indicates that collective actions achieve a higher welfare to users compared to independent actions.

As you can see, this research investigates multiple network functions and takes into consideration the self-interest of autonomous agents while at the same time achieving a system performance close to that of traditional networks with unconditionally cooperative nodes. Independent nodes perform separate optimizations and reaches efficient equilibrium under our proposed models. **Chapter 9** concludes this dissertation and proposes several directions for future research.

CHAPTER 2

RELATED WORK

In this Chapter, we give a brief survey of related research in three areas: packet forwarding games, network security games, and user provider games. Also, before discussing related work in packet forwarding games, we will present two other approaches used to induce autonomous node cooperation, namely the virtual currency system and the reputation mechanism. We will summarize each approach and identify their problems and limitations.

2.1 Virtual Currency System

Nuglets and Sprite are the two most popular approaches that use the virtual currency system to address cooperation. Both offer direct incentives to cooperating nodes. We first present Nuglets followed by Sprite.

Buttyan and Hubeau [28], use an economic approach to propose Nuglets, a virtual currency system. The authors present two models to pay the packet forwarding service: the packet trade model and the packet purse model. In the packet trade model, the sender does not pay to send a packet. The next hop buys the packet and sells it for more Nuglets until the destination that pays the total cost of forwarding the packet. The intermediate nodes have some incentives to forward the packet and earn Nuglets. This model is vulnerable to network overload because senders can send unimportant messages through the network and do not have to pay. The Packet purse model solves the network overload problem. In this model, the senders pay to send a packet by loading some Nuglets in the packet. Each intermediate node acquires some Nuglets from the packet before forwarding it. If a packet does not have enough Nuglets, it is dropped. Both models require tamper resistant software and hardware to store the correct amount of Nuglets. In addition, nodes at the periphery of the networks do not have the same opportunity to accumulate the virtual currency. Another serious difficulty includes the per-hop charge to forward a packet or the possibility of collusion in case of auction to find the cheaper next hop.

In [28], Zhong *et al*. propose Sprite. Sprite focuses on selfish nodes and does not address the case of malicious and faulty nodes. Sprite relies on a central authority: the Credit Clearance Service (CCS). Nodes keep a receipt of each message they receive. They submit those receipts to the CCS to claim payment. The CCS determines the credit to each intermediate node and charges the sender. This system does not target a balanced payment but motivates nodes to cooperate. Also, Sprite cautiously computes payment to prevent cheating and collusion among nodes. Unlike Nuglets, Sprite does not require tamper proof hardware. However, the fact that the CCS is a central authority violates a premise of ad hoc networks which are distributed in their nature.

2.2 Reputation System

Unlike the virtual currency system, which provides direct incentives to cooperating nodes, reputation systems punish non-cooperating nodes. Many models have been used to model reputation, in particular, the Beta reputation system [24, 30-31] which has a strong foundation in statistics. Here, a watchdog mechanism, in each node, is used to monitor the behavior of its neighbors. After sending a packet, the node listens and observes if the packet has been forwarded or not, and records the result in a reputation table. Those models assume that omnidirectional antennas are used. A second assumption is the promiscuous mode of the wireless interface. Models differ on whether to share reputation information or not. First hand information is obtained by a node itself. Second hand information is reputation information that the node receives from other nodes. Second hand information helps to accelerate the convergence time of the algorithm. However, propagating second hand information creates traffic load in the network. Models also differ in weight given to new and old information. When giving more weight to new information, cooperating nodes can lose their reputation in low network activity. On the other hand, when giving more weight to old information, a malicious node can accumulate a good reputation and start dropping packets with impunity. Another common assumption is that all packet loss is due to misbehavior. Thus, packet losses due to congestion or noise are not taken into account.

Marti *et al* [32] proposed a system with two tools. A watchdog that listens to the its neighbors and identifies misbehaving nodes; and a pathrather, that selects the best route to avoid malicious nodes. The system achieves an acceptable throughput in the presence of misbehaving nodes. However, this system does not eliminate misbehaving nodes. Misbehaving nodes can send their own packets in the network even though they do not forward packets from other nodes.

Michiardi and Molva [33] develop CORE (Collaborative Reputation), a reputation mechanism for mobile ad hoc networks (MANET). The mechanism uses three types of reputation: subjective reputation from first hand information, indirect reputation from second hand information, and functional reputation calculated with respect to different functions like routing and forwarding. Functional reputations are combined into the global reputation by giving accurate weight to each function. This combination is a problem in itself; combining reputation does not allow the mechanism to trust a node for a specific function and not for others. The weight to give to each function is also problematic. Moreover, only positive information is propagated to avoid DoS. However, malicious nodes can collude, propagate positive reputation of each other in the network, and gain longer access. Core is simulated using the Dynamic Source Routing (DSR) protocol and shows adequate performance.

Buchegger and Boudec [34], recommend Cooperation Of Nodes-Fairness In Dynamic Ad hoc NeTworks (CONFIDENT). CONFIDENT detects and isolates misbehaving nodes. The CONFIDENT protocol has four components: the monitor, the reputation system, the path manager, and the trust manager. The monitor is similar to the watchdog, the reputation system rates nodes, the trust manager issues ALARM messages, and the path manager makes decisions. Unlike CORE, CONFIDENT propagates only negative information. The authors argue that malicious behavior is the exception, not the norm. However, this allows misbehaving nodes, even without collusion, to mount false accusation attacks and then eliminate cooperating nodes from the network.

Ganeriwal and Srivastava [31] introduce a Reputation-based Framework for Sensor Networks (RFSN). First and second hand information are used for reputation, but only first hand information is propagated. This prevents trust recommendations from looping back at the originating nodes. Moreover, only positive information is transmitted. This is to avoid bad-mouthing attacks. To combine direct information and second hand information and obtain a new reputation value, Dempster-Shafer belief theory is used. This is to take into account the fact that reputation information from the most trusted nodes must have more weight. Nodes with low reputation are detected and eliminated from the networks. Aging is used to give more weight to fresh information. Thus, cooperating nodes can lose their reputation in low network activity.

2.3 Game Theory

Game theory is used to model strategic interaction among rational players. It cannot be used to model irrational misbehavior of faulty nodes. Nevertheless, it is adequate to mitigate selfishness and malicious behaviors. In ad hoc network, the players are the nodes. Each node wants to maximize its own utility (payoff). That means sending the most possible packets while forwarding the least packets and saving energy and bandwidth. The best objective in a network is to converge to a Pareto efficient Nash equilibrium [35]. However, the main challenge is that the allocations in the Nash equilibrium are not always Pareto efficient [35]. The following are some of the approaches using game theory. A survey on wireless ad hoc networks games is available in [20].

Srinivasan *et al*. [7], motivated cooperation in a network using Generous Tit For Tat (GTFT). The authors proved that GTFT is a Nash equilibrium of the forwarding game. However, to compute the equilibrium of the game, each node must know all nodes in the network, its own utility, and the utility of all the other nodes. This is a strong requirement for distributed networks. Our work in Chapter 3-5 presents distributed game theoretic algorithms able to enforce autonomous nodes' cooperation.

Yan [10], and also Yan and Hailes [11-12] developed models for cooperation in wireless multi-hop networks. They used the Prisoners' Dilemma game as the base of their model. They assumed that any two neighbors have uniform network traffic demand. We believe this is not always the case in a network. This assumption is relaxed in Chapter 3-5.

Felegyhazi *et al*. [6] used game theory in combination with graph theory to investigate and prove the conditions under which cooperation can evolve in the network. The authors concluded that the probability of full network cooperation is very small. Nonetheless, as the authors point out, local subsets of cooperating nodes may exist. Actually, this model relies on a dependency loop. However, a node may not be able to know its dependency loop in a fully distributed network. Each node only knows its neighbors as opposed to the full network topology. Clearly, dependency loop will not be common knowledge among nodes.

Jade *et al*. [9] combined virtual currency and Stochastic Game Theory (SGT) to formulate an optimal policy to forward packets towards route in peer-to-peer networks. When a source node requests data from the destination, each intermediate node is paid a virtual currency to relay the packet. Their optimal policy is achieved based on cost, free bandwidth, and service capacity. Their incentive-based routing protocol shows better performance compared to the DSR protocol.

Sagduyu and Ephremides [8] used SGT to address the cross layer problem of joint Medium Access Control (MAC) and routing in ad hoc wireless networks. They used a simple network consisting of a transmitter, a relay node, and a common destination to present their model. At the routing layer, the transmitter can send its packet directly to the destination or through the relay node, while the relay decides to accept packets from the transmitter or to send its own packets. At the MAC layer, the transmitter and the relay decide to transmit or to wait in each time slot. Each node selects the probability distribution of the available actions with the objective of optimizing their individual performance measure of throughput, delay, and energy consumption. Because the sender is in the communication range of the destination in this model, our outstanding problem of cooperation in multi-hop networks packet forwarding is not addressed.

Srivastava and DaSilva [13] relax the assumption of perfect monitoring. They model the network packet forwarding game as a game of imperfect public monitoring. In such a game, past actions of nodes are imprecise and noisy, but it is assumed that nodes commonly observe a public signal about the actions of others. However, the availability of such signals in a distributed network is not always guaranteed.

Ji *et al*. [14-15] use a game of imperfect private monitoring to model noise in the network packet forwarding game. A public signal is not needed. Nodes do not have common knowledge about the history of the game but each node has a private history of the game. Each node needs to infer the private history of other nodes based on their own imperfect observations. Those inferences become complicated in the long term. This is called a *belief-based* approach because the equilibrium strategy depends on the opponents' private history. We present a simpler approach in Chapter 5, namely a *belief-free* equilibrium approach. A *belief-based* approach requires more computational power than our model. Next to the work of Ji *et al*, Yu and Liu [17] propose a game theoretic approach to a secure cooperation in ad hoc network.

Comparably, mechanism design has been used to enforce node cooperation and develop optimum and truthful routing mechanism. By definition, mechanism design is a field of game theory that investigates how privately known preference of several strategic players can be aggregated toward a desirable outcome. The desirable outcome is sometimes the maximization of some utility function or to have strategic players truthfully reveal their private information.

Brahma [18] proposes a path auction based routing to enforce node cooperation. Each node announces their privately known capacity and cost. His scheme ensures that each node maximizes its profit by truthfully reporting their cost and capacity. His model is supported by theoretical demonstration as well as simulation. However, to prevent selfish nodes from dropping packets after collecting the payment, he proposes that the destination make the payment. Consequently, as mentioned before, malicious nodes can freely send useless packets to saturate the network.

Finally, Neely [36] suggests free market to induce node cooperation. A different market model is proposed in [37].

2.4 Evolutionary Game Theory

Researchers have also used EGT to motivate cooperation in multi-hop networks. Unlike game theory, EGT does not assume rational nodes. EGT derives from game theory and evolutionary biology. EGT can be used to analyze the evolution of strategies in a network. Specifically, EGT can describe how the frequencies of strategies change over time depending on their relative success. Moreover, EGT helps identify which strategies will resist random mutation in the long term.

Crosby and Pissinou [5] used EGT in multi-domain Wireless Sensor Networks (WSN) to develop a new protocol: Patient Grim Strategy (PGS). A player following PGS cooperates and continues cooperating until the other player defects *N* times and then defects forever. A drawback of PGS is its inability to cooperate again after *N* defections. Thus, in case of network congestion or noise, cooperating nodes can be punished or eliminated from the network forever. The result can be the collapse of the network when all nodes defect.

Komathy and Narayanasamy [19], used Tit For Tat (TFT), Pavlov, and generic algorithms to improve the Ad hoc On-demand Distance Vector algorithm (AODV). Their results show that AODV-Pavlov and AODV-TFT achieve a higher packet delivery ratio than the normal AODV. Chapter 6 proposes a packet forwarding game based on EGT.

2.5 Network Security Game

The interest of using game theory to address network security challenge has increased in recent years. In general, the main objective of the attacker is to intelligently choose its strategy to maximize the damage to the network while the manager tries to minimize the damage. The attacker's and the defender's objectives are strictly opposed. This justifies the use of a zero-sum game to model network security [38]. When each player applies the best response to its opponent strategy, the game reaches the well known Nash equilibrium. Neither the attacker nor the manager can unilaterally make a profitable deviation from the Nash equilibrium. Other models relax the assumption of rationality used in game theory and use the mathematical framework of EGT to model network security [39].

In the game presented in [40], the strategy and the payoffs are assumed to be common knowledge. In this case, the network security game is a game of complete information. Otherwise, we have a game of incomplete information that can be formulated as a Bayesian game as in [41]. The network security game can also be modeled as a static game [40-41], a repeated game, or more generally as a stochastic game [38, 42]. A stochastic game is a generalization of a repeated game. In a repeated game, players play the same stage game in all periods, whereas in a stochastic game, the stage game can randomly change from one period to the next. Game theory also provides a solid framework to model intrusion detection in a network [41, 43]. A survey of game theory as applied to network security is provided in [44]. A detailed presentation of game theory and EGT is found in [35] and [45] respectively. Our approach to network security game is presented in Chapter 7.

2.6 User Provider Game

There is a rich literature on game theoretic modeling of user-provider interactions. Hassan *et al* [46] show that the user can use a brinkmanship technique to provide credible threats to the provider and therefore constraint the provider to allocate more resources to users. Sengupta *et al* [47] investigate a market in which multiple service providers compete to get a large portion of the spectrum and sell it to the maximum number of users. Other works analyzing provider' price competition to attract users include [48-50].

The work in [51] examines one provider’s Nash equilibrium price under asymmetric information. A comprehensive survey of wireless service providers and user’s interactions can be found in [52]. Chapter 8 analyzes the game between one service provider and its multitude of users.

CHAPTER 3

GAME THEORETIC ANALYSIS OF COOPERATION IN AUTONOMOUS MULTI-HOP NETWORKS: THE CONSEQUENCES OF UNEQUAL TRAFFIC LOAD

Researchers have investigated non cooperative issues in wireless sensor networks and mobile ad hoc networks in the past decade. In particular, packet forwarding is of critical importance in such multi-hop networks. This is because there are no preexisting infrastructures. Each node must not only send and receive its own packets but also relay packets to other nodes. However, most works overlook the fact that the traffic load at the center of the network is significantly higher than at the border when using existing routing protocols. Therefore, a node at the center is requested to forward more packets than a node at the border of the network. This inequality in traffic load can break down many solutions proposed to motivate node cooperation. In this Chapter, we quantify the number of packets a node is requested to forward as a function of its position. We show that, if a load balancing algorithm is not used in a distributed static network, cooperation ultimately breaks down as the number of nodes increases. We support our result with mathematical proofs.

3.1 Introduction

Wireless ad hoc networks and WSN are distributed in their nature. They do not rely on a preexisting infrastructure. They are self configurable. They can promptly be deployed in case of disaster recovery. Packets go through a multi-hop communication route from a sender to a destination if the destination is not in the communication range of the sender. Therefore, each node is at the same time a terminal and a router. As a terminal, a node sends and receives its own packets. As a router, a node is requested to relay packets from other nodes. Most routing protocols in multi-hop networks assume full cooperation of nodes to participate in route discovery, route maintenance, and packet forwarding. The assumption of full node cooperation is true when the nodes have the same managers. Only the collective interest of the global network is taken into consideration. Individual nodes follow all the recommendations of the routing protocols and so there is no conflict of interests in those networks.

However, in a multi-hop network without a central authority, the decision to cooperate becomes decentralized and, in extreme cases, each node is autonomous and decides only for its own best interests. This creates conflict between self and collective interests. A selfish node is tempted to drop packets from other nodes to save energy and bandwidth. For instance, in Vehicular Ad Hoc Network (VANET), each vehicle is equipped with a node to provide communication between nearby vehicles. Nodes in VANET are autonomous. In fact, many other applications in the future will require autonomous devices to interact, and cooperation will be the first problem to solve in such networks.

There are several approaches to motivate cooperation in ad hoc networks. In models using virtual currency [28-29], the sender must pay intermediate nodes to relay packets. Intermediate nodes accumulate the virtual currency and use it in the future to send their own packets. Thus, nodes at the center of the network accumulate more virtual currency compared to nodes at the border. Thus, a border node may not have enough virtual currency to send its packets. On the other hand, in models using a reputation system [31-34], nodes must relay packets to maintain a good reputation. Nodes with bad reputation are eliminated [31, 33-34]. Therefore, maintaining a good reputation is more costly for a node at the center while being cheap for a border node. In fact, no reputation mechanism can guaranty full cooperation in the network if for some nodes, the cost to maintain a good reputation is higher than the benefit. Other approaches use game theory [1-3, 6-11, 13-17] and evolutionary game theory [4-5] to model cooperation. However, the games designed and the strategies developed do not fully characterize the set of individually rational and feasible payoffs that directly depend on the traffic load distribution in the network.

Unequal traffic loads arise in the network because existing routing protocols in multi-hop networks are designed to choose the shortest available path between the sender and destination. The result is a higher traffic load at the center compared to the border of the network [53-54]. When the nodes are moving according to the random waypoint (RWP) mobility model [55], the nodes are concentrated in the center of the network creating an increase of traffic inequality compared to static nodes [56-57]. Accordingly, we will scrutinize traffic load balancing algorithms in the context of cooperation. However, the goal is not to discuss a specific load balancing algorithm.

The main contribution of this Chapter is to provide an analytical cost benefit analysis of cooperation in packet forwarding using game theory. We use a top down analytical approach starting from the global network level and translate the result to the local node level. Our analytical results allow for interpretation of selfish node cooperation in a general context. To the best of our knowledge, there is no paper that takes the unequal traffic load in networks into account when evaluating the proposed solution to solve the problem of cooperation. Finally, we recommend a *community enforcement* mechanism to sustain node cooperation.

The rest of the Chapter is divided as follows. In Section 3.2, we evaluate the exact traffic load at any node located at any point in a randomly deployed multi-hop network. Section 3.3 is committed to the general assumptions of this Chapter as well as Chapter 4, 5 and 6. Section 3.4 presents our game model. After, we use the results of Section 3.2 to provide an analytical cost benefit analysis of cooperation for any selfish rational node in the network in Section 3.5. To do that, we use a game theoretic approach. Section 3.6 is dedicated to our theoretical results. Section 3.7 concludes the Chapter.

3.2 Traffic Load Analysis

In this analysis, we start by assuming that a shortest path routing algorithm is used as in most routing protocol. We will examine this assumption later in this Section. We also presuppose a dense ad hoc network such that the packets from the sender to the destination follow an almost straight line. We start to analyze the traffic load distribution in a straight line to determine that of a rectangle followed by the special case of a square. Then, we determine an exact value of the traffic relayed by a node depending on its position. We start our analysis with a static ad hoc network before taking into account node mobility for MANET. We also examine the special case of WSNs.

A. Traffic load distribution in a straight line

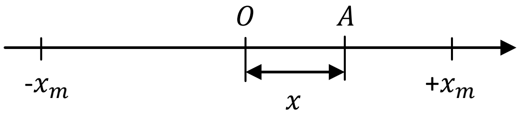

Figure 3.1 Line segment

Let us consider a straight line with center *O*, see Fig.3.1. Suppose that *n* static nodes are randomly and uniformly distributed on a segment $[-x_m, x_m]$ to form a dense ad hoc network. Consider point *A* on the line such that the Euclidean distance $OA = x$. The proportion of nodes to the left of *A* is $\frac{(x_m+x)}{2x_m}$. Similarly, the proportion of nodes to the right of *A* is $\frac{(x_m-x)}{2x_m}$. Let us consider a node *i* located in *A*. Node *i* is requested to forward a packet in multi-hop communication in the segment if the sender is located to the left of *A* and the destination is located to the right of *A* or vice versa. Therefore, the relative likelihood that node *i* is requested to forward a packet is proportional to $2 * \frac{(x_m+x)}{2x_m} * \frac{(x_m-x)}{2x_m}$. This means that the probability density function of the number of packets forwarded by a node *X* is in the form

$$f_X(x) = k\frac{(x_m + x)(x_m - x)}{x_m^2}.$$

k is a scaling factor.

Moreover, we must have

$$\int_{-\infty}^{\infty} f_X(x)dx = 1 \Rightarrow \int_{-x_m}^{x_m} k\frac{(x_m + x)(x_m - x)}{x_m^2}dx = 1.$$

Thus, $k = \frac{3}{4x_m}$. And finally,

$$f_X(x) = \frac{3}{4x_m^3}(x_m^2 - x^2). \tag{1}$$

This result is consistent with the result obtained in [57] where the primary analysis is the spatial node distribution of the random waypoint mobility model.

B. Traffic load distribution in a rectangle

As in the straight line, suppose that *n* static nodes are randomly and uniformly distributed in the rectangle to form a dense ad hoc network. The traffic generated can be decomposed into two types, which are, a horizontal traffic and a vertical traffic. Consider again a node *i* located in *A(x, y)*. Let us assume that the horizontal traffic is independent of the vertical traffic. Therefore, the joint probability density function $f_{X,Y}(x,y)$ of the number of packets forwarded is given by:

$$f_{X,Y}(x,y) = f_X(x) * f_Y(y) \Rightarrow$$

$$f_{X,Y}(x,y) = \frac{3}{4x_m^3}(x_m^2 - x^2) * \frac{3}{4y_m^3}(y_m^2 - y^2) \Rightarrow$$

$$f_{X,Y}(x,y) = \frac{9}{16x_m^3 y_m^3}(x_m^2 - x^2)(y_m^2 - y^2). \tag{2}$$

In WSNs, all data collected are transmitted to the sink. The sender destination pair of a packet is not random. The optimum position of the sink, to save energy and minimize end to end delay, is the center of the network. If the sink is unique, static, and located at the center, a similar reasoning shows that:

$$f_{X,Y}(x,y) = \frac{(x_m - |x|)(y_m - |y|)}{x_m^2 \cdot y_m^2}. \tag{3}$$

C. Expected Euclidean distance between two random nodes

The average distance traveled by a packet in an ad hoc network deployed in a rectangle is equivalent to the expected Euclidean distance between two random points in a rectangle. The exact analytical solution to this problem is evaluated in [58]. The result is given in Theorem 1.

Theorem 1: Let a and b be any positive real constants. Let $R = ([0,a] \times [0,b]) \cap \mathbb{R}^2$ designates a rectangle of dimension a and b. Let P and Q be independent random variables, each with uniform distribution over R. Let the Euclidean distance between P and Q be the random variable $D = d(P,Q)$. The expected value of D is given by:

$$E[D] = \frac{a^5 + b^5 - (a^4 - 3a^2b^2 + b^4)\sqrt{a^2+b^2}}{15a^2b^2}$$

$$+\frac{a^2}{6b} ln\left(\frac{b+\sqrt{a^2+b^2}}{a}\right) + \frac{b^2}{6a} ln\left(\frac{a+\sqrt{a^2+b^2}}{b}\right). \tag{4}$$

Remember that in our rectangle, $a = 2x_m$ and $b = 2y_m$.

In the special case of a square, $a = b$ and

$$E[D] = \frac{2+\sqrt{2}}{15} + \frac{ln(1+\sqrt{2})}{3} \approx 0.521405a = 1.04281x_m. \tag{5}$$

The value of $E[D]$ is not exactly the same in WSNs. In fact, $E[D]$ is evaluated for a random sender and destination pair whereas the destination is the sink in WSNs.

D. Number of packets under transmission in the network

Here we evaluate the total number of packets under transmission as in [56]. Let λ specify the average packet sending rate between any two nodes in the network. The total number of packets sent by a node i is$(n-1)\lambda$. Thus, the total sending rate of packets is $\Lambda = n(n-1)\lambda$. Let *d* be the constant transmission range of all nodes. The average number of hops a packet travels from the sender to the destination in a dense multi-hop route is $E[D]/d$. Let $1/\mu$ represent the transmission time of a packet. Thus, if the network is not congested or the queuing delay can be neglected, the average time a packet spends in the network is $\frac{E[D]}{(d\,.\mu)}$. From Little's law, the average number of packets under transmission in the network is:

$$\bar{N} = \frac{\Lambda. E[D]}{(d\,.\,\mu)} = \frac{n(n-1)\lambda E[D]}{(d\,.\,\mu)}. \tag{6}$$

E. Total traffic at a node

In any time unit, the total number of packets transmitted by a node has two components. The packets generated by the node itself and the packets forwarded from other nodes. Clearly, we have:

$$P_T = P_G + P_F. \tag{7}$$

P_T is the total number of packets transmitted per time unit

P_G is the total number of packets generated per time unit

P_F is the total number of packets forwarded per time unit

$$P_G = (n-1)\lambda. \tag{8}$$

P_G is the same for all nodes regardless of the node location. However, as explained before, the number of packets forwarded by a node P_F depends of the node position. The value of P_F is:

$$P_F(x,y) = \bar{N} * f_{X,Y}(x,y) = \frac{n(n-1)\lambda E[D]}{(d\,.\,\mu)} f_{X,Y}(x,y). \tag{9}$$

F. Node mobility

Node mobility has several effects on multi-hop network performance. The exact node mobility is approximated by a mobility model to facilitate analytical analysis and implementation in network simulators. There are many mobility models for MANET. However, starting with a uniform distribution of nodes over a domain as we assume in this Section, the node distribution over that domain does not always remain uniform in the long run. Therefore, the quantitative results about the probability density function of the number of packets forwarded by a node (1), (2), (3) and the expected Euclidean distance between sender and the destination nodes (4), (5), are modified according to the node mobility model. Nevertheless, all those results are preserved if the node mobility model maintains a uniform distribution of nodes.

The most commonly used mobility model is RWP first proposed in [55]. When the nodes move according to RWP, the node density at the center of the network is higher than the node density at the border [56-57]. As a result, the probability density function of the number of packets forwarded by a node is more condensed at the center of the network when compared to a uniform node distribution. Equivalently, the traffic load relayed by a node at the center of the network, when the nodes are moving according to RWP, is much higher than when the nodes are uniformly distributed.

G. Load balancing algorithm

Considering the result in this Section, obtained when using a shortest path routing algorithm, one clearly understands the importance of using a load balancing algorithm to reduce the traffic load at the center of the network. However, regardless of the approach, load balancing algorithms always increase the path length of packets and thus $E[D]$. Certainly, there is a tradeoff between load balancing and path length. The main goal of load balancing algorithms in the literature is to reduce battery consumption, end to end delay, and network congestion.

3.3 General Assuptions

We assume that each node is autonomous. We model a wireless multi-hop network as an arbitrary connected undirected graph *G (V, E)*. *V* is a set of *n* nodes. *E* is the set of edges. Node i and node j are elements of V. *(i, j)* is an element of E if and only if i is in the communication range of j. In this case, we also say that *i* is in the neighborhood of j. The nodes have the same communication range *d*. Therefore, the neighborhood relation is symmetric. We consider the communication channel to be bidirectional. We presume that nodes communicate via multi-hop communication if the sender is not in the communication range of the receiver. Thus, packets, from source to destination, are forwarded by intermediate nodes. When forwarding a packet, the total cost for a node in battery and bandwidth is β. When a node drops a packet as a result of defection, we suppose there is no cost or gain for that node. Packets are equal in size. We presume that the costs for receiving and listening are equally distributed in the network. Thus, those costs are ignored. No node trusts any other node but itself. Each node has only two choices: cooperate or defect. A node cooperates when it forwards a packet on time and defects when it does otherwise. We assume time is divided into time slots and that each time slot is just long enough to send and receive a packet if there is one to send or receive and to decide to cooperate or defect. Each node is equipped with an omnidirectional antenna and operates in promiscuous mode to monitor its neighbors using a mechanism similar to the watchdog [32]. These assumptions are valid for this Chapter as well as Chapter 4, 5, and 6.

3.4 Game Model

The number of nodes n is large enough to form a dense network. We neglect the traffic from direct communication between two neighbors. The value of the reward for the sender is v. To be clear, if a packet is dropped by an intermediate node, there is no reward for the sender or any other node. We consider that future payoffs are discounted by a common discount factor δ after every time unit and nodes maximize the average δ-discounted average payoff (19).

We assume that each node acts rationally. This means that each node is intelligent, selfish, and independently performs a cost benefit analysis to decide whether to forward a packet or not while interactively taking into account the decision of other nodes. Under this game model and the above assumptions, for a single interaction, we can see that the packet forwarding game generally has the following three properties:

1. For any number of cooperating nodes, the payoff of defectors is higher than the payoff of cooperators. In other words, Defect is the dominant strategy. This is because cooperators spend their energy to forward packets whereas defectors use the network freely.

2. The payoffs of cooperators and of defectors increase with the number of cooperating nodes. This is because more routes become available.

3. Lastly, the total payoff obtained by summing all the individual payoffs increase with the number of cooperating nodes. As we will see in the next Section, this last property does not always hold. Forwarding all packets is not always globally efficient.

This is an *n* persons Prisoner's Dilemma Game. Therefore, a rational node must defect in this game if the game is played one time. Moreover, if the game is repeated and the players know the end of the game, a backward induction argument shows that cooperation is impossible among rational players. However, if the interactions are repeated, cooperation can emerge if no player knows the end of the game. This is because a defection will prompt a future defection from other nodes and taking the future discounted payoff into account can make defection unprofitable.

3.5 Game Theoretic Analysis of Cooperation

In this Section, we perform a cost benefit analysis of cooperation. We determine under which conditions cooperation can be supported as equilibrium in the packet forwarding game. We first evaluate the global network payoff before evaluating the individual payoffs of different nodes.

A. Global Network payoff

Let us consider the exceptional case where all nodes in the network fully cooperate to forward packets and where all packets sent reach the destination. This consideration is only to quantify the total payoff of full cooperation. The sender's gain is *v* when a packet reaches the destination. In the long run, the number of packets sent equal the number of packets reaching the destination. In a time unit, the gain *G* of each node is:

$$G = P_G . \nu = (n-1). \lambda . \nu. \tag{10}$$

This gain is common for all nodes regardless of their positions. The total gainG_T in the network will be:

$$G_T = nG = n(n-1). \lambda . \nu = \Lambda . \nu. \tag{11}$$

On the other hand, the cost to forward a packet is *β*. The total cost per time unit C_T to relay all the packets is the total number of packets under transmission in the network times the cost to forward a packet. This means that:

$$C_T = \overline{N}.\beta = \frac{n(n-1)\lambda E[D].\beta}{(d\,.\,\mu)} = \frac{\Lambda.E[D].\beta}{(d\,.\,\mu)}. \qquad (12)$$

The global network payoff U_T will be the difference between the total gain and the total cost. Thus,

$$U_T = G_T - C_T = \Lambda.\nu - \frac{\Lambda.E[D].\beta}{(d\,.\,\mu)}. \qquad (13)$$

Instead of full cooperation, let us consider the case that all nodes defect. Obviously, the total cost to forward packets is zero and so is the total gain. Therefore, we have $U_T = 0$. In summary, full cooperation in a network is globally better than full defection if and only if:

$$n(n-1).\nu - \frac{n(n-1).E[D].\beta}{(d\,.\,\mu)} > 0 \text{ or } \frac{(d\,.\,\mu)}{E[D]} > \frac{\beta}{\nu}. \qquad (14)$$

Theorem 2: In a wireless multi-hop network, full node cooperation globally yields a higher payoff than full defection if and only if:

$$n(n-1).\nu - \frac{n(n-1).E[D].\beta}{(d\,.\,\mu)} > 0 \text{ or } \frac{(d\,.\,\mu)}{E[D]} > \frac{\beta}{\nu}. \qquad (14)$$

Remark 1: This is a general result. It is true for ad hoc networks as well as WSNs and is indifferent to node mobility. This result is also independent of the average packet sending rate between two nodes λ and the number of nodes *n*. Clearly, full cooperation can be globally efficient regardless of the number of nodes.

Remark 2: When (14) does not hold, we do not have an *n* persons Prisoner's Dilemma Game because the third condition specified in Section 3.4 is not respected. Surely, if (14) does not hold, no mechanism can permanently guarantee full cooperation among selfish nodes. This is because some individual nodes will find cooperation harmful. In other words, (14) is a necessary condition of full cooperation.

B. Individual node payoff

Suppose that all nodes cooperate. Let us consider a node *i* located in *A(x, y)* in a two dimensional area. The gain of node *i* is given in (10). The cost of cooperation for node *i* is:

$$C_i(x,y) = \frac{n(n-1)\lambda.E[D].\beta}{(d\,.\,\mu)} f_{X,Y}(x,y). \tag{15}$$

Note that those costs are not uniformly distributed in the network. The cost to forward packets is maximal at the center and is zero at the border. The payoff of node *i* is:

$$U_i(x,y) = (n-1).\lambda.v - \frac{n(n-1)\lambda.E[D].\beta}{(d\,.\,\mu)} f_{X,Y}(x,y). \tag{16}$$

Theorem 3: In a wireless multi-hop network, full node cooperation for a node *i* located in *A(x, y)* yields a higher payoff than full defection if and only if:

$$(n-1).\lambda.v - \frac{n(n-1)\lambda.E[D].\beta}{(d\,.\,\mu)} f_{X,Y}(x,y) > 0\,. \tag{17}$$

Remark 3: If (17) holds for all nodes, full node cooperation has an individually rational and feasible payoff for all nodes.

Remark 4: We observe that (14) can hold, but (17) does not hold for some nodes because of traffic load inequality in the network. In this case, it is essential to design a load balancing algorithm that minimizes the maximum traffic load and as a consequence may allow (17) to hold for all nodes. However, if (17) holds for all nodes, (14) must also hold. Finally, if in a multi-hop network, (17) initially holds for all nodes, a load balancing algorithm to force node cooperation is optional.

C. Strategy development

The strategies we develop in this Subsection should be able to force self interested nodes to achieve full network cooperation. Those strategies should also be consistent with the following characteristics of our network:

1. Autonomy: each node independently decides to cooperate or defect based on its self interest.

2. Local views: each node can perfectly monitor the behavior of its neighbors using a watchdog mechanism [32]. In other words, a node does not have a global view of the network and can decide solely based on local information available in its communication range. An effective strategy should rely only on self observation. In fact, a mechanism using second hand information to allow each node to have a global view of the network may be manipulated. Also, packets containing second hand information are costly and can be dropped.

3. Decentralization: there is no central authority. The network is distributed and self organized.

4. Mobility: nodes can be mobile or fixed. Node mobility reduces the discount factor in the game between two neighboring nodes because future interactions are less probable. Selfish nodes in MANET may avoid the center of the network, but if all nodes follow this strategy, it results in another dilemma.

All together, those four characteristics make cooperative behavior hard to achieve. There are two main means to constrain selfish nodes to cooperate that capitalize on long term relationships. Using *personal enforcement*, only the node that is a victim of defection punishes the defector. However, using a *community enforcement* mechanism [59], defection is punished not only by the victim, but also by any other node that observes the defection. In our model, the victim of defection is the sender of a packet. A packet can be dropped a few hops away or outside the communication range of the sender. Therefore, a personal enforcement mechanism is not robust in a distributed network given the four characteristics above. Without any credible punishment, the essential feature that motivates cooperation is lost and thus, cooperation is impossible.

A community enforcement mechanism to force node cooperation in the network is more robust. Each node should punish its neighbor after a defection regardless of whether or not that node is the victim. This way, each packet can be monitored hop by hop until the destination. Each node forwards the packet in fear of being punished by its own neighbor. This is even more effective if the nodes are static. In this scenario, one approach can be to analyze the network packet forwarding game as several two players game in each link *(i, j)*. Each node *i* will be involved in as many games as neighbors it has. However, the different games involving node *i* will not be independent and analyzing the correlation among those games will be an enigmatic problem. For instance, a node i with five neighbors can stop cooperation with three of those neighbors and continue to cooperate with the other two neighbors while sending the packets it generates through those two neighbors. This is one reason that makes the general equilibrium analysis of such network packet forwarding games complicated.

To avoid this complication, let us consider another strategy also based on community enforcement. Suppose that the nodes have two *types*. A node is of type *c* if it has never observed a defection in the past, and otherwise, it becomes type *d* forever. Let us also consider a strategy that requires each node to play its type. Therefore, if all nodes use this *contagious strategy*, a single defection spreads like an epidemic in the whole network. The ultimate choice a selfish node has to make is between cooperating forever and a future network collapses where all nodes defect forever. Cooperation is enforced in the network because each node fears to start the contamination process and destroy its future gain. This strategy is valuable in a network if each node can perfectly monitor the behavior of its neighbors. Next, we prove that this strategy is Subgame Perfect Nash Equilibrium (SPE) of the packet forwarding game if full cooperation has an individually rational and feasible payoff.

D. Equilibrium analysis

Theorem 4: The contagious strategy described above is a SPE if for all node i, (17) holds and,

$$\frac{C_i}{U_i} \leq \frac{\delta^{(1+H_{max})}}{1-\delta}. \tag{18}$$

H_{max} is the maximum number of hops between two nodes.

C_i and U_i are according to (15) and (16) respectively.

Proof: when (17) holds for all nodes, full cooperation has a feasible and individually rational payoff. Assuming (17) holds for all nodes; let us check if a type *c* player has an incentive to defect when not observing any defection. Let σ be our contagious strategy. Since we assume that the nodes maximize the average δ-discounted average, node i's expected payoff is:

$$V_i(\sigma) = (1-\delta)\sum_{t=0}^{\infty} \delta^t U_i(a^t). \tag{19}$$

a^t is the players' action at time *t*

Let σ' be a strategy that deviates from σ only once. According to the one-shot deviation principle of dynamic programming, σ is a SPE if and only if:

$$V_i(\sigma) \geq V_i(\sigma_i^{'}, \sigma_{-i})$$

$$\Rightarrow (1-\delta)\sum_{t=0}^{\infty} \delta^t U_i(\sigma) \geq (1-\delta)\sum_{t=0}^{\infty} \delta^t U_i(\sigma_i^{'}, \sigma_{-i})$$

$$\Rightarrow\ U_i\left(\sigma_i^{'}, \sigma_{-i}\right) - U_i(\sigma) \leq \sum_{t=1}^{\infty} \delta^t U_i(\sigma) - \sum_{t=1}^{\infty} \delta^t U_i(\sigma_i^{'}, \sigma_{-i})$$

$$\Rightarrow\ C_i \leq \frac{\delta U_i}{1-\delta} - \sum_{t=1}^{\infty} \delta^t U_i(\sigma_i^{'}, \sigma_{-i}). \tag{20}$$

We can derive the maximum value of the second term in the right hand side of (20). To do that, we consider a scenario in which after deviation, the packets from the deviating node are not dropped until full network contamination. We assume a dense network and uniform contamination in all directions. We consider also a traffic load such that each node sends at least one packet to each of its neighbors in each time unit. Thus, the maximum number of time units until full network contamination is $H_{max} = \frac{D_{max}}{d}$. (Assuming static nodes, but we have a similar result for mobile nodes)

D_{max} is the maximum distance in the network.

d is the communication range.

Therefore,

$$\sum_{t=1}^{\infty} \delta^t U_i\left(\sigma_i^{'}, \sigma_{-i}\right) \leq U_i(\delta + \delta^2 + \cdots + \delta^{H_{max}})$$

$$\Rightarrow \sum_{t=1}^{\infty} \delta^t U_i\left(\sigma_i^{'}, \sigma_{-i}\right) \leq \frac{U_i \delta (1 - \delta^{H_{max}})}{1 - \delta}. \tag{21}$$

Combining (20) and (21), a type *c* player keeps cooperating if:

$$C_i \leq \frac{\delta U_i}{1 - \delta} - \frac{U_i \delta (1 - \delta^{H_{max}})}{1 - \delta} = \frac{U_i \delta^{(1 + H_{max})}}{1 - \delta}$$

$$\Rightarrow \frac{C_i}{U_i} \leq \frac{\delta^{(1 + H_{max})}}{1 - \delta}. \tag{18}$$

■

We can see that (18) must hold for δ close enough to 1.

3.6 Theoretical Results

The utility function (16) of a node $U_i(x, y)$as a function of its coordinate *(x, y)* in a static ad hoc network is represented in Fig.3.2. We can see that $U_i(x, y)$ has a minimum at $x = y = 0$, the center of the network. For the indicated parameters, $U_i(x, y) > 0$ for all nodes and full cooperation has an individually rational and feasible payoff. If we decrease the gain v from 6 to 5, we will have $U_i(x, y) < 0$ for some nodes at the center and those nodes cannot cooperate. They will cause a massive traffic disruption in the network since a relatively greater number of routes go through those nodes compared to others.

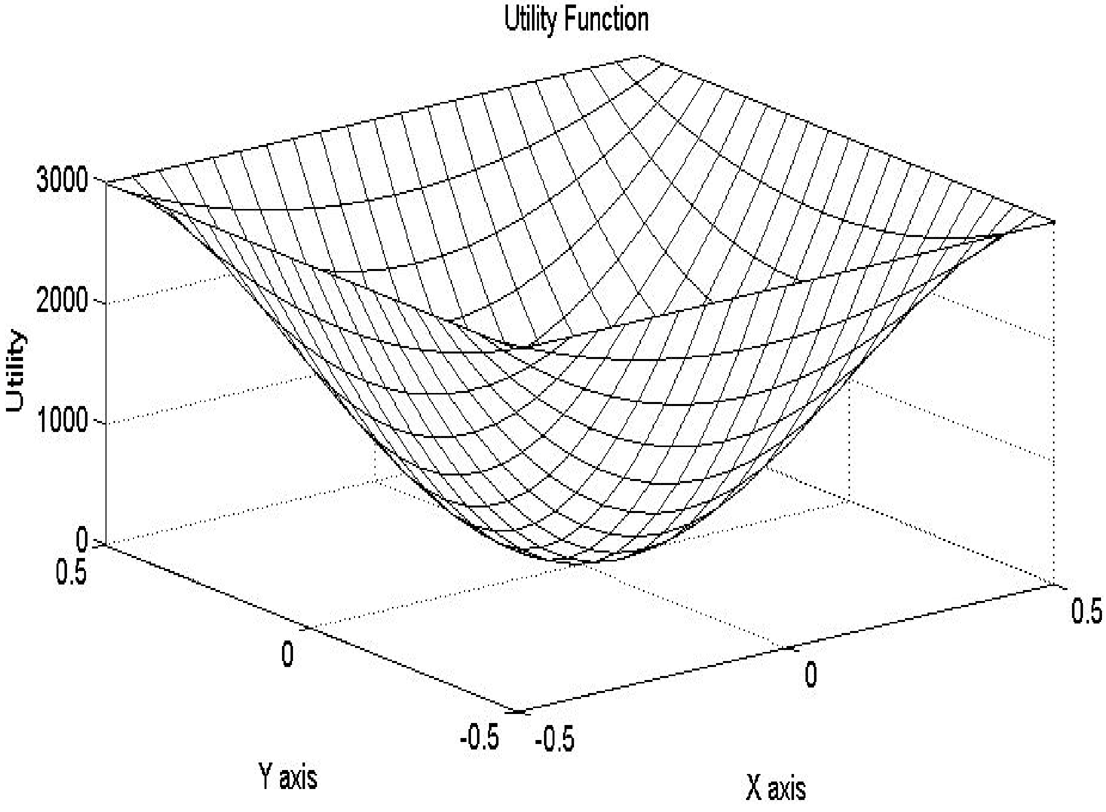

Figure 3.2. ***$U_i(x,y)$*****: *x_m=y_m=0.5; d=0.1; n=500; λ=1; β=1; v=6; μ=1000.***

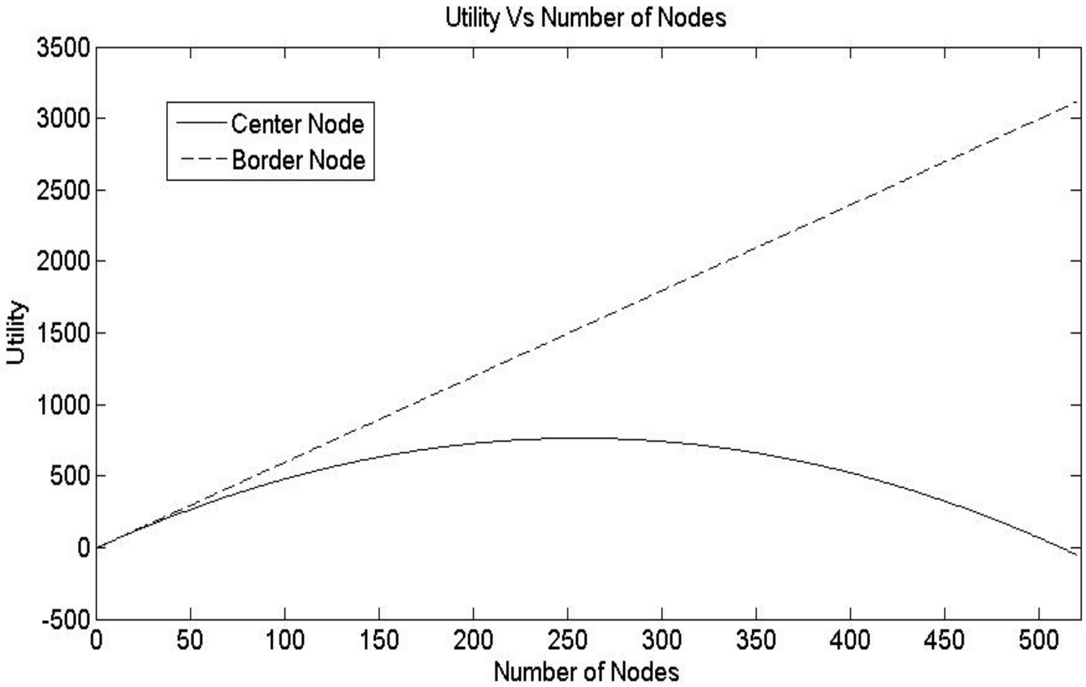

Figure 3.3. ***$U_i(n)$*: *x_m=y_m=0.5; d=0.1; λ=1; β=1; ν=6; μ=1000.***

The utility function (16) as a function of the number of nodes in a static ad hoc network is represented in Fig. 3.3. $f_{X,Y}(x,y) = 0$ for a border node and makes $U_i(n)$a linear and increasing function of n. However, for any other node, $f_{X,Y}(x,y) > 0$ and $U_i(n)$ is a second degree polynomial. Starting from one node, as the number of nodes increases, there is an optimum number of nodes that maximize the payoff followed by a decrease. Finally, we have a maximum number of nodes above which $U_i(n)$becomes negative and cooperation will be impossible for node i. Therefore, in a static network without a load balancing algorithm, cooperation ultimately breaks down as the number of nodes increases.

3.7 Conclusion

We have analyzed the consequences of traffic load inequality on cooperation in multi-hop networks. We specify the conditions under which cooperation has an individually rational and feasible payoff. We show that *community enforcement* mechanisms can compel selfish nodes to cooperate using only local information. We prove that, without a load balancing algorithm in a static network, cooperation breaks down in a large distributed network.

We assumed perfect monitoring in our model. In Chapter 5, we will analyze cooperation under imperfect monitoring. In the future, we will also consider network congestion and queuing delay.

CHAPTER 4

MITIGATING SELFISH MISBEHAVIOR IN AUTONOMOUS WIRELESS MULTI-HOP NETWORK USING STOCHASTIC GAME THEORY

Cooperation is a critical issue in autonomous multi-hop networks. This is because there is no infrastructure. Each node in the network is at the same time a terminal and a router. Moreover, cooperation to forward a packet from other nodes is costly. As such, it is not in the best interest of an individual node to cooperate. In this Chapter, we propose Punish Only *n* Times (PONT): a distributed algorithm based on SGT that can force intelligent selfish autonomous nodes to cooperate without a contract in a multi-hop network.

4.1 Introduction

Multi-hop wireless communication includes WSN, ad hoc networks, and peer-to-peer (P2P) networks. They are networks without infrastructure. Each node of the network is at the same time a router and a terminal. Therefore, communication between a sender and a receiver that are not in range of each other relies on the cooperation of intermediate nodes to forward packets. In many application domains, including some military applications, nodes have the same manager.

However, this is not the case in other applications, where sensors are embedded in any device to form a ubiquitous computing environment, such as health monitoring networks and VANET. In the latter scenario, each vehicle is equipped with a sensor, but sensors from different vehicles are not managed by the same authority. Here, sensors are autonomous but still have to cooperate and forward packets from others. In this situation, cooperation and trust can no longer be assumed as “de facto”. A node, especially a selfish node, could send its own packets and not forward packets from others. If all nodes adopt such a solution, it will be a costly one and the network will collapse because basic functions such as packet forwarding cannot be accomplished.

In this Chapter, we present PONT, a model based on SGT. To the best of our knowledge, this is one of the first works that uses SGT to model cooperation in multi-hop networks. We believe that this is one of the most promising approaches. Especially since network interactions are random processes, nodes are becoming exceptionally intelligent and game theory formalizes rational nodes collaboration. We present the conditions under which autonomous selfish nodes cooperate and support our result by a mathematical proof and MATLAB plots. This Chapter deals only with selfish nodes and proposes an algorithm to force them to cooperate although any model able to detect and eliminate faulty and malicious nodes can be incorporated.

The game model in this work is similar to those using the Prisoners’ Dilemma game [4-5, 10-12]. However, we introduced a stochastic dimension to eliminate the unrealistic assumption that each node has packets to send in each time slot or that any two neighbors have uniform network traffic demand. Also, our model does not reward intermediate nodes.

The next Section presents our stage game model. Section 4.3 contains the analysis of the repeated game. Section 4.4 exposes our theoretical results and Section 4.5 concludes the Chapter.

4.2 Stage Game Model

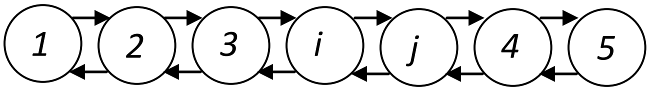

Figure 4.1: Simple Network illustration.

We use the simple bidirectional static ad hoc network in Fig. 4.1 to illustrate our model. The general assumptions of Section 3.3 apply here. The network consists of seven nodes in a straight line. We analyze the interactions between two nodes, node *i* and node *j*. Node *i* needs the cooperation of node *j* to relay its packets to node *4* and node *5*. On the other hand, node *j* needs the cooperation of node *i* to relay its packets to nodes *1, 2* and *3*. In general, each node in a network is involved in a packet forwarding game with each of its neighbors.

In some peer-to-peer applications, such as file downloading, the destination is interested in receiving packets from the sender. However, in other applications, such as advertisement, the sender is interested in having its packets reach the destination. In both cases, the intermediate nodes are not rewarded but are required to spend energy to relay packets. For this reason, this model does not reward intermediate nodes in multi-hop communication but punishes defecting nodes. We present the case of applications in which only the sender is rewarded, but all results can easily be transposed to the cases that reward only the destination or both the sender and the destination. The cost to relay a packet is β.

Node *i* requests node *j* to relay two types of packets. The first type are packets from node *i* itself and the second type are from nodes *1, 2* and *3*. When node *i* sends a packet originating from node *1, 2 or 3* to node *j* and node *j* relays that packet, node *i* is not rewarded because node *i* is a relay node in this scenario. However, if node *j* relays a packet from node *i* to node *4* or *5*, node *i* is rewarded and we can distinguish two cases. In the first case, where the destination is node *4*, node *i* gets the full reward price λ. However, in the second case, where the destination is node *5*, node *i* gets a partial reward price γ. In this second case, a packet from node *i* needs to be relayed by both node *j* and node *4* to reach the destination node *5* to give the full reward price λ to node *i*. The partial reward γ represents the increase in the expectation to get the reward price λ if node *j* and all the other intermediate nodes relay the packet until the destination.

A node cooperates if it relays a packet on time regardless of the type of packet, this means regardless of if the node that observes the packet being forwarded originated the packet or not. A node defects otherwise. We assume that time is divided into time slots. Each time slot is just long enough to send and/or receive one packet. We presuppose that sending or receiving a packet in a time slot is a Bernoulli process. The probability that node *i* requests node *j* to relay a packet in a time slot is p_i and vice versa. The probability that node *i* is the originator of a packet given that node *i* requests node *j* to relay a packet in a time slot is q_i and vice versa. Note that the possibility that node *i* can request node *j* to relay a packet that originates from node *1*, *2* or *3* will impact q_i. Moreover, note that q_i is generally small for center nodes compared to border node. Our model can easily capture the consequences of unequal traffic load on cooperation as developed with more details in the last Chapter [1].

There will be three possibilities for each node in each time slot. For instance, the possibilities for node i will be:

1- Send its own packet to node j which happens with probability $p_i q_i$

2- Forward a packet from another node (node 1, 2, 3) to node j with probability $p_i(1 - q_i)$

3- Not have any packet to send which happens with probability $(1 - p_i)$

There are nine (3^2) possible interactions between nodes i and j in a time slot. Each possibility will constitute a state of our stochastic game. The nine states are shown in Table 4.1. The 1 and 0 in Table 4.1 represent the Success or Failure in a time slot of the Bernoulli process. We have 16 lines to represent all the 2^4 combinations of events from p_i, p_j, q_i, q_j. Note that two or four lines can give us the same state. Also, when a node does not have a packet to send in a time slot, there is no need to differentiate if that node is a sender or a forwarder.

Since we have a Bernoulli process, the state in the next time slot is independent of the current or past state. Moreover, the states are not defined by the action of the players but by discrete events (p_i, p_j, q_i, q_j) independent from one time slot to the next. Consequently, our game model will be a repeated asymmetric game with random states.

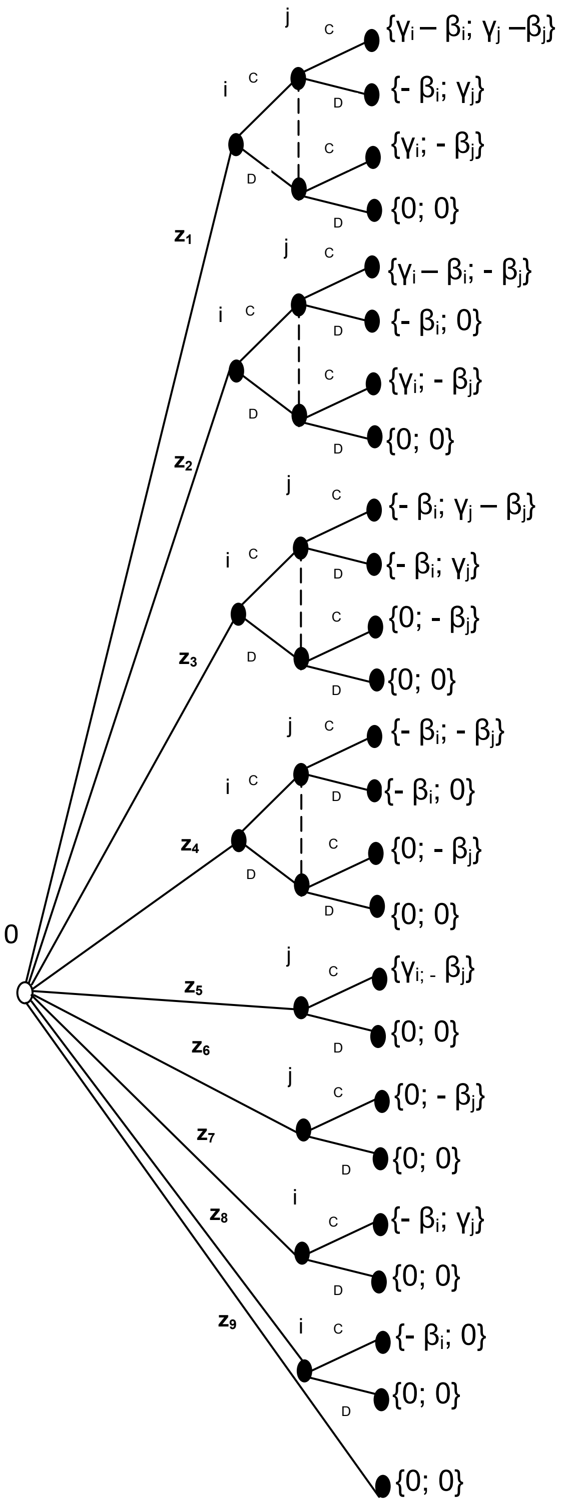
j
C
$\{\gamma_i - \beta_i; \gamma_j - \beta_j\}$
i
C
D
$\{- \beta_i; \gamma_j\}$
C
$\{\gamma_i; - \beta_j\}$
D
D
$\{0; 0\}$
z_1
j
C
$\{\gamma_i - \beta_i; - \beta_j\}$
i
C
D
$\{- \beta_i; 0\}$
C
$\{\gamma_i; - \beta_j\}$
D
D
$\{0; 0\}$
z_2
j
C
$\{- \beta_i; \gamma_j - \beta_j\}$
i
C
D
$\{- \beta_i; \gamma_j\}$
C
$\{0; - \beta_j\}$
z_3
D
D
$\{0; 0\}$
j
C
$\{- \beta_i; - \beta_j\}$
i
C
D
$\{- \beta_i; 0\}$
C
$\{0; - \beta_j\}$
z_4
D
D
$\{0; 0\}$
0
j
C
$\{\gamma_i; - \beta_j\}$
z_5
D
$\{0; 0\}$
z_6
j
C
$\{0; - \beta_j\}$
z_7
D
$\{0; 0\}$
i
C
$\{- \beta_i; \gamma_j\}$
z_8
D
$\{0; 0\}$
z_9
i
C
$\{- \beta_i; 0\}$
D
$\{0; 0\}$
$\{0; 0\}$

Figure 4.2: Stage game model in extensive form

TABLE 4.1: PROBABILITY DISTRIBUTION OF THE STOCHASTIC GAME STATES

Line	p_i	p_j	q_i	q_j	Probability		state
1	1	1	1	1	$p_i p_j q_i q_j$	z_1	1
2	1	1	1	0	$p_i p_j q_i (1 - q_j)$	z_2	2
3	1	1	0	1	$p_i p_j (1 - q_i) q_j$	z_3	3
4	1	1	0	0	$p_i p_j (1 - q_i)(1 - q_j)$	z_4	4
5	1	0	1	1	$p_i (1 - p_j) q_i$	z_5	5
6	1	0	1	0			
7	1	0	0	1	$p_i (1 - p_j)(1 - q_i)$	z_6	6
8	1	0	0	0			
9	0	1	1	1	$(1 - p_i) p_j q_j$	z_7	7
10	0	1	1	0	$(1 - p_i) p_j (1 - q_j)$	z_8	8
11	0	1	0	1	$(1 - p_i) p_j q_j$	z_7	7
12	0	1	0	0	$(1 - p_i) p_j (1 - q_j)$	z_8	8
13	0	0	1	1	$(1 - p_i)(1 - q_j)$	z_9	9
14	0	0	1	0			
15	0	0	0	1			
16	0	0	0	0			

The stage game in extensive form is shown in Fig. 4.2. Table 4.1 shows the probability distribution of the game states. The state of the game in a time slot is random. We characterize that by a nature player in decision node 0. From that decision node, each branch corresponds to a state with the associated probability.

When the two nodes have a packet to send, they make decisions simultaneously. This is represented in the first four states of Fig 4.2 by the dotted line between the two decision nodes of node j. In states 5, 6, 7 and 8, only one node has a decision to make because the other node does not have any packet to forward and therefore does not decide to cooperate or defect.

Let us transform the game into its strategic form for further analysis. Table 4.2 shows the result after all calculations. In fact, in each time slot, the payoffs have two parts. For node i those parts are: the expected reward from node j cooperation $p_i q_i \gamma_i$ and the expected cost $p_j \beta_i$ to relay a packet from node j. If node j decides to cooperate in a time slot, the probability that node i sends its own packet to be relayed is $p_i q_i$ and therefore, the expected gain in that time slot for node i is $p_i q_i \gamma_i$. Also, if node i decides to cooperate, the probability that node j sends a packet to node i is p_j. Thus, node i's expected cost is $p_j \beta_i$. All the payoffs can be evaluated this way.

The strategic form of the game confirms the result from the extensive form. Defect is the dominant strategy of the stage game. Mutual cooperation is Pareto efficient if we have:

$$p_i q_i \gamma_i - p_j \beta_i > 0, \tag{1}$$

And

$$p_j q_j \gamma_j - p_i \beta_j > 0. \tag{2}$$

TABLE 4.2: STAGE GAME IN STRATEGIC FORM

		Node j	
		Cooperate	Defect
Node i	Cooperate	$\{p_i q_i \gamma_i - p_j \beta_i;\ p_j q_j \gamma_j - p_i \beta_j\}$	$\{-p_j \beta_i;\ p_j q_j \gamma_j\}$
	Defect	$\{p_i q_i \gamma_i;\ -p_i \beta_j\}$	$\{0;\ 0\}$

We can say that the stage game in strategic form is a Prisoners' Dilemma game with random payoffs or a Stochastic Prisoners' Dilemma (SPD) game.

A recurrent and important parameter of this model is the expected cost over gain ratio. Let us call this ratio x. We have for any node i of the network:

$$x_i = \frac{\beta_i}{\gamma_i}. \tag{3}$$

When we normalize the payoff with that of mutual cooperation, we have the game in Table 4.3. We suppose that $g_i, g_j > 0$ since otherwise, mutual cooperation is not Pareto efficient. An allocation is Pareto efficient if there is no other allocation that can make at least one individual better off without making any other individual worse off. Fig. 4.3 gives the high level description of PONT for node i playing against an opponent, node j. The next Section illustrates this algorithm and shows that it is a Nash equilibrium.

TABLE 4.3: STAGE GAME WITH NORMALIZED PAYOFF

		Node j	
		Cooperate	Defect
Node i	Cooperate	$\{1;1\}$	$\{-g_i; 1+g_j\}$
	Defect	$\{1+g_i; -g_j\}$	$\{0; 0\}$

$$g_i = \frac{p_j\beta_i}{p_i q_i \gamma_i - p_j\beta_i} > 0 \text{ and } g_j = \frac{p_i\beta_j}{p_j q_j \gamma_j - p_i\beta_j} > 0. \tag{4}$$

4.3 Repeated Game Analysis

The last Section shows us that the only equilibrium of our stage game is when both players defect. This is not beneficial in a multi-hop network. The ultimate result will be the collapse of the network. In this Section, we will analyze the conditions under which repeating this stage game indefinitely will yield the nodes that will always cooperate just from the expectation of future gain. We assume that each node has a unique identity that cannot be falsified. We also assume that the “sender address” of a packet cannot be falsified.

```
Start by cooperation, $C_i$
If ($C_i$, $C_j$) then
  $C_i$ in the next time slot node $i$ decides (state 1, 2, 3, 4, 7, 8)
Else
  If ($C_i$, $D_j$) then
    Play $D_i$ $n$ times in state 1, 3, 7 and $C_i$ in states 2, 4, 8 then
      While playing $D_i$,
        Count the number $b$ of $D_j$
      End
    If b=0
      Play $C_i$
    Else
      Play $D_i$ forever
    End
  End
  If ($D_i$, $C_j$) then
    While node $j$ play $D_j$ $n$ times in state 1, 2, 5 and $C_j$ elsewhere
      Node $i$ play $C_i$ in all states it can decide
    End
  End
  If ($D_i$, $D_j$) then
    While node $i$ play $D_i$ $n$ times in state 1, 3, 7 and $C_i$ elsewhere
      Node $j$ plays $D_j$ $n$ times in state 1, 2, 5 and $C_j$ elsewhere
    End
  End
End
```

Figure 4.3: High level description of PONT for node *i* playing against *j*

We assume perfect monitoring, no error in perception and no error in implementation. This means for instance that when node i defects; node i knows that it defected and its neighbor node j knows that node i defected with no ambiguity. We will relax this assumption in the next Chapter. We consider that sending or receiving a packet is a Poisson process and we use its discrete time version, namely the Bernoulli process. Each node is autonomous, selfish, and rational. This means that each node acts only for its own self-interest. Lastly, we suppose that all nodes discount future payoff by a common discount factor δ with $0 \leq \delta < 1$ and want to maximize the expected δ-discounted average of their sequence of payoffs (8). We will first present our distributed algorithm, PONT. Then, we will show that PONT is a SPE of the game.

4.3.1 PONT Algorithm and Description

PONT starts playing cooperate and continues playing cooperate until the opponent defects. If the opponent defects, PONT defects n times in the states that the opponent is the originator of the packet, but PONT cooperates in the other states where the opponent is the forwarder. After the n defections, PONT resumes cooperation. Any subsequent defection of the opponent when PONT is still punishing a previous defection causes PONT to punish forever. However, PONT can be extended to not punish subsequent defections forever but for a limited number of time slots. The number of punishments n is a parameter that can be adapted. It must be common knowledge among players. As we will see later, an increase in the value of n makes punishment more severe, discourages opportunistic deviation, but also decelerates or discourages the return to mutual cooperation in case of an unavoidable defection caused by network congestion. We present here the general case when n can be any natural number but in practice, the optimum number of punishments n must be the smallest number large enough to discourage opportunistic deviations taking into account different parameters such as $p_i, p_j, q_i, q_j, x_i, x_j,$ and δ. Note that PONT does not randomize and is therefore a pure strategy.

Let us mention three important properties of PONT here. First, PONT defects by dropping only the packets originating from the opponent. This guarantees that the opponent, after a defection, will not send *n* packets for which it is the forwarder, knowing they will be dropped, and then not directly lose anything after a defection. This attribute makes the cooperative equilibrium more stable because it assures that the network packets from other nodes are not affected by the punishment. Second, PONT guarantees that two players come back to mutual cooperation after an imperative defection. A node can be imposed to defect during network congestion, for instance. When that happens, the defecting node is aware of it, accepts the *n* punishments and then the two nodes resume mutual cooperation. If both PONT players defect simultaneously, they punish each other and come back to mutual cooperation. Third, a node that is playing PONT must punish after a defection even though it is not the originator of the dropped packet. Since punishment consists of dropping *n* packets originating from the defecting node, and since dropping a packet is not costly, selfish nodes would not withhold punishment regardless of whether they originated the packet or not. This feature is particularly interesting in a distributed multi-hop network for two reasons. First, no node can monitor its own packets until a destination located several hops away. Second, even if it was possible for a node to monitor its own packets until the destination, the opportunity for that node itself to punish the node that dropped the packet a few hops away will be rare and make cooperation among nodes more difficult.

4.3.2 Proof of PONT Cooperative Equilibrium

***Theorem 1*:** PONT is a SPE of the stochastic packet forwarding game if for any two neighbor nodes in the network, node *i* and node *j*, we have:

$$\frac{\mathrm{p}_i \mathrm{q}_i}{\mathrm{p_j}} > x_i, \tag{5}$$

And

$$\frac{\mathrm{p}_j \mathrm{q}_j}{\mathrm{p}_j} > x_j. \tag{6}$$

And for any node *i* in the network, we have:

$$\left(\frac{\delta p_i q_i}{1-\delta(1-p_i q_i)}\right)\left\{\frac{1-\left(\frac{\delta p_i q_i}{1-\delta(1-p_i q_i)}\right)^n}{1-\left(\frac{\delta p_i q_i}{1-\delta(1-p_i q_i)}\right)}\right\} \geq x_i. \tag{7}$$

Proof:

According to the Folk Theorem and the stage game in strategic form in Table 4.2, we can support (C_i, C_j) as equilibrium if and only if the expected utility of (C_i, C_j) for each node is higher than the expected utility of (D_i, D_j). This means (1) and (2) are respected and therefore (5) and (6) hold.

Considering that (5) and (6) are respected, cooperation is then Pareto efficient. We use the one-shot deviation principle [60] to show that PONT is a SPE when inequality (7) is true. In fact PONT is a SPE if and only if there is no profitable one-shot deviation. We do not need to check all alternative strategies. We need only compare the equilibrium payoff to that of an alternative strategy that deviates only once and comes back to the equilibrium strategy [60].

Let σ be our strategy, PONT. Since it is assumed that the players maximize the expected δ-discounted average, player i's expected payoff is:

$$U_i(\sigma) = E^{\sigma}\{(1-\delta)\sum_{t=0}^{\infty}\delta^t u_i(s^t, a^t)\}. \tag{8}$$

s^t is the state at time t

a^t is the players' action at time t

E^{σ} represents the expected gain of strategy σ.

Let σ' be a strategy that deviates from σ only once.

σ is a SPE if and only if:

$$U_i(\sigma) \geq U_i\left(\sigma_i^{'}, \sigma_{-i}\right) \tag{9}$$

$$\Rightarrow E\{(1-\delta)\sum_{t=0}^{\infty}\delta^t u_i(\sigma)\} \geq E\{(1-\delta)\sum_{t=0}^{\infty}\delta^t u_i(\sigma_i^{'}, \sigma_{-i})\}$$

$$\Rightarrow E\{(1-\delta)u_i(\sigma)+(1-\delta)\sum_{t=1}^{\infty}\delta^t u_i(\sigma)\}$$

$$\geq E\{(1-\delta)u_i\left(\sigma_i^{'},\sigma_{-i}\right)+(1-\delta)\sum_{t=1}^{\infty}\delta^t u_i(\sigma_i^{'},\sigma_{-i})\}$$

$$\Rightarrow\ E\{\sum_{t=1}^{\infty}\delta^t u_i(\sigma)-\sum_{t=1}^{\infty}\delta^t u_i(\sigma_i^{'},\sigma_{-i})\}\geq E\{u_i\left(\sigma_i^{'},\sigma_{-i}\right)-u_i(\sigma)\}$$

$$\Rightarrow E\left\{\sum_{t=1}^{\infty}\delta^t\left[u_i(\sigma)-u_i\left(\sigma_i^{'},\sigma_{-i}\right)\right]\right\}\geq E\{u_i\left(\sigma_i^{'},\sigma_{-i}\right)-u_i(\sigma)\}. \qquad (10)$$

The first term of inequality (10) represents the expected future loss and the second term is the current gain from deviation of the equilibrium or defection. Inequality (10) is intuitive and simple. From (10), a strategy of the stochastic game is a SPE if the expected future loss exceeds the current gains from opportunistic deviation.

From the extensive form game of Fig. 4.2, in any state that node *i* can make a decision and deviate from cooperate to defect, the associated gain is β_i. Thus,

$$E\{u_i\left(\sigma_i^{'},\sigma_{-i}\right)-u_i(\sigma)\}=\beta_i. \qquad (11)$$

Now, let us evaluate the future loss from the time of defection. In fact, in (10)

$$u_i(\sigma)-u_i\left(\sigma_i^{'},\sigma_{-i}\right)=\begin{cases}\gamma_i\ if\ j\ \ drop\ a\ packet\ from\ i\\ \qquad 0\ otherwise\end{cases} \qquad (12)$$

As explained in the game model, a node *i* originates a packet in a time slot with probability p_iq_i. Then, the opponent node *j* will drop the packet successfully in a time slot with probability p_iq_i. The probability distribution of the number of trials in a sequence of Bernoulli trials needed to get a specified number (integer-valued) of success is a *Pascal* distribution. After a defection, a PONT player drops only the next *n* packets originating from the defecting node. Thus, dropping *n* packets is a $Pascal(n, p_i q_i)$ process. Specifically, the probability mass function of this *Pascal* distribution will be:

$$f(t) = \Pr(X = t) = C_{t-1}^{l-1}(1 - p_i q_i)^{t-l}(p_i q_i)^l. \tag{13}$$

X is the total number of time slots needed to drop *l* packets.

Therefore,

$$E\left\{\sum_{t=1}^{\infty} \delta^t \left[u_i(\sigma) - u_i\left(\sigma_i^{'}, \sigma_{-i}\right)\right]\right\} = \sum_{l=1}^{n}\sum_{t=1}^{\infty} \delta^t \gamma_i C_{t-1}^{l-1}(1 - p_i q_i)^{t-l}(p_i q_i)^l \tag{14}$$

$$= \gamma_i \sum_{l=1}^{n} \frac{(p_i q_i)^l}{(1 - p_i q_i)^l} \sum_{t=1}^{\infty} \delta^t C_{t-1}^{l-1}(1 - p_i q_i)^t$$

$$= \gamma_i \sum_{l=1}^{n} \frac{(p_i q_i)^l}{(1 - p_i q_i)^l} \sum_{t=1}^{\infty} C_{t-1}^{l-1}[\delta(1 - p_i q_i)]^t$$

$$= \gamma_i \sum_{l=1}^{n} \frac{(p_i q_i)^l}{(1 - p_i q_i)^l} [\delta(1 - p_i q_i)] \sum_{t=1}^{\infty} C_{t-1}^{l-1}[\delta(1 - p_i q_i)]^{t-1}$$

$$= \gamma_i \sum_{l=1}^{n} \frac{(p_i q_i)^l}{(1 - p_i q_i)^l} [\delta(1 - p_i q_i)] \sum_{t=0}^{\infty} C_t^{l-1}[\delta(1 - p_i q_i)]^t$$

$$= \gamma_i \sum_{l=1}^{n} \frac{(p_i q_i)^l}{(1 - p_i q_i)^l} [\delta(1 - p_i q_i)] \left\{\frac{[\delta(1 - p_i q_i)]^{l-1}}{(1 - [\delta(1 - p_i q_i)])^l}\right\}.$$

We have the last equality because (15) holds for a fixed *l* and any number *y*,

$$\sum_{t=0}^{\infty} C_t^l y^t = \frac{y^l}{(1-y)^{l+1}}. \tag{15}$$

Then

$$\gamma_i \sum_{l=1}^{n} \frac{(p_i q_i)^l}{(1-p_i q_i)^l}[\delta(1-p_i q_i)]\left\{\frac{[\delta(1-p_i q_i)]^{l-1}}{(1-[\delta(1-p_i q_i)])^l}\right\} = \gamma_i \sum_{l=1}^{n}\left(\frac{\delta p_i q_i}{1-\delta(1-p_i q_i)}\right)^l$$

$$= \gamma_i \left(\frac{\delta p_i q_i}{1-\delta(1-p_i q_i)}\right)\left\{\frac{1-\left(\frac{\delta p_i q_i}{1-\delta(1-p_i q_i)}\right)^n}{1-\left(\frac{\delta p_i q_i}{1-\delta(1-p_i q_i)}\right)}\right\}.$$

Thus, (10) implies

$$\gamma_i \left(\frac{\delta p_i q_i}{1-\delta(1-p_i q_i)}\right)\left\{\frac{1-\left(\frac{\delta p_i q_i}{1-\delta(1-p_i q_i)}\right)^n}{1-\left(\frac{\delta p_i q_i}{1-\delta(1-p_i q_i)}\right)}\right\} \geq \beta_i, \tag{16}$$

And finally

$$\left(\frac{\delta p_i q_i}{1-\delta(1-p_i q_i)}\right)\left\{\frac{1-\left(\frac{\delta p_i q_i}{1-\delta(1-p_i q_i)}\right)^n}{1-\left(\frac{\delta p_i q_i}{1-\delta(1-p_i q_i)}\right)}\right\} \geq x_i \tag{7}$$

■

Remark 1: The left part of (7) is a geometric sequence with common ratio $\Delta = \left(\frac{\delta p_i q_i}{1-\delta(1-p_i q_i)}\right)$. The expected loss incurred by node *i* when node *j* drops the *l*th packet is $\gamma_i \Delta^l$. The left term of (16) is thus the total expected loss from the *n* dropped packets. Equation (16) can also be written as:

$$\gamma_i \Delta^1 + \gamma_i \Delta^2 + \gamma_i \Delta^3 + \gamma_i \Delta^4 + \cdots + \gamma_i \Delta^n \geq \beta_i. \tag{17}$$

Equation (17) discloses the fact that the future losses of a node after a single deviation from the equilibrium strategy are discounted by Δ instead of δ. We call Δ the adjusted discount factor.

4.4 Theoretical Results

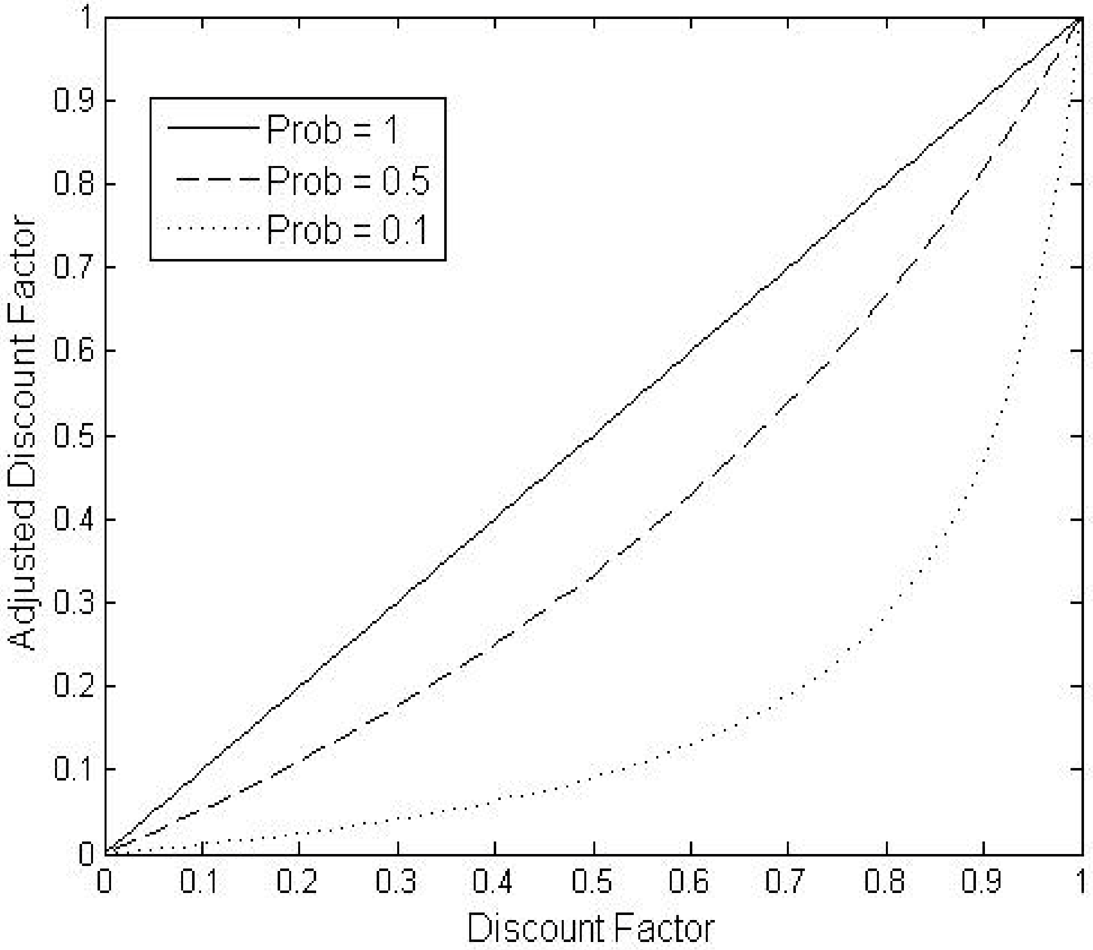

Figure 4.4: Adjusted discount factor vs. Discount factor ($p_i q_i = 1, 0.5, 0.1$)

Most works use simulation to support their model due to the complexity of network interactions. However, we assert that analytical approaches better describe PONT and express more explicit quantitative results. We use a few plots from MATLAB to visualize our results. We analyze first the adjusted discount factor followed by the minimum discount factor able to force cooperation among selfish nodes.

An increase in Δ increases the future loss of a node after an opportunistic deviation and then reinforces nodes cooperation. Fig. 4.4 indicates that the adjusted discount factor Δ increase with the discount factor δ. This is a natural result from the definition of δ.

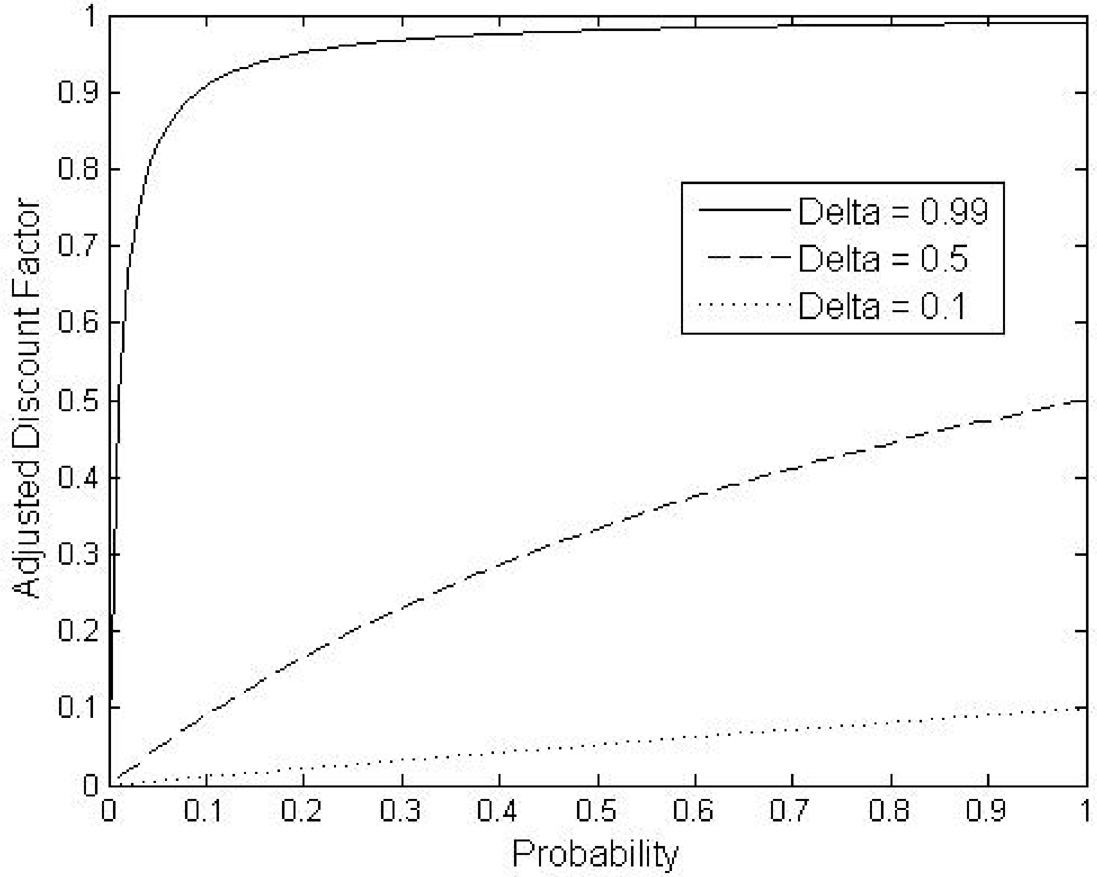

Figure 4.5: Adjusted discount factor vs. Probability $p_i q_i (\delta = 0.99,\ 0.5,\ 0.1)$

Fig. 4.5 show that Δ increases with the product of probabilities $p_i q_i$. This testifies that cooperation can be facilitated when each node sends its own packets at a higher rate. Accordingly, when $p_i q_i = 0, \Delta = 0$, the nodes' cooperation breaks down. In fact, there is no reason for a rational node to cooperate when it does not expect to send its own packets. We also observe in Fig. 4.5 that $\Delta \leq \delta$ with equality if and only if $p_i q_i = 1$. In short, for any value of $p_i q_i$ we have $0 \leq \Delta \leq \delta < 1$.

As explained above, as δ increases, Δ increases and so does the left part of (7). Therefore, if we replace in (7) the superior or equal sign≥() with the equal sign (=), we find the minimum discount factor (δ_{min}) capable of forcing nodes to cooperate.

Fig. 4.6 represents δ_{min}as a function of the cost over gain ratio x_i for the two extreme values of n. n=1 for one period punishment and n=infinite for infinite punishment. Remember that, in this model, we have $0 \leq \delta < 1$. Thus, $\delta_{min} \geq 1$ means that cooperation is impossible. In Fig. 4.6, for $n = 1$, $\delta_{min} \geq 1$ when $x_i \geq 1$. Consequently, for one period of punishment, cooperation is impossible if $x_i \geq 1$ or $\beta_i \geq \gamma_i$. On the contrary, when $\beta_i \leq \gamma_i$, one period punishment should be enough when nodes have a large discount factor.

In general, it can be proven that the number of punishments n should be greater than the cost over gain ratio x_i to allow cooperation. For $n > x_i$, cooperation is possible for a large enough δ. Accordingly, n does not need to be infinite in order for PONT to force cooperation among patient nodes. An infinite n should be avoided to allow the possibility of mutual cooperation in the future after a single defection.

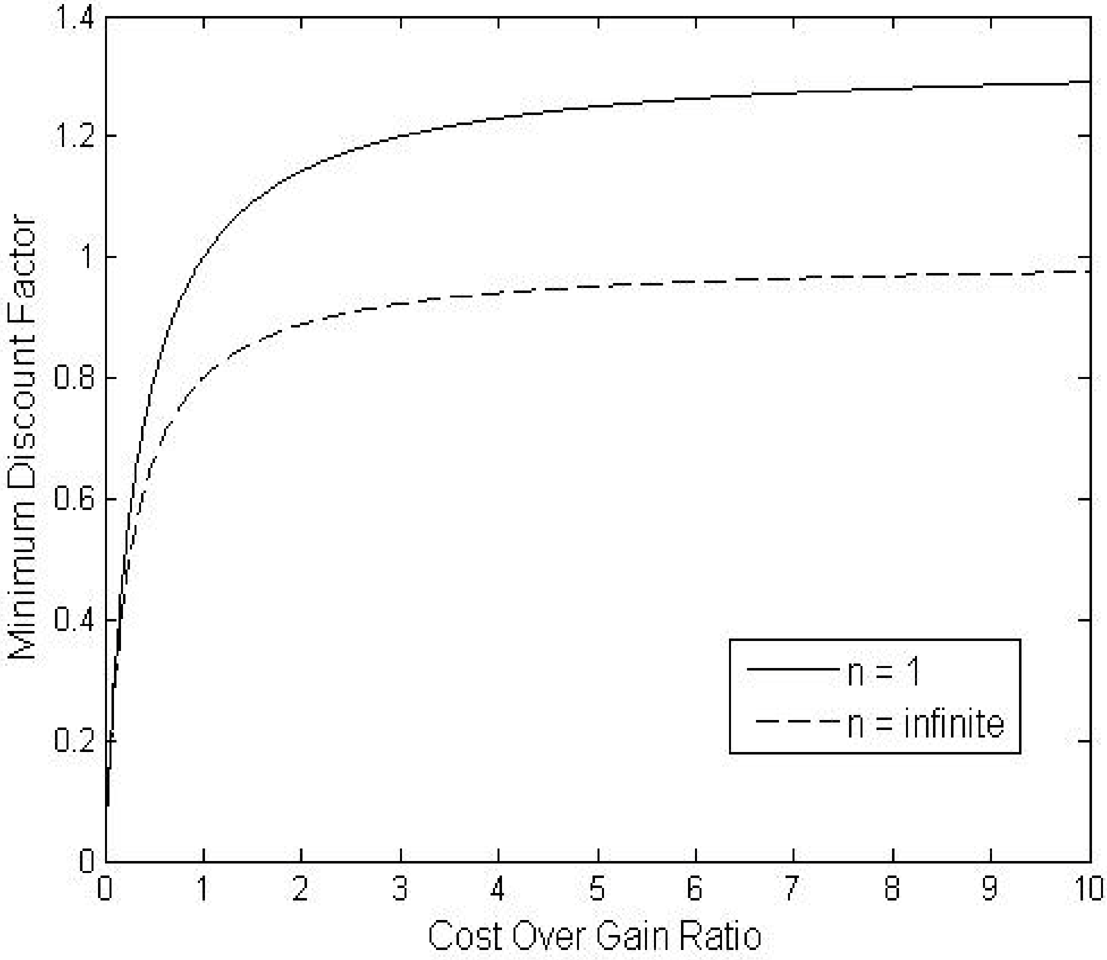

Figure 4.6: Minimum discount factor $\boldsymbol{\delta_{min}}$ vs. Cost over gain ratio $\boldsymbol{x_i}$

$(\boldsymbol{n = 1, n = \infty, p_i = q_i = 0.5})$

Fig. 4.6 affirms that for infinite n, $\delta_{min} < 1$ regardless of how big x_i becomes. The graph is limited to $x_i = 10$ but in fact $\delta_{min} = 1$ is the horizontal asymptote of this graph. Therefore, in the extreme case that n is infinite, cooperation is always possible for a large discount factor (close to 1), regardless of how big x_i is. Note that the value of x_i still needs to satisfy (5). According to (5), in the limiting case we have equality; an increase of x_i implies either an increase of $p_i q_i$ or a decrease of p_j. This means that, for a high value of

x_i, node i cooperates if it sends its own packet at a high rate while being requested to forward costly packets at a lower rate.

Fig. 4.7 clarifies that δ_{min} decreases as the probability $p_i q_i$ that a node sends its own packet in a time slot, increases. In addition, δ_{min} decreases with the number of punishments n. Thus, the number of punishments can be increased when the discount factor is low and cannot motivate cooperation. However, n should be just large enough to motivate cooperation.

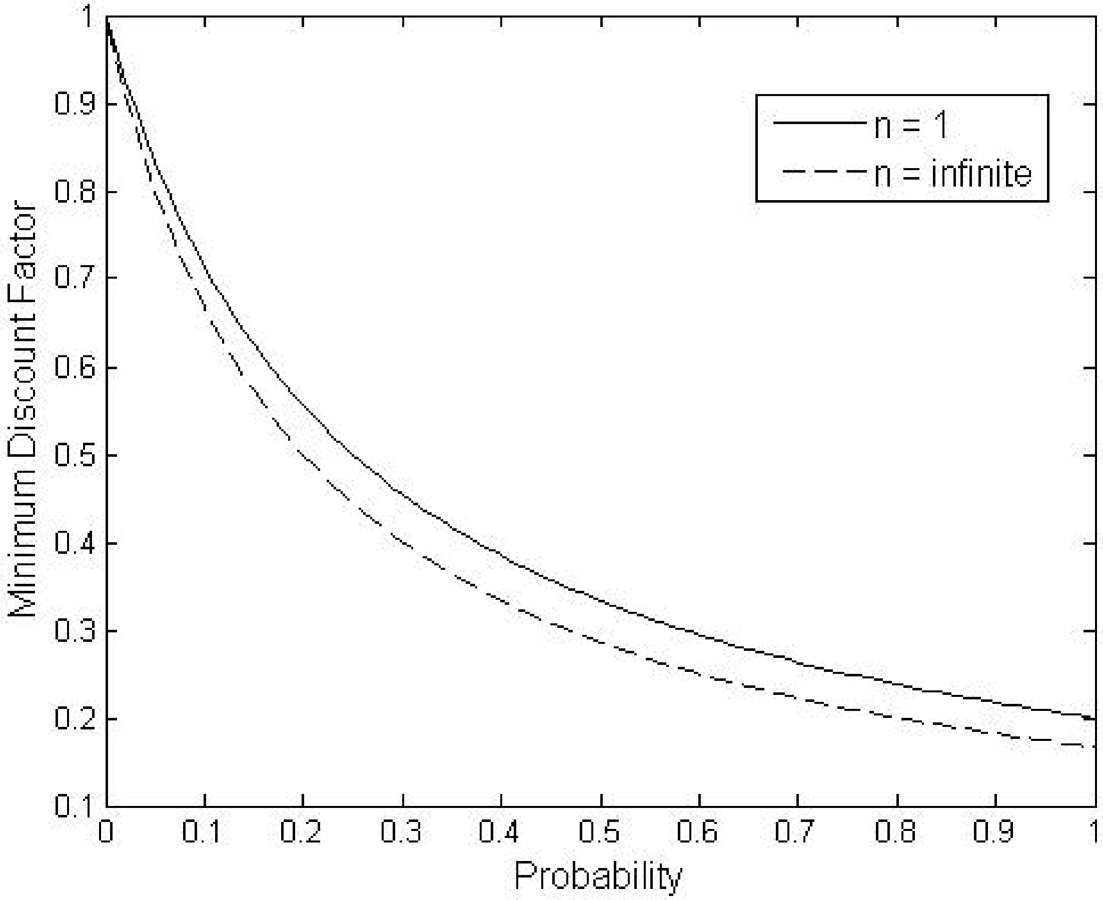

Figure 4.7: Minimum discount factor $\boldsymbol{\delta_{min}}$ vs. Probability $\boldsymbol{p_i q_i}$

$(\boldsymbol{n = 1, n = \infty, x_i = 0.2})$

In summary, the results from Theorem 1 shows that when the nodes in a network use the PONT algorithm, cooperation is always possible among sufficiently patient nodes (discount factor close to 1) for a finite number of punishments (n) higher than the cost over gain ratio (x_i) when each node has packets to send $\left(p_i q_i \neq 1 \text{ and } p_j q_j \neq 1\right)$.

4.5 Conclusion

In this work, we focused on local interaction to design PONT, a distributed algorithm SPE of the packet forwarding game in multi-hop wireless communication. We used the mathematical framework of SGT to prove that cooperation is possible among sufficiently patient nodes if each node has packets to send. We analyzed in detail the impact on cooperation of parameters such as the packet sending rate, the packet forwarding rate, the cost to forward a packet, the payoff to the sender when the packet reaches the destination, and most importantly the discount factor of the nodes.

Our game model is a SPD game instead of an Iterated Prisoners' Dilemma (IPD) game. The intermediate nodes are not rewarded when forwarding a packet but are punished if they drop a packet. In case of defection, the last node to forward the packet has the responsibility to punish, not the originator of the packet. Only the sender (or destination) of a packet is rewarded. We believe this reward mechanism to be realistic. Moreover, our model incorporates the pragmatic case when some nodes do not have a packet to send in a number of time slots or do not have uniform traffic demand.

We based our analysis on two neighboring nodes' interaction to make reasonable predictions for each node in the global network. This way, we avoided a centralized approach to solve the problem of cooperation in distributed networks. In fact, solving the problem of cooperation with equilibrium or the utility function, designed to incorporate full network parameters to be calculated, will be demanding but less convenient and most likely subject to manipulations.

The main advantage of this model is that it is fully distributed. This is consistent with the nature of multi-hop ad hoc networks. Each node is only required to know its neighbor, not the full topology of the network. Each node takes the decision to cooperate or defect only from the information that it can determine or monitor itself. This way, our model is more robust against selfish and malicious nodes that can attempt to manipulate some information for their interests. The full analysis in this model is done at the packet level, not at the route or network level.

In this Chapter we assumed that the interactions between node *i* and node *j* are independent of the interaction between node *i* and node *k*, for instance. In contrast, we can have a scenario in which a selfish node *i* simply refuses to forward messages from node *j* and avoids node *j* by re-routing all of its traffic through node *k.* We will investigate this scenario in the future. In the next Chapter, we will consider imperfect monitoring when there is noise in observing the neighbor's actions. In the future, we will implement PONT on an adequate routing protocol in a real multi-hop network to analyze the impact of network flow, network topology, and node mobility.

CHAPTER 5

BELIEF-FREE EQUILIBRIUM OF PACKET FORWARDING GAME IN AD HOC NETWORKS UNDER IMPERFECT MONITORING

Future applications will require autonomous devices to be interconnected and form ad hoc networks. In such networks, cooperation will be the first problem to solve at all layers of the protocol stack. This work deals with one of the basic functions of a network, namely packet forwarding. We model packet forwarding as a stochastic game in which each node monitors the behavior of its neighbors. We consider the realistic scenario when the monitoring technology used by the nodes is imperfect. In reality, there can be inconsistencies between the true action of a node and the observations of its neighbors. Therefore, in an ad hoc network, each node receives only noisy private information about the past play of its neighbors. We develop a simple one period memory strategy that constrains self-interested nodes to cooperate under noise. We use a belief-free approach. A belief-free approach delivers a tremendous computational advantage because nodes' belief about the neighbors' private history does not need to be computed. We support our results by mathematical proofs and simulations.

5.1 Introduction

Ad hoc networks are networks without infrastructure. Each node is at the same time a terminal and a router. As a terminal, each node sends and receives its packets. As a router, each node performs different functions such as route discovery, route maintenance and, packet forwarding. In many applications, nodes have the same manager and fully cooperate to achieve the common goal of the network.

However, future applications will require autonomous devices to be interconnected. Each node will be its own manager and will act of its own self-interest. Selfish nodes will be tempted to drop packets from other nodes to save critical resources such as battery power and bandwidth. However, the network collapses if all nodes refuse to relay other nodes' packets; no node benefits from such an outcome. This situation is similar to the Prisoners' Dilemma Game found in game theory. Thus, the PDG will be the basis of our model.

In recent years, researchers have used game theory to model the packet forwarding game in ad hoc networks [1-3, 6-11, 13-17]. The basic idea behind the packet forwarding game theoretic model is to use repeated interaction among nodes to provide the necessary incentive to cooperate. Future packets originating from defectors are dropped to punish them. If for each node, at all times the future lose is greater than the current gain from defection, the network reaches a Nash equilibrium where all selfish nodes always cooperate. In a cooperative Nash equilibrium no node can profit by dropping packet unilaterally. Moreover, forwarding costly packets is the best response to the behavior of other nodes.

However, the implicit assumption of perfect monitoring is present in most papers [1-2, 6-11]. The packet forwarding game equilibrium constructed in those papers are not robust when considering noise. A game is of perfect monitoring if the history of chosen actions is common knowledge among the players. Perfect monitoring is a strong assumption in a distributed network for three primary reasons. First, no node can directly monitor the behavior of nodes outside its communication range. Second, generally, in game theoretical models, each node is equipped with a watchdog mechanism [32] to monitor the actions of other nodes in its communication range and perform the best response based on the observed actions. A node's watchdog mechanism cannot monitor those actions at the same time that the node sends or receives packets. Third, even though a node decides to cooperate, a packet can be dropped because of transmission errors and link breakages. Surely, there will be congestion, interference, collisions, and noise that create inconsistency between observed actions and the true actions in a distributed network.

Srivastava and DaSilva [13] relax the assumption of perfect monitoring. They model the network packet forwarding game as a game of imperfect public monitoring. In such a game, past actions of nodes are imprecise and noisy but it is assumed that nodes commonly observe a public signal about the actions of others. However, the availability of such signals in a distributed network is not always guaranteed.

Ji *et al*. [14-15] use a game of imperfect private monitoring to model noise in the network packet forwarding game. A public signal is not needed. Nodes do not have common knowledge about the history of the game but each node has a private history of the game. Each node needs to infer the private history of other nodes based on their own imperfect observations. Those inferences become complicated in the long term and require more computational power than our model. This is called a *belief-based* approach because the equilibrium strategy depends on the opponents' private history.

A sequential equilibrium [60] is *belief-free* if, after every history, each player's continuation strategy is optimal independently of the opponents' history [60-67]. We use a *belief-free equilibrium* approach because it is easily tractable. The need to infer other nodes private history is eliminated and the computation of optimal strategy is simplified.

The work in [16] is closely related to this Chapter. However, the equilibrium constructed in [16] is more complex than ours. A node using our strategy starts by cooperating and cooperates for sure after observing (with the possibility of erroneous observation) that the opponent cooperated in the last stage. The node cooperates with a probability less than one (to be carefully calculated in Section 5.2) if it observed the defection of the opponent in the last stage. As, you can see, our strategy depends on a single parameter, the probability to cooperate after observing a deviation from the opponent. Also, contrary to [16], our strategy does not depend on a node's own action. Only the last action of the opponent is needed. Therefore, our model easily translates from a two nodes game to an *n* nodes game using new research from [67]. Last but not least, our model introduces a stochastic dimension to the game to synthesize the randomness of network interaction, the traffic load inequality in the network and the border effect. Our game model is a stochastic PDG. As such, this Chapter extends the work in the last Chapter [2] while considering noise. Our simulation results indicate that our simple strategy enforces cooperation among autonomous nodes and is robust to noise with only a small performance degradation compared to a network with a central manager.

Other approaches that address the problem of cooperation in ad hoc networks include EGT [4-5], reputation mechanisms [31-34] and virtual currency systems [28-29]. EGT relaxes the assumption of rationality used in game theory but relies on the statistical distribution of strategies used in the network. Moreover, nodes do not choose strategy in EGT models. Nodes are programmed to use certain strategies in the game. Reputation mechanisms require each node to monitor its neighbor and compute a reputation for each neighbor. Nodes with a low reputation are either avoided [32] or isolated [31, 33-34]. Virtual currency systems give a virtual payment to intermediate nodes to relay packets. The payment collected by each node can be used to send its own packets. The remainder of this Chapter is organized as follows: Section 5.2 presents our repeated game model with two nodes. The stage game model is similar to the one presented in Chapter 4 in Section 4.2. Section 5.3 proposes an extension of the two node model in Section 4.2 for *n* nodes. Section 5.4 presents our simulation results, and Section 5.5 concludes the Chapter.

5.2 Repeated Game under Imperfect Private Monitoring

We analyzed the stochastic game in Table 4.2 under perfect monitoring in Chapter 4 and in [2]. Recall that after payoff normalization, the gain from deviation

$$g_i = \frac{p_j \beta_i}{p_i q_i \gamma_i - p_j \beta_i} > 0. \tag{1}$$

We now consider the impact of noise caused by watchdog imperfection. We can have inconsistencies between the observed action of node and the true action. For instance, in Fig. 4.1, node *j* can relay a packet from node *i* at the same time that node *i* is receiving a packet from node 3. Therefore, node *i* is not able to observe that node *j* cooperated. Node *i* will observe a defection from node *j* even though node *j* cooperated. Moreover, although a node decides to cooperate, a packet can be dropped because of transmission error and link breakage. We now extend that model to an imperfect private monitoring model.

One of the most important results in repeated games is the folk theorem. That is, self interested agents can achieve any individually rational and feasible payoff if they are engaged in a long term relationship with repeated and frequent interactions. The folk theorem has been proven under perfect monitoring and imperfect public monitoring but the folk theorem for imperfect private monitoring is still an active research area [60-68]. The main difficulty is that players are not able to coordinate to punish a deviation from the equilibrium strategy because players have different observations of other players' actions. The history of the game is private.

Bhaskar and Obara [68] prove that the symmetric efficient outcome can be approximated in any PDG. Their approach is belief-based. That requires each player to infer the opponents' private history. Their results are applied to ad hoc networks in [14-15]. Our model uses the current advances in research presented in [60-67]. Ely and Valimaki [61] use a belief-free approach to prove the folk theorem in the two players PDG. Ely *et al.* [62] provide a characterization of the set of belief-free equilibrium payoffs in all two-person games for any discount factor and any accuracy of the monitoring technology. Yamamoto proves that the payoff of mutual cooperation can be achieved in the *n* player PDG [65] and later proves the folk theorem in *n* person PDG in the limit as the noise in monitoring vanishes and the discount factor is close to one [66]. Takahashi [67] uses a belief-free equilibrium approach in the context of *community enforcement*. A simple presentation of belief-free equilibrium is found in [64]. An interested reader is recommended to read [60] and [63] for a detailed presentation on repeated games under different forms of monitoring.

A. Model

In this Section, we analyze the two node game. The players are nodes i, (*i=1, 2*). In fact, each node is involved in a two node game with each of its neighbors. Each node chooses action a_i from the set A_i={C, D}. C designates cooperation, and D designates defection. A profile of actions is a vector $a \in A = A_i \times A_j$. Also, each node receives a signal y_i from the set $Y_i = \{c_i, d_i\}$. c_i and d_i are the observation of cooperation and defection by node *i* respectively.

A monitoring technology is a collection of probability distributions $\{m(.|a): a \in A\}$ over $Y = Y_i \times Y_j$. This means that each node receives a signal $y_i \in Y_i$, and that each signal profile (y_i, y_j) is obtained with probability $m(y_i, y_j|a)$. The marginal distribution over node *i*'s signal is denoted $m_i(.|a)$. Node *i*'s realized payoff $r_i(a_i, y_i)$ depends on the action $a_i$and the private signal y_i. Thus, the expected payoff of node *i* from action profile a is:

$$u_i(a) = \sum_{(y_i, y_j) \in Y} m(y_i, y_j|a)\, r_i(a_i, y_i). \qquad (2)$$

Node *i*'s private history in the repeated game is $h_i^t = (a_i^1, y_i^1, \dots, a_i^{t-1}, y_i^{t-1})$. A profile of history is a vector $h^t = (h_i^t, h_j^t)$. Node *i*'s sequence of stage game payoffs is $(u_i^t)_{t=1}^{\infty}$. We consider that all nodes discount future payoffs by a common discount factor δ with

$0 < \delta < 1$ and want to maximize the expected δ-discounted average of their sequence of payoffs. Therefore, the repeated game payoff is:

$$(1-\delta)\sum_{t=1}^{\infty} \delta^{t-1} u_i^t. \tag{3}$$

We presuppose that the monitoring technology or watchdog mechanism in each node is ε-perfect. A monitoring technology is ε-perfect if $m_i(c_i|a_i, C_j) > 1-\varepsilon$ and $m_i(c_i|a_i, D_j) < \varepsilon$ with $\varepsilon \geq 0$. $m_i(c_i|a_i, C_j)$ and $m_i(c_i|a_i, D_j)$ represent the probability that node i observes a cooperative behavior from node j given that node j cooperated or defected respectively.

This definition gives us the possibility to study the packet forwarding game under several forms of monitoring structures.

- A game will be of perfect monitoring if $\varepsilon = 0$.
- A game will be of imperfect public monitoring if nodes' signals are perfectly correlated.
- Monitoring is conditionally independent if $m_i(.|a)$ and $m_j(.|a)$ are independent distributions for each player's action profile a.

We consider that in a network, one node's observation of an action profile is conditionally independent of another node's observation of that same profile. Also, the probability distribution over the signal received by a node depends only on its opponent's action. Moreover, each node's history of the game is private. Therefore, we will use a conditionally independent ε-perfect private monitoring. Then, we have:

$$m_i(y_i|a) = \begin{cases} 1-\varepsilon, & \text{if } y_i = c_i \text{ and } a_j = C_j, \text{or} \\ & y_i = d_i \text{ and } a_j = D_j, \\ \varepsilon, & \text{otherwise,} \end{cases} \tag{4}$$

And

$$m_j(y_j|a) = \begin{cases} 1-\varepsilon, & \text{if } y_j = c_j \text{ and } a_i = C_i, \text{or} \\ & y_j = d_j \text{ and } a_i = D_i, \\ \varepsilon, & \text{otherwise,} \end{cases} \tag{5}$$

The joint probability distribution is given by:

$$m(y|a) = m_i(y_i|a)m_j(y_j|a). \tag{6}$$

B. Strategy Description

A strategy σ_i for node *i* is a sequence of functions $(\sigma_i^t)_{t=1}^{\infty}$ where σ_i^t maps each $h_i^t$to a probability distribution over A_i. We consider the strategy σ with one period memory:

$$\begin{cases} \sigma_i^1 = C_i \text{ with probability } 1, \\ \sigma_i^t = C_i \text{ with probability 1 if } y_i^{t-1} = c_i, \\ \sigma_i^t = C_i \text{ with probability } \omega_i \text{ if } y_i^{t-1} = d_{i.} \end{cases} \tag{7}$$

With

$$\omega_i = 1 - \frac{g_j}{\delta(1-2\varepsilon)(1+g_j)}. \tag{8}$$

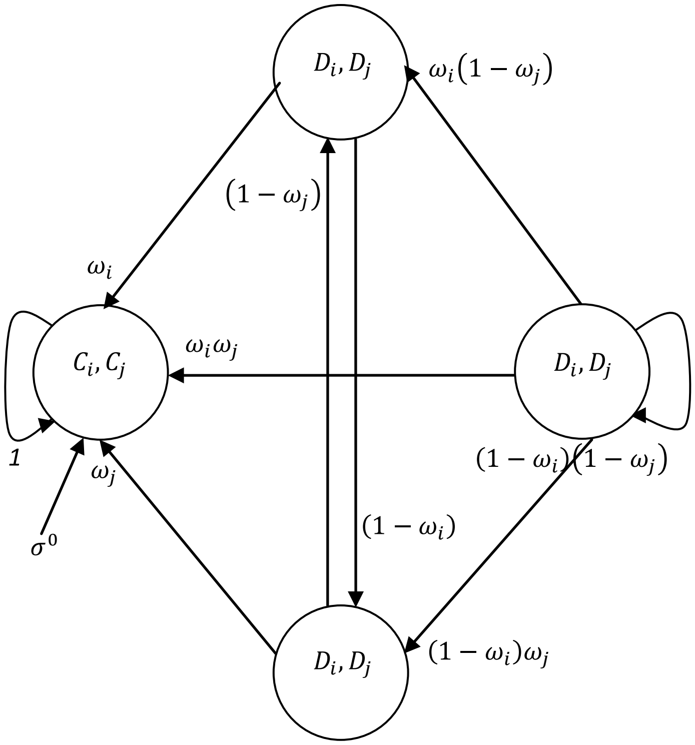

Figure 5.1: Finite State Machine of the strategy σ when noise in monitoring vanishes

The finite state machine of σ is represented in Fig. 5.1 for the two node game when noise in the monitoring technology vanishes ($\varepsilon = 0$). The states are represented in circles. C and D represent the action of each player in a given state. The state (C_i, C_j) is an absorbing state. When noise in the monitoring technology vanishes, the players start in the state (C_i, C_j) and remain there forever. In the presence of noise ($\varepsilon \neq 0$), nodes can move to other states. However, there is always a possibility to come back to the state (C_i, C_j). If an opponent deviates from σ, σ punishes the opponent by diminishing the probability of cooperation from 1 to ω in the next time slot. This diminution is enough to force selfish nodes to cooperate if the discount factor is close enough to 1. We analyze the equilibrium of σ and show how to calculate ω in the next Subsection.

C. Efficiency Analysis

We start with the general case as presented in [61] before going on to obtain our specific strategy. In the following equation, $\pi_{a_j a_i}^{j}$ represents the probability that node *j* cooperates given that in the last time slot node *j* played the action a_j and observed a_i from node *i*. We want to construct the four probabilities $\pi_{a_j a_i}^{j}$such that the history of play becomes irrelevant and therefore calculating posterior belief over the private history of the opponent is not necessary. This can be represented by dynamic programming equations in (9) through (12). Clearly, this means that the four probabilities $\{\pi_{a_j a_i}^{j}: a_j a_i \in \{C, D\}^2\}$ must be such that:

- If node *j* plays C in a time slot, node *i* obtains the average payoff V_C^i and is indifferent between playing C and D. This is represented by (9) and (10).
- If node *j* plays D in a time slot, node *i* obtains the average payoff V_D^i and is indifferent between playing C and D. This is represented by (11) and (12).

$$V_C^i = (1-\delta) + \delta V_C^i\left[(1-\varepsilon)\pi_{cc}^{j} + \varepsilon\pi_{cd}^{j}\right] + \delta V_D^i\left[\varepsilon\left(1-\pi_{cd}^{j}\right) + (1-\varepsilon)\left(1-\pi_{cc}^{j}\right)\right]. \quad (9)$$

$$V_C^i = (1-\delta)(1+g_i) + \delta V_C^i\left[\varepsilon\pi_{cc}^{j} + (1-\varepsilon)\pi_{cd}^{j}\right]$$

$$+\delta V_D^i\left[(1-\varepsilon)\left(1-\pi_{cd}^{j}\right) + \varepsilon\left(1-\pi_{cc}^{j}\right)\right]. \quad (10)$$

$$V_D^i = -g_i(1-\delta) + \delta V_C^i\left[(1-\varepsilon)\pi_{dc}^j + \varepsilon\pi_{dd}^j\right]$$

$$+\delta V_D^i\left[\varepsilon\left(1-\pi_{cd}^j\right) + (1-\varepsilon)\left(1-\pi_{dc}^j\right)\right]. \quad (11)$$

$$V_D^i = \delta V_C^i\left[\varepsilon\pi_{dc}^j + (1-\varepsilon)\pi_{dd}^j\right] + \delta V_D^i\left[(1-\varepsilon)\left(1-\pi_{dd}^j\right) + \varepsilon\left(1-\pi_{dc}^j\right)\right]. \quad (12)$$

Expressing the probabilities as a function of the two payoffs and the monitoring error, we get:

$$\pi_{cc}^j = \frac{\left(V_C^i - \delta V_D^i\right)(1-2\varepsilon) + (1-\delta)\left(g_i + \varepsilon - (1-\varepsilon)(1+g_i)\right)}{\delta(1-2\varepsilon)\left(V_C^i - V_D^i\right)}. \quad (13)$$

$$\pi_{cd}^j = \frac{\left(V_C^i - \delta V_D^i\right)(1-2\varepsilon) + (1-\delta)\left(\varepsilon - (1-\varepsilon)(1+g_i)\right)}{\delta(1-2\varepsilon)\left(V_C^i - V_D^i\right)}. \quad (14)$$

$$\pi_{dc}^j = \frac{(1-\delta)\left(g_i(1-\varepsilon) + V_D^i(1-2\varepsilon)\right)}{\delta(1-2\varepsilon)\left(V_C^i - V_D^i\right)}. \quad (15)$$

$$\pi_{dd}^j = \frac{(1-\delta)\left(V_D^i(1-2\varepsilon) - g_i\varepsilon\right)}{\delta(1-2\varepsilon)\left(V_C^i - V_D^i\right)}. \quad (16)$$

We can see that (13) through (16) have two-dimensional indeterminacy or two free variables. Thus, using the first free variable, we can set:

$$V_C^i - V_D^i = (1-\delta)(1+g_i), \quad (17)$$

And it is easy to verify that taking (17) into (13) through (16), we get:

$$\pi_{cc}^j = \pi_{dc}^j = \frac{(1-2\varepsilon)V_C^i - (1-2\varepsilon)(1-\delta)(1+g_i) + (1-\varepsilon)g_i}{\delta(1-2\varepsilon)(1+g_i)}, \quad (18)$$

and

$$\pi_{cd}^{j} = \pi_{dd}^{j} = \frac{(1-2\varepsilon)V_C^i - (1-2\varepsilon)(1-\delta)(1+g_i) - \varepsilon g_i}{\delta(1-2\varepsilon)(1+g_i)}. \tag{19}$$

Moreover, subtracting (18) and (19) give:

$$\pi_{cc}^{j} - \pi_{dd}^{j} = \frac{g_i}{\delta(1-2\varepsilon)(1+g_i)}. \tag{20}$$

Therefore, using the second free variable, we can also set:

$$\begin{cases} \pi_{cc}^{j} = \pi_{dc}^{j} = 1, \qquad \text{and} \\ \pi_{cd}^{j} = \pi_{dd}^{j} = \omega_j = 1 - \dfrac{g_i}{\delta(1-2\varepsilon)(1+g_i)}. \end{cases} \tag{21}$$

We can verify that when the discount factor is close to 1 ($\delta \to 1$) and noise in the monitoring vanishes ($\varepsilon \to 0$), we have:

$$0 < \omega_j = \pi_{cd}^{j} = \pi_{dd}^{j} < \pi_{cc}^{j} = \pi_{dc}^{j} = 1. \tag{22}$$

Equation (22) indicates that our strategy is well defined. In fact, we must have the four probabilities $\{\pi_{a_j a_i}^{j}: a_j a_i \in \{C, D\}^2\}$ in the interval [0, 1]. The same way, we can have a similar strategy for node i. However, since the game is asymmetric, we will have $\omega_i \neq \omega_j$. Node *i*'s strategy is a best reply to node *j*'s strategy. The strategy profile (π^i, π^j) will be a sequential equilibrium and thus a belief-free equilibrium by construction since player's belief about the opponent's private history is irrelevant. Also, from (21), we must have for the two nodes

$$\varepsilon < \frac{1}{2} - \frac{g}{2\delta(1+g)} < \frac{1}{2}. \tag{23}$$

If $\varepsilon > \frac{1}{2}$, the noise in the monitoring technology is so pervasive that a node's opponent is more likely to observe a defection when the node cooperates. In this case, intelligent nodes prefer to always defect. The incentive to cooperate is lost due to noise.

Moreover, the fact that $\pi_{cd}^{j} = \pi_{dd}^{j} < \pi_{cc}^{j} = \pi_{dc}^{j}$ provides the incentive to cooperate. A deviation by a node from cooperation to defection increases the chance that the opponent will defect in the future and thus diminishes its continuation value from V_C^i to V_D^i. The difference between those continuation values balances the gain from deviation in any time slot (17).

Finally, after all calculations, we can see that:

$$V_C^i = 1 - \frac{\varepsilon g_i}{1 - 2\varepsilon}. \tag{24}$$

Therefore, as noise in the monitoring technology vanishes, both nodes get the payoff of full cooperation. This shows the efficiency of our belief-free equilibrium strategy. In summary, the belief-free equilibrium approach has several advantages [66].

- A belief-free equilibrium can be used with any type of monitoring technology, from perfect monitoring to the extreme case where the monitoring technology is fully noisy and fully private [62].

- A belief-free equilibrium strategy can have one period memory. A memory one strategy can be represented by a finite state machine, an automaton or a Markov process.

- Node belief about the opponent's private history is irrelevant to its best reply. This represents a tremendous advantage over a *belief-based equilibrium*. The computational power needed to compute the equilibrium is reduced.

- Continuing to play a belief-free equilibrium is sequentially rational even if a node receives additional information about the opponent's history.

- A belief-free equilibrium can be used with a community enforcement mechanism as in [67]. We will use a model similar to [67] when extending our model to *n* nodes in the next Section.

5.3 Extension to *N* Nodes

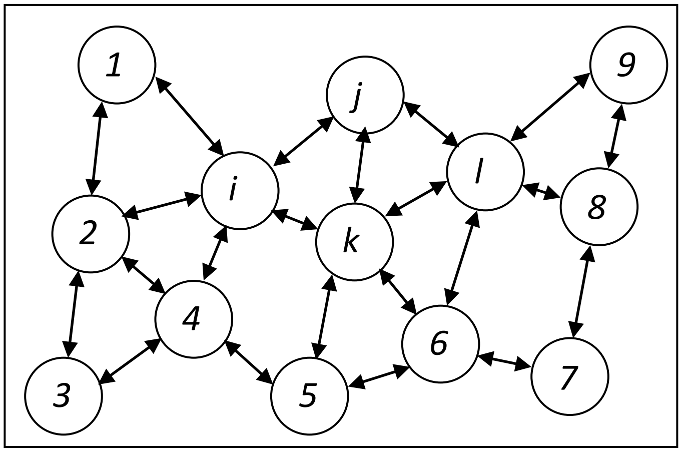

Figure 5.2: Randomly deployed network in a rectangle

Figure 5.2 represents 13 nodes randomly deployed in a rectangular area. As we did in Section 4.2 with Fig.4.1, we will use the network in Fig. 5.2 to clearly expose some complications in modeling the full network packet forwarding game before presenting some solutions.

First of all, the packet forwarding game in a network is not an isolated game. It is part of the routing game at the routing layer which also depends on the nodes' interactions at the MAC layer and the transport layer. At the MAC layer, there can be congestion that causes nodes to drop packets. A protocol such as TCP at the transport layer recommends packet retransmission if a packet is lost. Retransmitting packets will influence the packet forwarding game because of the extra cost involved. At the routing layer, routing packets in MANETs generally involves four steps (not common to all routing protocols):

- Route discovery: when a sender wants to communicate with a destination outside its communication range, it broadcasts a route request message that is relayed until the destination. The destination sends back a route reply for each route request received.

- Route selection: the sender selects one route among the routes available to it.

- Packet forwarding: the sender sends the packets that are relayed along the selected route until the destination. This is our main focus.

- Route maintenance: when a node observes a link breakage, for instance due to node mobility, it reports that to other nodes using a route error message.

Each of these steps involves strategic interactions with severe conflict of interest. Each step can be modeled by a game. For instance, in the route selection game, when node *1* wants to send packets to node *9*, node *1* may have to choose between several routes. One possible route goes through nodes $i, j,$ and $l,$ and the second possible route goes through node *2, 4, 5, 6, 7*, and *8*. Considering its best interest, node *1* will most likely choose the first route because of shorter delay whereas nodes $i, j,$ and l will prefer the second route to be used to avoid a higher cost in the packet forwarding game. Moreover, node *1*'s decision to use the first route instantly increases the probabilities p_i, q_i, p_j, q_j in the interaction between node i and j. This is one of the reasons indicating that a stochastic game better models packet forwarding than a repeated game.

In this Chapter, we focus only on the packet forwarding game and do not consider the specifics of any routing protocol or the interactions from other layers. The packet forwarding game analysis manifests several difficulties. As we stated before, the first complication is that there is less incentive to cooperate with border nodes, such as nodes *1, 3,* and *9* if the network is static. Those nodes are useless to their neighbors in the packet forwarding game. Therefore, border nodes can easily be disconnected in a static network if we rely only on the two player game.

The second problem concerns node mobility that creates dynamic changes of neighbors. The third problem concerns the monitoring technology in distributed networks. In general, a node has no private signal at all about the actions of nodes outside its communication range. As a consequence, monitoring cannot be ε-perfect in *n* node interactions. For instance, in Fig.5.2, the action of node 3 is completely unknown to node

9. ε-perfect monitoring in the *n* node game will imply that node 9 can monitor the action of node 3 with the probability of observing an erroneous signal being less than ε. This is not possible without an external system or database recording each node's action that is accessible to all nodes.

A. Model Extension Using a Database

The existence of an appropriate database where each node submits a report indicating other nodes' actions as the one described in [29] facilitate the game analysis and can solve the three problems presented here. In fact, the model described in [29] is cheat-proof, selfish nodes cannot manipulate the data to their advantage. The database is not a central manager; the database does not impose the action to be taken by a node. Each node remains free to decide to cooperate or not. Using a belief-free equilibrium approach, the database will not store the reputation or the full history of the game but only the last action of each node. Recall that node belief about the opponent's private history is irrelevant. In this case, there are three papers [65-67] that provide a formal proof that a belief-free equilibrium strategy can enforce cooperation in *n* player PDGs under imperfect private monitoring. The payoff of mutual cooperation is achieved in the limit as noise in monitoring and discounting vanishes.

We use the approach presented in [67] which is a relatively simple approach that can easily translate our two node game of Section 5.2 to an *n* node game. The research from that paper allows us to treat the *n* person PDG under imperfect private monitoring when each node observes only the last action of a single node at the beginning of each stage. Such mathematical development has a tremendous importance for distributed network packet forwarding game modeling. Only the noisy observation of the opponent's last action is relevant, not the player's own action. This is consistent with the belief-free equilibrium strategy we developed for the two node game in Section 5.2. In fact, *independent and indifferent equilibria* [67] must satisfy the following two properties:

- Players are indifferent between cooperation and defection at all histories.
- Players choose actions independently of their own record of play.

Therefore, our strategy in Section 5.2 is not only a belief-free equilibrium but also an independent indifferent equilibrium.

A mechanism that requires defection to be punished not only by the victim but by any other node that observes the defection is called community enforcement. We consider a large MANET of *2n* nodes. The network topology is dynamically changing and nodes interact with different nodes over time. At the beginning of each time slot, neighboring nodes are randomly matched in pairs to play the PDG in Table 4.3. Thus, in each time slot, there are *n* pairs of nodes playing the PDG. Without loss of generality, we consider symmetric payoff. Moreover, the matching is uniform and independent across time. Note that in our game, the incentive a node has to deviate when its opponent cooperates (*g*) equals the loss from cooperating when the opponent deviates. Thus, our game is neither strictly submodular nor strictly supermodular.

Nodes observe past action through direct observation or the database. This means that if a node did not directly monitor its current opponent's last action, possibly because it was outside its communication range, it can freely observe that from the database. There can be noise when observing the node's last actions in the database or when observing an opponent's action directly.

A database gives the incentive to each node to always cooperate with all nodes regardless of node position, node mobility or the network topology. A defection in a time slot increases the probability that the next opponent from the next time slot will defect in such a way that a node is always indifferent between cooperation and defection. The only difference with the two node game is that in the two node game, each node has a single opponent and that single opponent will increase its probability of future defection if observing a defection. However, in the *n* node game, any node that defects has its action recorded in the database. In the next time slot, its opponent (which may be a different node) observes that defection from the database and punishes by increasing the probability of defection.

Another excellent characteristic of our strategy is that the long-run equilibrium is robust to one time shock. In fact, if a small percentage of nodes mistakenly defect in a time slot, that defection increases the probability the future partners will defect and so on. However after all computations [67], the fraction of cooperation in the network remains constant over time.

B. Model Extension without a Database

When a database is not available in the network, enforcing cooperation in the network is more complex. Full cooperation should be achieved if each node cooperates with all of its neighbors. In Section 5.2, we have presented a simple belief-free equilibrium strategy for the two node game. Each node in Fig. 5.2 is involved in a two node game with each of its neighbors. The total number of such games in an ad hoc network will be equal to the number of links or wireless connections between nodes. However, a two node interaction is not independent of the other interactions in the network. Those games are correlated and the correlations between those games are complex to analyze. These correlations are neglected in other papers.

To illustrate the connections between the two node games, consider that mutual cooperation is Pareto efficient between any two neighbors in Fig. 5.2. Thus, as we showed previously, a sequential equilibrium can enforce cooperation in the two player game between node i and node j. However, node i has the option to end cooperation with node j, and to avoid punishment, reroutes all its packets through node k. As a result, the rate at which node i requests node k to relay packets increases (p_i increases) and so does the cost to node k to continue to cooperate with node j. Then, node k may decide to stop cooperation with node i if cooperation is no longer Pareto efficient ($g_k < 0$). As a consequence, the traffic at node 5 increases dramatically. Node 5 may also stop cooperating with node 4 resulting in a network disconnection, which has a worse outcome for node i compared to the initial one. Therefore, rational nodes in distributed networks can behave optimally from local and incomplete information, but the final outcome becomes damaging for them. A very small change of decision from a single node can destabilize the network cooperative equilibrium.

One may argue that the above scenario is impossible using a backward induction argument. That is, node i may have anticipated this sequence of action and not stopped cooperating with node *j* in the first place. Node *i* is worse off after network disconnection than before since it can now transmit its messages to only a small portion of the network. However, a backward induction argument does not work here because in distributed self organized MANETs without a database, the network topology is not common knowledge among the nodes. Each node knows only its neighbors but not their neighbors' neighbors. A rational node cannot predict the effect that its actions would have on the actions of other rational nodes.

Nevertheless, a belief-free equilibrium like the one we present in this Chapter is more robust against such a dramatic scenario compared to a belief based approach because the history of the game is irrelevant. In fact, when two nodes reach an outcome that is worse for the two, they can easily start to cooperate again and increase the number of available routes in the network. Using a belief-based approach [14-15], a node that defects successively for a few time slots causes its neighbors to correctly infer its private history and switch to defection forever. Our strategy can forgive after any number of defections.

Lastly, we can improve the network performance by recommending that, in the punishment phase (cooperation with probability less than 1), a node drops only the next packet originating from the punished neighbor. This way, packets from other nodes in the network are not affected while at the same time each node has the incentive to cooperate since it fears having some of its packets dropped in the future. However, this will extend the memory length of our strategy and increase its complexity.

5.4 Simulation Results

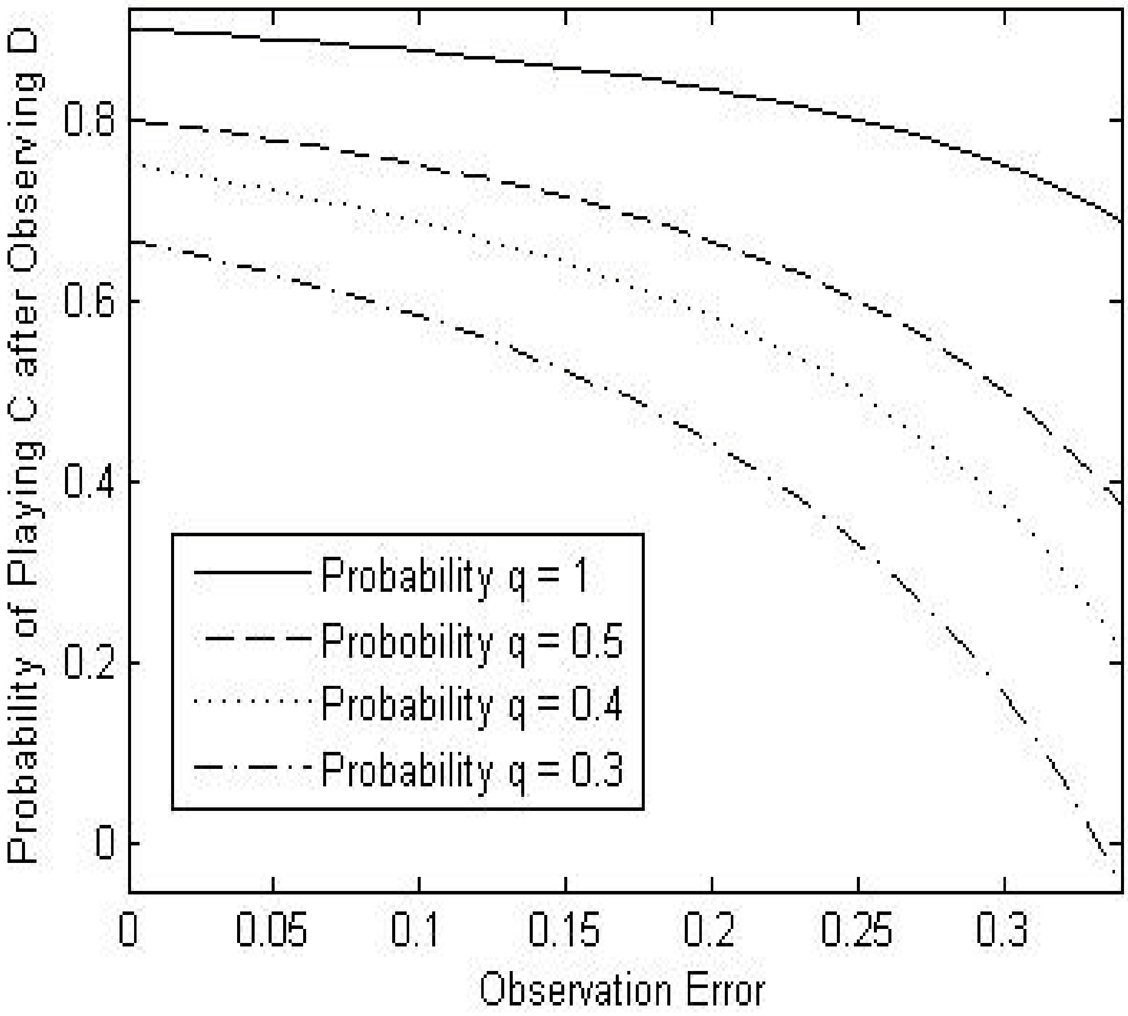

Figure 5.3: Variation of ω with ε and q

In this Section, we analyze by simulation the impact of noise on our belief-free equilibrium strategy. We also consider the traffic load inequality by considering different values for the probability $q_i$defined before. In our simulation, we set $\gamma = 10\beta$. This implies that it is globally efficient to relay a packet up to a maximum of 10 hops. Also, we set $p_i = p_j$. Thus, from (1), we have $g_i = \frac{1}{10q_i - 1}$. Also, $\delta = 0.999$ is the default value.

Figure 5.3 shows the variation of the probability with which a node cooperates after observing the opponent's defection (ω) as a function of the observation error (ε) and the probability q (21). We can see that ω decreases with ε and also decreases with q. There is a maximum value of ε above which $\omega < 0$.

When the observation error increases, nodes are more tempted to defect because the probability that their opponent will not observe that defection increases. To balance that temptation, a node should punish more severely after observing a defection by decreasing its probability of cooperation. The same way, a decrease in q increases g (1) which makes defection more profitable and cooperation among nodes more difficult. Therefore, we can say that a decrease of ω when q decreases is required to provide the necessary incentive for nodes to cooperate.

Figures 5.4 and 5.5 represent the variation of the probability of node cooperation as a function of time in the interaction between two nodes. That probability can be obtained analytically by a Markov analysis since our strategy obeys the Markov property.

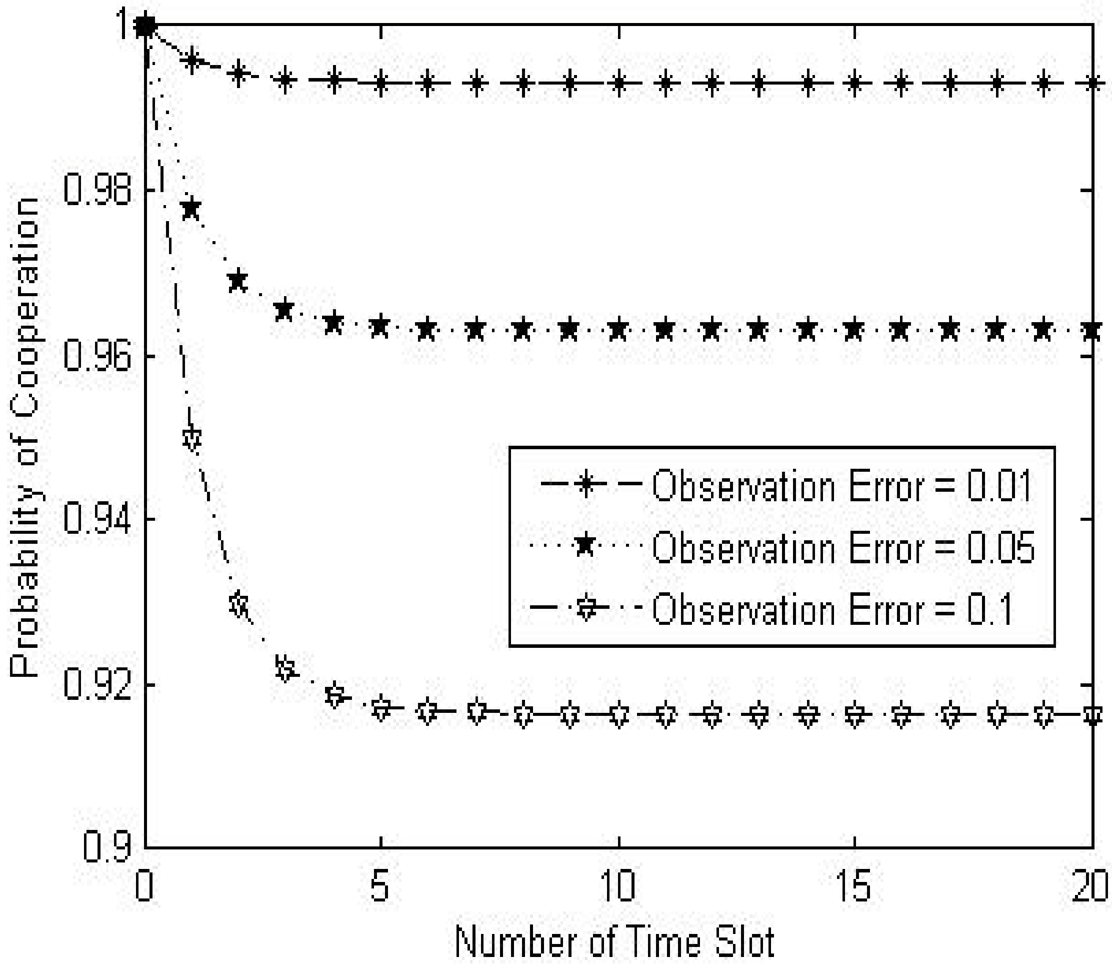

Figure 5.4: Variation of the probability of cooperation with time and ε.

In Fig. 5.4, we fix $q = 0.25$ and focus on the impact of noise. $q = 0.25$ will be a node for which 3 out of 4 packets it requests to forward come from other nodes. For different values of noise, the probability of cooperation quickly converges to the steady state after less than twenty time slots. Moreover, the probability of cooperation in the steady state is close to one in the limit as noise in monitoring vanishes.

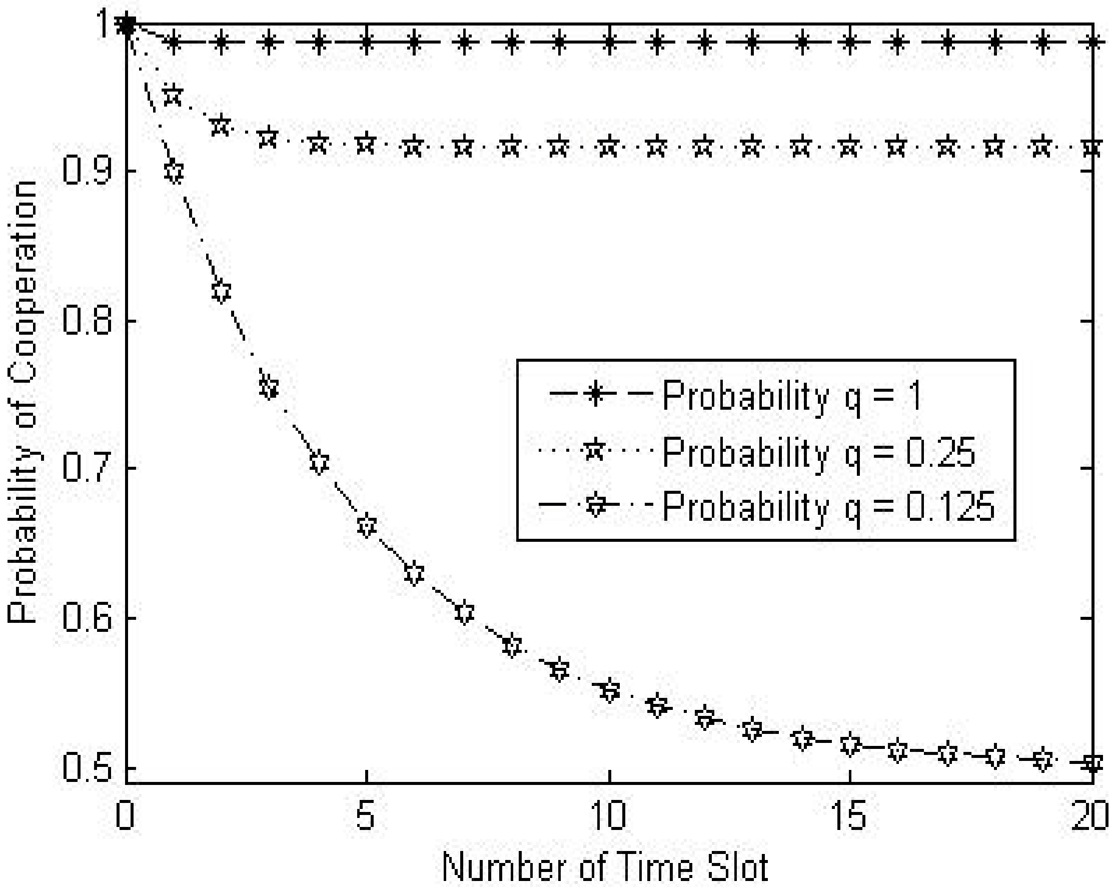

Figure 5.5: Variation of the probability of cooperation with time and *q*.

In Fig. 5.5, we fix $\varepsilon = 0.1$ and observe the change with q. The case $q = 1$ corresponds to a border node that requests only the packets it originates to be forwarded. Similarly, $q = 0.125$ will be a node close to the center for which 7 out of 8 packets it requests to forward come from other nodes. Recall that q can represent the proportion of packets that a node originates among the total packets a node requests to forward. As we can see, the probability of cooperation will not be the same for all nodes in the network with the same noise level. Saturated nodes will cooperate less.

We analyze our belief-free equilibrium strategy (BFES) in multi-hop communication. There are h hops between the sender and the destination ($h \in [1,10]$). We consider a channel lost probability of $\mu = 0.01$.

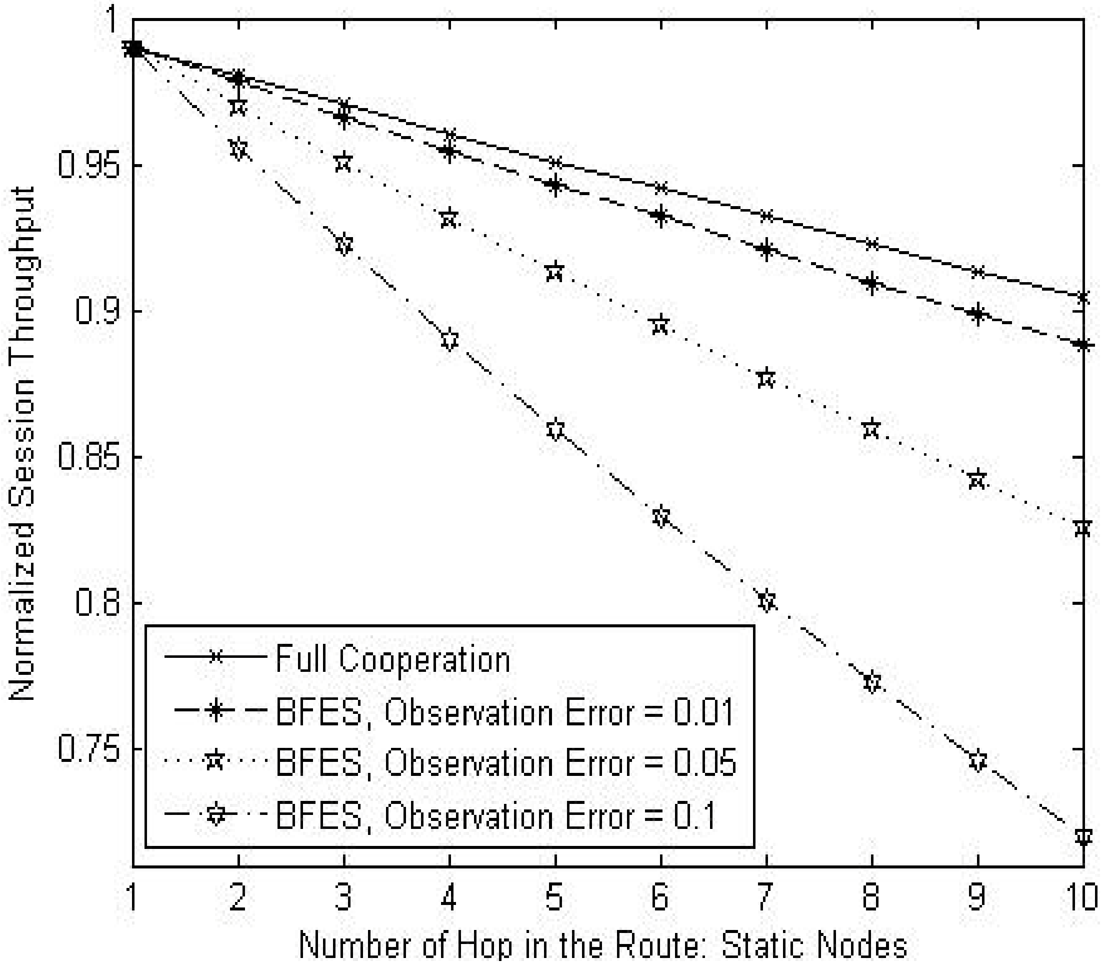

Figure 5.6: Variation of the normalized session throughput with the number of hop and noise for static nodes

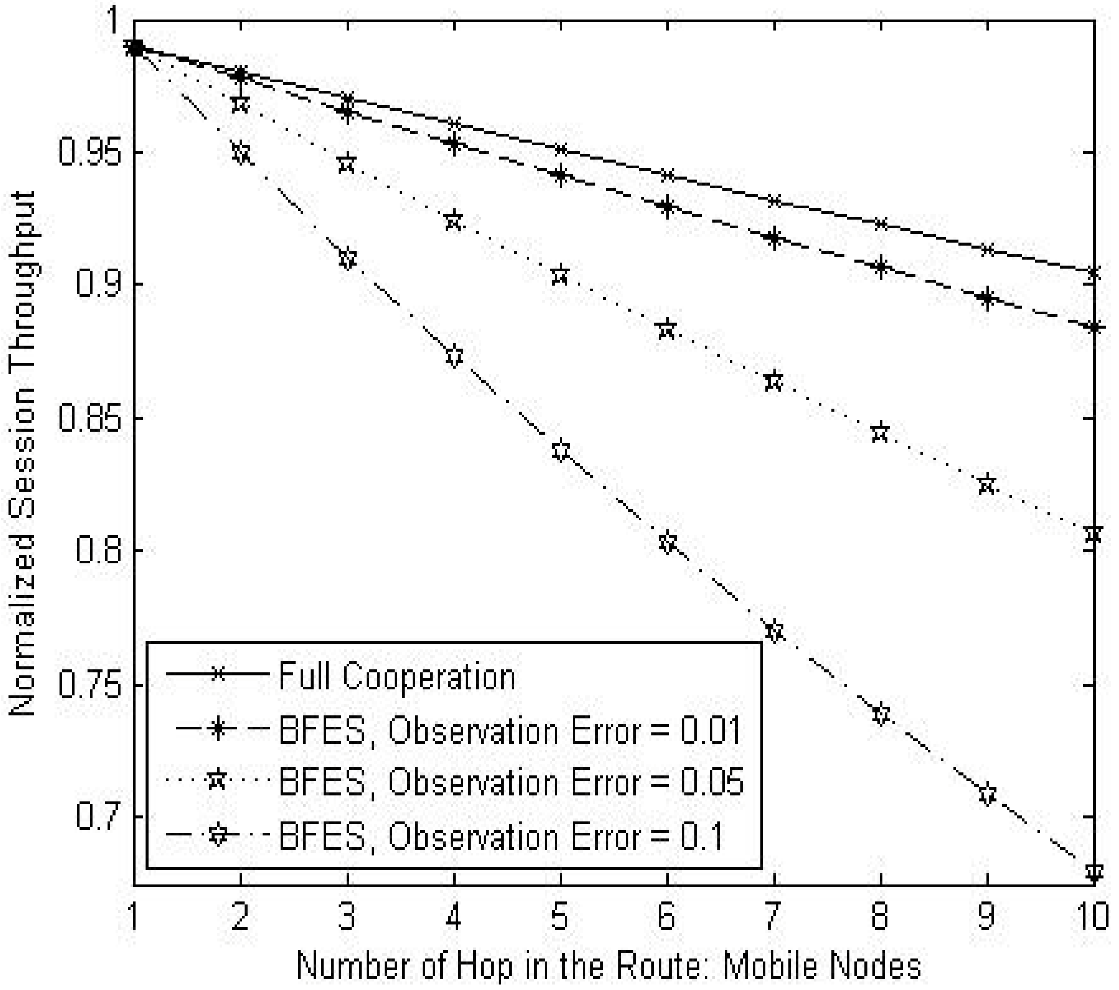

Figure 5.7: Variation of the normalized session throughput with the number of hop and noise for mobile nodes

Figures 5.6 and 5.7 represent the variation of the normalized session throughput with the number of hops in the route and noise for static and mobile nodes respectively. In the graph, full cooperation corresponds to a network with a central manager where node cooperates unconditionally. We can see that, with up to 5 hops, and 5% observation error, the normalized throughput decreases with BFES is less than 5% compared to that of unconditional cooperators both for static and mobile nodes.

In the game theoretical perspective, node mobility decreases the discount factor. This is because the expected number of future interactions decreases with node mobility. In fact, the expected number of future interactions is exponentially distributed with mean $\frac{1}{1-\delta}$. In the case of static nodes, $\delta = 0.999$ means that at the beginning of each time slot, each node expects the packet forwarding game to be repeated 1000 times. For mobile nodes, we decrease δ, $\delta = 0.8$.

5.5 Conclusion

We used the belief-free equilibrium concept to develop a packet forwarding game model. We took into account the fact that some inconsistency between a node's true action and its neighbors' observation of that action exists in a network. Our model is presented under imperfect private monitoring. A stochastic dimension is introduced to consider the different packet forwarding rates among nodes in ad hoc networks. Our two node model can easily be extended to a MANET of *n* nodes. Simulation results show that our model is robust to noise, traffic inequalities, and node mobility.

The model presented is simple and our packet forwarding game equilibrium relied on a single parameter: the probability of cooperation after observing a defection. That probability was adjusted such that the history of the game became irrelevant. However, it is not sure that the belief-free equilibrium approach will be robust if there is private payoff information. In fact, in the two node game for instance, each player needs to know its opponent payoff to perfectly balance the probability of cooperation after observing a defection in such a way that its opponent is indifferent between cooperation and defection after all histories. Considering private payoff information under a belief-free equilibrium approach is still an open research problem.

CHAPTER 6

MITIGATING ROUTING MISBEHAVIOR IN MULTI-HOP NETWORKS USING EVOLUTIONARY GAME THEORY

Multi-hop wireless networks have been an active research area for decades; however, the solutions proposed to solve routing misbehaviors are still not robust. In this Chapter, we use the EGT framework to address one issue of routing misbehavior, the problem of selfishness. We propose the use of distributed algorithms that are able to force selfish nodes to cooperate and forward packets from other nodes, despite their desire to "conserve energy" by not forwarding external packets.

6.1 Introduction

The lack of circumscribed infrastructure is a defining feature of the multi-hop wireless network, as practically all of the nodes in such a network must function as both a router and a client. For multi-hop traffic, in which the sender and receiver are not adjacent, communication depends upon the cooperation of intermediate nodes. This type of multi-hop wireless communication is an essential ingredient in WSN, ad hoc networks, P2P networks and wireless mesh networks (WMN). However, without a common routing authority, cooperation and trust between nodes cannot be guaranteed, and may result in network breakdown.

Nodes, in many application domains, including military applications, do share a manager in order to properly route traffic; however, this is not always the case. For example, networks made up of sensors embedded in devices to form a ubiquitous computing environment, such as health monitoring networks or VANET may provide an infrastructure where each vehicle or node can be equipped with one or more autonomous sensors without a common manager. In order for these networks to operate, each sensor (node) must forward packets from the others. Any node, especially a selfish node, could refuse to forward any packets except its own. If all nodes operate in this manner, the network will not function.

Packet forwarding failure is far more critical in multi-hop networks, as has been demonstrated by Tanachaiwiwat *et al* [69] where the research indicates that only a small percentage of misbehaving nodes has a large impact on the network. In their research, Tanachaiwiwat *et al*. [69] found that if 5% of the nodes in a grid-sensor network are misbehaving, 60% of routes are affected. In a randomly placed sensor network, 35% of routes are affected. If the number of misbehaving nodes increases to only 10%, the number of affected routes rises to an incredible 88% in a grid-sensor network and 54% in a randomly placed sensor network. In a mobile ad hoc network, throughput is reduced by 20% if 10% of the nodes in that network misbehave [32]. From the above data, it becomes obvious that a small number of misbehaving nodes creates large problems in the network.

In general, there are three types of misbehaving nodes in a multi-hop wireless network: faulty nodes, selfish nodes, and malicious nodes. Faulty nodes have defects that do not allow them to follow the recommendations of the protocol. Selfish nodes do not forward packets to save resources such as battery and bandwidth. Malicious nodes launch various attacks such as DoS.

In this Chapter, we focus on selfish nodes and present a model based on EGT in which we demonstrate that the model is able to encourage selfish nodes to cooperate and forward packets from others with only one period of punishment if nodes are sufficiently patient. For the case of impatient selfish players, we recommend more sophisticated algorithms with limited punishments.

In Section 6.2, we present our game model. Next, in Section 6.3, we describe some strategies from other researches that are useful in refining the model and which complement the model by their analysis as outlined in Section 6.4. In Section 6.5 we present our simulation results and in Section 6.6 our conclusions.

6.2 Game Model

The assumptions presented in Section 3.3 apply here. In multi-hop networks, sending, receiving, or forwarding packets are random processes. If all nodes cooperate by forwarding other nodes' packets, all nodes will benefit, but each of them will have to expend some energy to forward others' packets. Also, a selfish node in a network of cooperating nodes benefits even more because it does not expend any energy to forward packets from others. In fact, each node is better off choosing defect rather than cooperate regardless of the number of nodes who choose to cooperate. However, the network collapses if all nodes behave selfishly and then all nodes loose.

This situation is similar to the Prisoners' Dilemma game found in game theory and we use it as the base of our model. In this game, the players are the *n* nodes, $n >> 1$. Pairs of nodes are repeatedly drawn with equal probability to play the Prisoners' Dilemma game. The nodes are programmed to play the same pure or mixed strategy. For the stage game, the pure strategies of a node are: Cooperate (C) or Defect (D). For the repeated game, the strategies of a node are any deterministic or random combination of C and D depending on the history of the game.

From the previous assumptions and for consistence with the notation in the Prisoners' Dilemma game, the payoff of the stage game is calculated as follows. We assume that all nodes spend the same energy at the cost of β to send or forward a packet. All nodes have packets to send and the expected gain for cooperation, γ, is equal for all nodes. When a node drops a packet as a result of defection, we assume that there is no cost or gain for that node. Thus, if both players cooperate, each of them gets the reward price $R= \gamma - \beta$. This is because each node gains γ for having its packet forwarded, but has to spend the energy of cost β to send a packet for the others. Nodes get the punishment price $P = 0$ for mutual defection. This is justified by the fact that no nodes spend or gain anything. When one defects and the other cooperates, the defector gets the temptation price $T = \gamma$ when the cooperator gets the sucker price $S = -\beta$.

We assume that the following inequalities hold: $\gamma>\beta>0$. In fact, it will not be globally more efficient to forward packet than to discard packets when $\gamma<\beta$. We can observe that $T>R>P>S$. Moreover, we have: $2R>T+S$. This makes two runs of mutual cooperation collectively more efficient than an alternation of cooperation and defection. The stage game of our model is represented in Table 6.1.

We can see that Cooperate is strictly dominated by Defect. Therefore, the only Nash equilibrium is Defect. The two players get the punishment prize P. However, if the two players choose to cooperate, the two players will get the Pareto efficient outcome R. We can see that when rational intelligent players play the game, they will achieve a non-efficient outcome. This illustrates that the Nash equilibrium is not always Pareto efficient. Nevertheless, when the game is played repeatedly, cooperation among players can emerge. Players could cooperate expecting future gain and reach an efficient equilibrium while all players cooperate.

In our game model, we assume that no single node knows the end of the game and that at any time; all nodes believe that there is a large probability ω that the game will continue. Then, future payoffs are discounted by ω. Also, at any time, the expected duration of the game is $\frac{1}{1-\omega}$ time slots. Consequently, the game is infinite if ω=1. $\omega < 1$ in our model.

TABLE 6.1: THE STAGE GAME MODEL IN STRATEGIC FORM

		Player *j*	
		Cooperate	Defect
Player *i*	Cooperate	$\{\gamma - \beta, \gamma - \beta\}$	$\{-\beta, \gamma\}$
	Defect	$\{\gamma, -\beta\}$	$\{0,0\}$

6.3 Strategy Description

The Iterated Prisoner’s Dilemma is the Prisoner’s Dilemma repeated over time. This has been used to model similar situations found in other fields of science such as biology and economics, where researchers from multi-disciplinary fields have investigated cooperative behavior [70-72]. Memory one strategies, such as these in which present actions depend only on the most recent action, are suitable for wireless networks due to the nodes' limited battery and computational power. Moreover, memory one strategies are easier to model, as they can be described by a finite state machine, a Markov Process or an automaton. The ultimate goal is to find the best strategy that will result in cooperation in the network without being exploited by defectors.

Axelrod [72] simulated the IPD in a tournament using different strategies. In his simulation, TFT was the winner. A TFT player begins with cooperation and then repeats the opponent's action in the last move. However, TFT is a strategy vulnerable to error in perception or stochastic perturbation. When two players use TFT, if one cooperates and the other, because of noise perceives a defection, the result will be an alternating cycle of cooperation and defection resulting in a non-efficient payoff.

Nowak and Sigmund [70] simulated all randomly generated strategies with memory one in the presence of noise and identified the emergence in the long run of a strategy called Pavlov. This strategy outperforms TFT. A Pavlov player cooperates if and only if the player and the opponent used the same move in the previous round. Unlike TFT, Pavlov is immune to errors in perception.

Boerlijst *et al*. [71] improved Pavlov using tagging to produce *p*Pavlov. *p*Pavlov is a memory one strategy like Pavlov and TFT. However, *p*Pavlov applies two periods of punishment instead of one in Pavlov. *p*Pavlov follows Pavlov in most case. However, *p*Pavlov has two taggings for defect: D_0 and D_1. *p*Pavlov normally plays D_1, and plays D_0

only after a mutual defection or an erroneous defection. In a game between two *p*Pavlov players, an erroneous defection is followed by two rounds of mutual defection and then mutual cooperation. In this case, the successive states are: (C, D_0), (D_1, D_1), (D_0, D_0), (C, C). *p*Pavlov, like other memory one deterministic strategies, can be represented by a finite state machine.

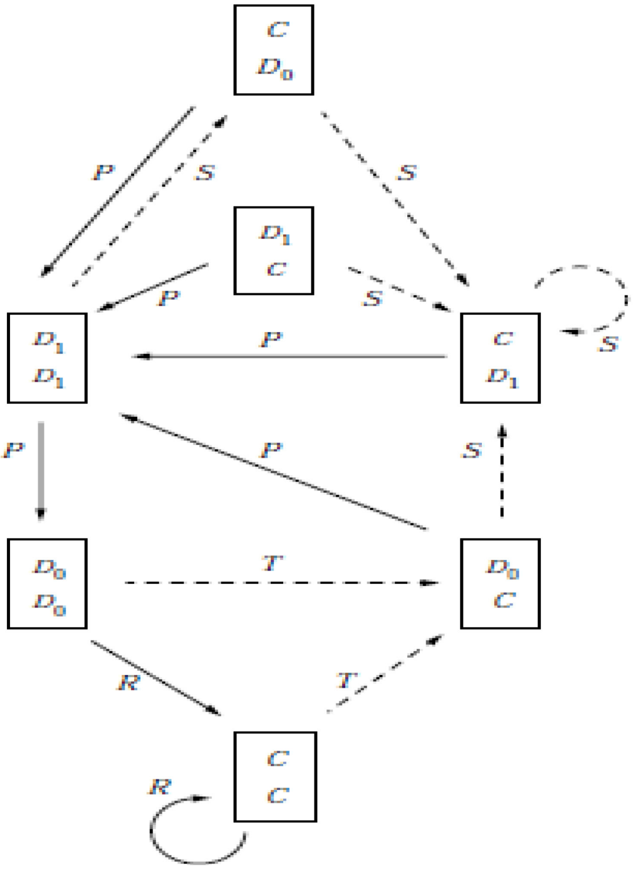

Figure 6.1: Finite state machine of *p*Pavlov.[71]

Fig. 6.1 gives the finite state machine of *p*Pavlov. The vertices of the graph represent the states. The *p*Pavlov player state is at the lower position and the opponent state is in the upper position. The opponent can choose to Cooperate (C) or Defect (D). Therefore, from each state, we have two transitions. The arrows indicate the transitions. Solid lines indicate the moves specified by the *p*Pavlov strategy; dotted lines indicate the alternative moves. One advantage of *p*Pavlov is that *p*Pavlov only monitors its own tagging and is immune to errors in perception. Since *p*Pavlov applies two periods of punishment instead of one, a strategy based on *p*Pavlov is safer against defectors than those based on the classic Pavlov. An opponent who always chooses Defect takes advantage of Pavlov every two rounds but only exploits *p*Pavlov every three rounds. Similar memory one strategies can be designed using the appropriate tagging to provide any number of punishments before returning to mutual cooperation. The extreme case is Grim which punishes forever after a defection.

6.4 Analysis

We will use the framework of EGT to analyze the previous strategy. EGT has its foundation in game theory and evolutionary biology. Evolutionary biology studies the processes of change in populations of organisms. Selection and mutation are two important mechanisms of change in evolutionary biology. Mutation provides diversity in the population, whereas selection promotes some characteristics over others. Therefore, two key concepts of EGT are ESS that deals with mutation mechanisms and the replicator dynamic that deals with selection mechanisms. The assumptions of rationality and common knowledge used in game theory are relaxed in EGT. Unlike game theory, where players are intelligent, rational and choose strategies, in EGT, players are programmed to perform some strategies in the game. Players are randomly and repeatedly drawn from large populations and play the same pure or mixed strategies. The payoff is the individual fitness or expected number of surviving offspring. Now, let us be more formal in our analysis. We start with the ESS concept followed by the replicator dynamic.

A strategy x is an ESS if and only if:

$$u[x, \varepsilon y + (1 - \varepsilon)x] > u[y, \varepsilon y + (1 - \varepsilon)x], \tag{1}$$

for a small proportion of mutant ε.

u is the utility function or payoff

x and y are the incumbent and mutant strategy, respectively.

Definition 1: A strategy x is an ESS if for every strategy $y \neq x$ there exist some $\bar{\varepsilon_y} \in (0,1)$ such that inequality (1) holds for all $\varepsilon \in (0, \bar{\varepsilon_y})$.

Considering the linearity of u, (1) can be written as:

$$(1 - \varepsilon)u(x, x) + \varepsilon u(x, y) > (1 - \varepsilon)u(y, x) + \varepsilon u(y, y)$$

For small values of ε, (1) is equivalent to either

$$u(x, x) > u(y, x)\ \forall y, \tag{2}$$

or

$$u(x, x) = u(y, x) \text{ and } u(x, y) > u(y, y)\ \forall y \neq x. \tag{3}$$

Equations (2) and (3) represent ESS as first formulated by Smith in [73].

A population of individuals programmed to play an ESS resist against mutation. This means that such a population cannot disappear in the long run evolution.

Let us turn our attention to the replicator dynamic. As described previously, ESS focuses on mutation while the replicator dynamic highlights the role of selection. This model was first proposed by Taylor and Jonker [74]. Formally, the population dynamics for the population shares x_i is given by:

$$\dot{x}_i = \left[u\left(e^i, x\right) - u(x, x)\right] x_i. \tag{4}$$

$u\left(e^i, x\right)$ is the expected payoff to pure strategy *i* at a random match.

$u(x, x)$ is the payoff of an individual drawn at random in the population or the average payoff per individual.

x_i is the proportion of individuals programmed to pure strategy *i* and its time derivative is $\dot{x}_i$.

Equation (4) is a system of differential equations that allows us to draw an important conclusion. Subpopulations programmed with better than average strategies grow whereas subpopulations programmed with worse than average strategies sink. Another important factor to note is that an ESS is stable in the replicator dynamic (4). In the next Section, our simulator implements the discrete time version of the replicator dynamic.

In [71], the authors proved that Pavlov is an ESS when the probability ω of a next round is such that:

$$\mathrm{R} + \omega\mathrm{R} > T + \omega P. \tag{5}$$

Pavlov is ESS, meaning here that if there is a small, but non-zero probability of mis-executing a move, every strategy that departs from what a Pavlov rule would command will gain less against a Pavlov player than it would have by following the Pavlov rule.

Now, let us see why pPavlov is also ESS. pPavlov is ESS if in the finite state machine of pPavlov in Fig. 6.1, the payoff discounted step by step following the branch of pPavlov is always higher than any alternative payoff. This is the case if:

$$\mathrm{R} + \omega\mathrm{R} + \omega^2\mathrm{R} > T + \omega P + \omega^2\mathrm{P}. \tag{6}$$

The probability ω is less for pPavlov compared to that of Pavlov. This is good because players can cooperate even if a next round is less probable or when players are less patient. To better comprehend (5) and (6), take P=0 as it is in our game model. In fact, the more severely a strategy punishes, the lower the value of the discount factor ω needed for that strategy to be ESS. Thus, when Pavlov is ESS, pPavlov is also ESS under the same condition. Furthermore, in the extreme case of infinite punishment, Grim is ESS if:

$$\frac{R}{1-\omega} > T. \tag{7}$$

Equation (7) shows us that in the extreme case of Grim, we can have an ESS if ω is very close to 1. However, Grim is not a wise solution in wireless networks because occasional noise or congestion can damage cooperation forever. TFT can also mitigate selfishness but TFT is not ESS. In fact, TFT can be invaded by unconditional cooperators which in turn are invaded by defectors.

To summarize, Pavlov and *p*Pavlov are ESS. This implies first that they cannot be invaded. Second, Pavlov and *p*Pavlov are also dynamically stable in the replicator dynamic. Third, in a network of nodes following Pavlov or *p*Pavlov, no selfish node will benefit by dropping packets and saving energy if (5) and (6) hold respectively. Thus, selfish nodes will be constrained to cooperate and cooperation will be maintained for the good of each node. For the three reasons above, we first recommend Pavlov to mitigate selfish behavior in multi-hop networks. However, when the probability ω is not high enough for (5) to hold, a memory one strategy, using tagging which applies more than one period of punishment, like *p*Pavlov, must be used.

6.5 Simulations

TABLE 6.2: THE STAGE GAME PAYOFF

		Player *j*	
		Cooperate	Defect
Player *i*	Cooperate	{*4,4*}	{-*1,5*}
	Defect	{*5,-1*}	{*0,0*}

In this Section, we use the simulator from [75] to present our simulation result. This simulator implements a round robin tournament combined with the discrete time version of the replicator dynamic (4). The vertical axis in all figures shows the population share that is related to the payoff by (4). The stage game payoff is shown in Table 6.2. Those payoffs are for β = 1and γ = 5. This can be justified since a packet will go through more than five intermediates nodes before reaching the destination in a multi-hop network, on average. Thus, for global efficiency we should have at least $\gamma = 5\beta$. Let us analyze a few scenarios.

A. Cooperator vs Defector

In this scenario, we have two types of nodes. The first type is preprogrammed to the pure strategy that always defects whereas the second type always cooperates. Originally, there are 25 defectors and 75 cooperators. Following the payoffs in Table 6.2, we have:

u(Cooperate*,* Cooperate) = 4 and

u(Defect*,* Cooperate) = 5 . Then,

u(Cooperate*,* Cooperate) $<$ u(Defect*,* Cooperate)

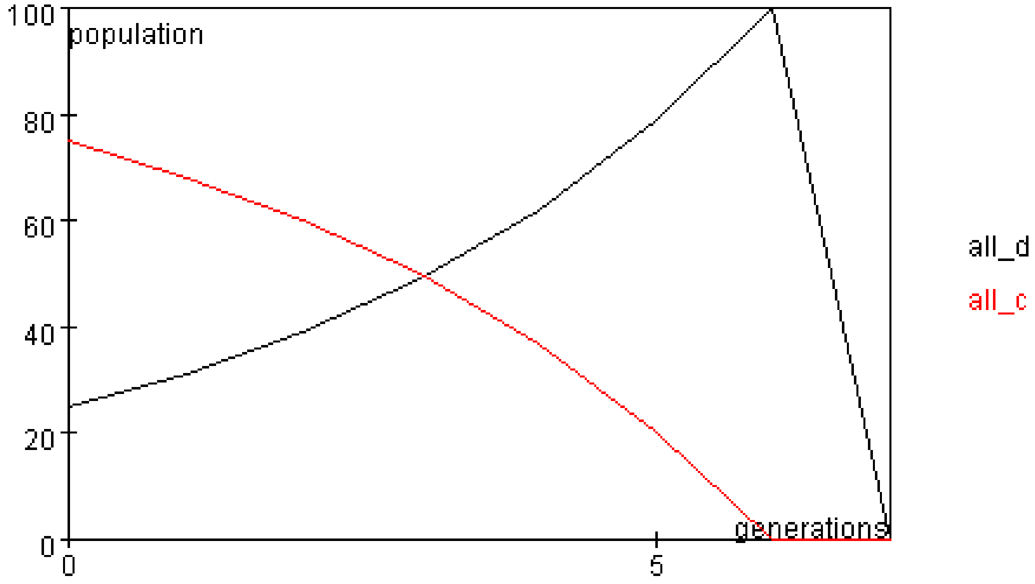

Figure 6.2 Dynamic of Cooperator vs Defector

It follows from (2) that Always Cooperates is not an ESS. Fig. 6.2 shows that Cooperators are invaded by Defectors. The population and payoff of Defectors increase whereas that of Cooperators decrease and become extinct after six generations. Next, confrontations only happen between Defectors which yield a payoff of zero. That explains the fall back to zero after the maximum.

B. Pavlov vs Defector

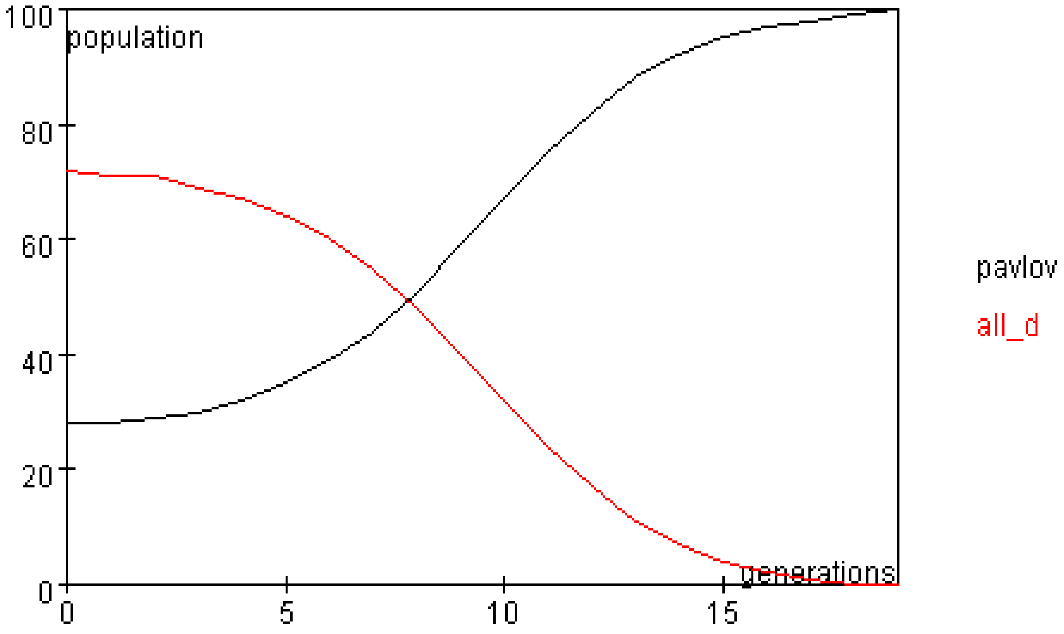

Figure 6.3 Dynamic of Pavlov vs Defector

In this scenario and the following analysis, we consider the performance of Pavlov, which as we discus in Section 6.4, is similar to *p*Pavlov. Fig. 6.3 shows the increase in payoff and populations of Pavlov. At the same time, the payoff of Defectors converges to zero after only 20 generations.

C. Pavlov, Defector and cooperator

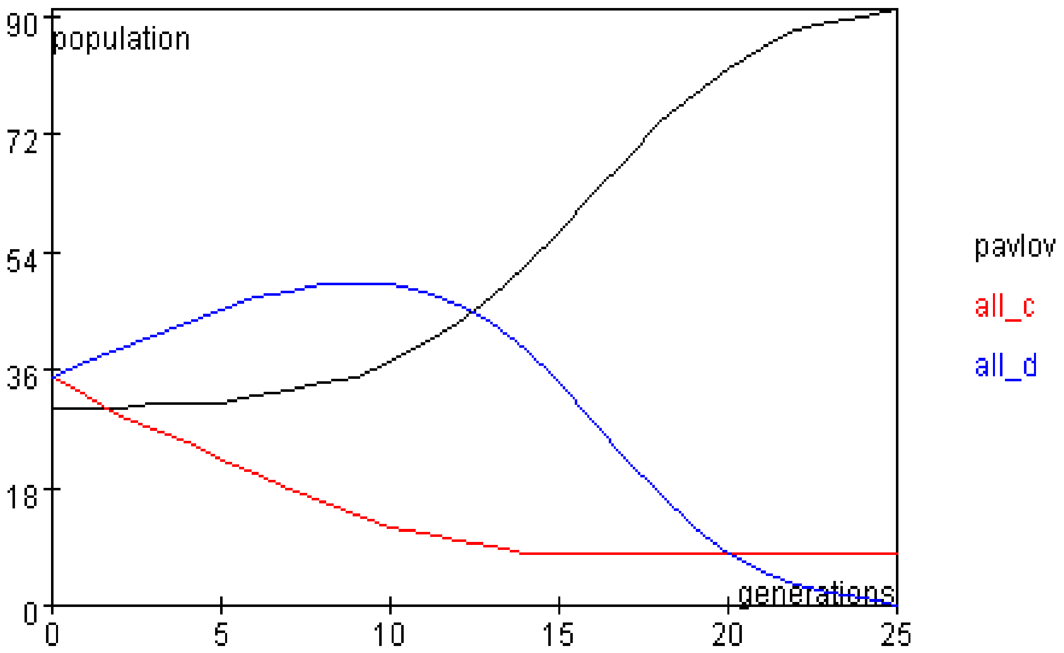

Figure 6.4: Dynamic of Pavlov, Defector and Cooperator

At the beginning, we have a mixture of 35 Cooperators, 35 Defectors, and only 30 Pavlov. After 25 generations, only Pavlov and Cooperator survive. A few Cooperators survive because Defectors vanish due to interactions with Pavlov. Note that interaction between Pavlov and Cooperators yield mutual cooperation and then the same payoff for both strategies. This is the reason why a few Cooperators can survive. In case of noise or stochastic perturbation that make monitoring imperfect, no Cooperator can survive. This is because Pavlov exploits unconditional Cooperators by defecting when they cooperate.

D. Pavlov, Defector, cooperator and Random

This scenario tests the effect of faulty and malicious nodes. Faulty nodes play randomly because they are damaged. Malicious nodes can play Always Defect to perform a DoS.

The simulation result in Fig. 6.5 shows a complete domination of Pavlov. With an initial population of 25 each, only Pavlov survives. The payoff and population share of Defectors starts increasing until the Cooperators and the Random are extinct. Then, Defectors only oppose Pavlov resulting in the decrease of payoff and population share of Defectors.

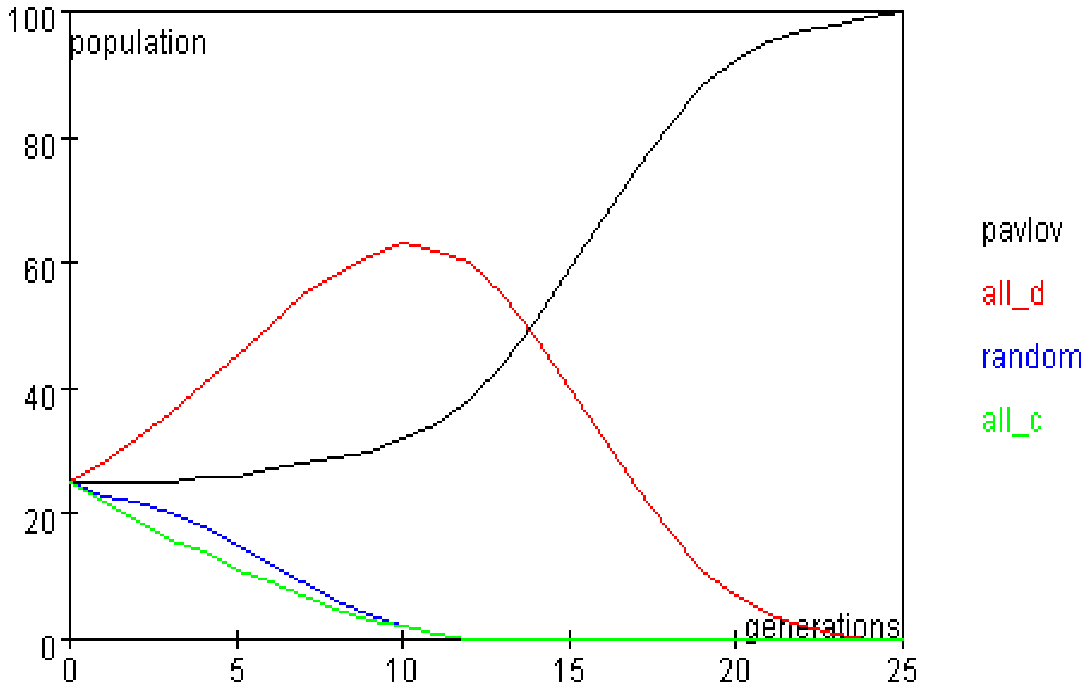

Figure 6.5 Pavlov, Defector, Cooperator, and Random

E. Pavlov, Defector, cooperator, Random, TFT

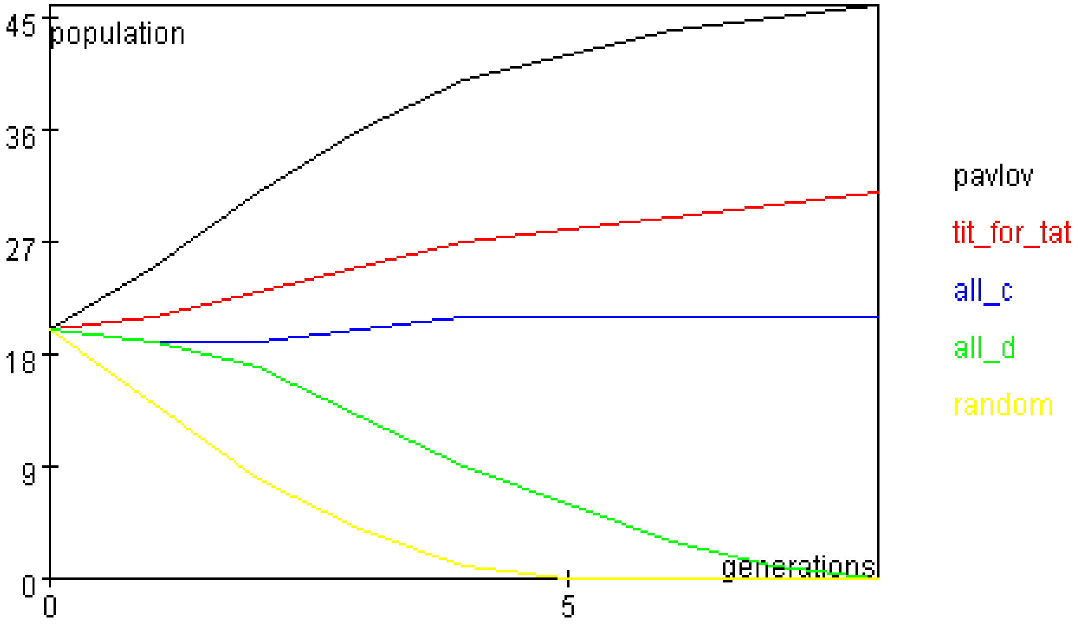

Figure 6.6: Pavlov, Defector, Cooperator, Random, TFT

We repeat the previous scenario adding TFT. The result in Fig. 6.6 shows Pavlov outperforming TFT as shown by Nowak and Sigmund in [70].

6.6 Conclusion

We have used EGT to recommend Pavlov and its extension *p*Pavlov for multi-hop networks. Both are ESS and then a Nash equilibrium of the packet forwarding game. Mathematical and simulation results show that both promote cooperation among selfish nodes. Pavlov and its extension are distributed algorithms and therefore require only local information. Those algorithms do not require a common knowledge of the network or any rationality of the nodes, as do models based on traditional game theory. *p*Pavlov has an advantage over Pavlov because it can force nodes to cooperate even though the discount factor is smaller.

In the future, we will implement Pavlov and *p*Pavlov in a real multi-hop network environment and evaluate their performance for different network flows and topology. We will also consider asymmetric links in the network.

CHAPTER 7

GAME THEORETIC MODELING AND EVOLUTION OF TRUST IN AUTONOMOUS NETWORKS: APPLICATION TO NETWORK SECURITY AND PRIVACY

Future applications will require autonomous devices to be interconnected to form a network. Such networks will not have a central manager; each node will manage itself and will be free to decide participation in any network function. As with traditional networks, these networks need to be secured to authenticate the nodes, prevent misuse, detect anomalies and protect user privacy. Network security and privacy protection without a central manager will be challenging. Several security mechanisms and privacy protections will require the cooperation of several nodes to defend the network from malicious attacks. We particularly investigate when, for each node, it is cost-effective to freely participate in the security mechanism or protect its privacy depending if that node believes or trusts that all other nodes, or at least a minimum number of other nodes, will do the same. In this case, each node will be involved in a trust dilemma that we will model using the mathematical framework of game theory and evolutionary game theory. The well known stag hunt game will be our basic game model. This Chapter will clearly present the interconnection between cooperation, trust, privacy, and security in a network.

7.1 Introduction

Multi-hop networks include ad hoc networks, sensor networks, peer-to-peer networks, and WMN. The main challenges in such networks are limited battery power, limited computational power, the lack of infrastructure (or limited infrastructure for WMN), and node mobility. In these networks, the sender and the destination directly communicate if they are in the power range of each other, otherwise, the sender and destinations communicate via multi-hop communication. Generally, a packet goes through several intermediate nodes before reaching the destination. All of these constraints make security and privacy protection in autonomous multi-hop networks more complicated compared to traditional networks.

Network security is closely related to trust. Security mechanisms in traditional networks rely on trusted systems like certificate authorities to operate. One of the differences is that security mechanisms can use efficient cryptographic algorithms to guarantee access control in the networks, but cannot detect and eliminate a malicious node that already participates in the network with all keys in its possession. To do that, we sometimes need a trusted system supported by a reputation system that requires each node to track the past behavior of its neighbors [31-34].

The nodes are autonomous if they have the freedom to choose their action and pursue only their self-interest. Autonomous nodes do not have a common manager. This is the case when autonomous devices are interconnected to form a network. For instance, in VANET, each car is equipped with a node. Nodes from different cars can communicate, but each node manages itself and is in charge of its own security in the network. Securing this autonomous network is a challenging problem because there is no central and trusted manager to protect the whole network. Thus, only distributed security mechanisms are viable in autonomous networks such as VANET.

Network security must rely on the cooperation of all nodes (or at least the majority of nodes) to protect the network. In many attack scenarios, a single node will not be able to defend itself if other nodes do not participate in the defense mechanism. However, a node will be protected if it defends itself and a minimum number of other nodes (the threshold) participate in the defense mechanism. If the threshold is not reached, a node that defends itself is worse off compared to a node that does not defend itself. This is because a node that defends itself wastes resources and ends up not being protected. However, if the threshold is reached, it is better for all nodes to participate in the defense mechanism to protect itself; thus, the network will be secured. As a result, if a node has a high expectation (trust) that the other nodes will participate in the security mechanism, it will also participate. A node should not participate otherwise.

This scenario describes the strong connection between trust, cooperation and security. In multi-hop network, trust is the confident expectation that the other nodes will cooperate and participate in the security mechanism or any other network function. Trust can help facilitate cooperation and security. Moreover, when the interactions are repeated, cooperation increases a node's reputation and therefore its trustworthiness. Privacy relates to security in the sense that security mechanisms need to be implemented and enforced to protect user privacy.

As we can see, network security and privacy in autonomous networks can be modeled as a strategic interaction. Any node's decision to participate or not in the security mechanism affects the decision of other nodes and the final result i.e., a secure or insecure network, protected or unprotected private information. Game theory is the branch of applied mathematics that formalizes strategic interaction among autonomous rational agents. If the agents are not assumed to be rational, EGT applies. We will use the game theoretic framework to model trust and use EGT to capture the dynamic evolution of trust behavior in the network. We will apply our result to network security and privacy.

Trust is a fundamental concept that has been studied across several disciplines including sociology, psychology, anthropology, philosophy, economy, political science, and theology. To the best of our knowledge, this is the first work that models trust in multi-hop networks in the framework of game theory and EGT.

The remainder of this Chapter is organized as follows: in Section 7.2, we present some basic principles on applying game theory to network security. In Section 7.3, we present a few scenarios where a network requires mutual trust to be effective, and model them as a game. In Section 7.4, we analyze the evolution of trust using EGT. Section 7.5 presents our simulation results and Section 7.6 concludes the Chapter.

7.2 Background on Game Theory and Evolutionary Game Theory Applied to Network Security

Most of the previous works that apply game theory to network security assume a network with a central manager. In those works, network security is modeled as a game between the central manager and an attacker. The attacker has several strategies at each layer.

At the physical layer, an attacker can create interference or jam the wireless media. At the MAC layer, an attacker can perpetrate a MAC address spoofing, an address resolution protocol (ARP) attack, or a dynamic host configuration protocol (DHCP) starvation. At the routing layer, the attacker strategies can be address spoofing, Internet protocol (IP) fragmentation attack, man in the middle attack, sinkhole attack, selective forwarding, DoS and distributed denial of service (DDoS). At the transport layer, the attacker can employ user datagram protocol (UDP) flooding, internet control message protocol (ICMP) flooding, TCP session poisoning, and TCP SYN flooding. At the application layer, the attacker can use domain name server (DNS) poisoning, introduction of viruses, worms and Trojans, introduction of sniffers, read, add, delete or modify data, or guest username and password. An attacker can also use a combination of these strategies. Moreover, there can be several attackers.

To protect the network, the strategies available to the manager are for instance: use antivirus software, use cryptographic techniques, install an intrusion detection system (IDS) and/or an intrusion prevention system (IPS), move critical data between different hosts, reconfigure the network, shut down the network, or a combination of these strategies. The network vulnerabilities and the defense mechanisms are presented in more detail in [76].

The utility function of the network security game maps each attacker and defender strategy to a payoff. This payoff depends on the cost to implement each strategy, the damage that a successful attack can create to the network, and the benefit of a successful attack to the attacker.

In the game presented in [40], the strategy and the payoffs are assumed to be common knowledge. In this case, the network security game is a game of complete information.

Otherwise, we have a game of incomplete information that can be formulated as a Bayesian game as in [41]. The network security game can also be modeled as a static game [40-41], a repeated game, or more generally as a stochastic game [38, 42]. A stochastic game is a generalization of a repeated game. In a repeated game, players play the same stage game in all periods, whereas in a stochastic game, the stage game can randomly change from one period to the next.

In all these games, the main objective of the attacker is to intelligently choose its strategy to maximize the damage to the network while the manager tries to minimize the damage. The attacker's and the defender's objective are strictly opposed. This justifies the use of a zero-sum game to model network security [38]. When each player applies the best response to its opponent strategy, the game reaches the well known Nash equilibrium. Neither the attacker nor the manager can unilaterally make a profitable deviation from the Nash equilibrium. Other models relax the assumption of rationality used in game theory and use the mathematical framework of EGT to model network security [39].

In an autonomous network, packet forwarding is a challenging problem because of the cost associated with forwarding packets. If not well addressed, the result will be selective forwarding and DoS, which are both important security concerns. A few works attempt to mitigate selfish routing misbehavior in the framework of game theory [1-3, 6-11, 13-18] and evolutionary game theory [4-5]. Game theory also provides a solid framework to model intrusion detection in a network [41, 43]. A survey of game theory as applied to network security is provided in [44]. A detailed presentation of game theory and EGT is found in [35, 60] and [45] respectively.

To summarize, game theory and EGT provide useful mathematical tools to analyze network security. If the game has a unique equilibrium, the manager can easily predict the attacker's behavior. However, besides selective packet forwarding, the very important case of security and privacy protection for autonomous nodes where there is no central manager is not yet well investigated. This is investigated in the next Section. Specifically, the main contribution of this Chapter [21] is to investigate security mechanisms and privacy protection schemes that require the cooperation of several nodes or agents to be effective and highlight the importance of trust in such a scenario.

7.3 Network Security and Privacy Protection in Autonomous Networks

Security in autonomous networks no longer depends on a central manager as was the case in the previous Section. Moreover, a node alone cannot defend itself from some of the attacks presented in the last Section. To achieve network security, autonomous nodes need to cooperate with and trust other nodes. Cooperation and trust will also be required for privacy protection. We give two examples here to motivate our model.

The first example is about privacy protection. In autonomous networks such as VANET, a subset of users can decide to share their private information. As in social networks such as Facebook, each user will at first decide to share its private information with another or not. Sharing private information with a friend has several benefits. For instance, a friend can have quick information about your location, your destination, or your daily and weekly schedule. You can also get from your friend private and useful information freely that otherwise would be costly to get. However, a malicious user or an enemy can undermine your reputation or cause several other damages from your private information. Protecting your privacy is successful if and only if you and all other persons or agents in possession of your private information also protect it. Moreover, private information protection in a network is costly. Not only do you need to buy expensive antivirus and authentication software to protect your privacy but also all your friends need to do the same. It will be cost effective to protect your private information if and only if you trust that all of your friends are also protecting that information. The decision to protect your private information or not will depend on how much you value your privacy, the cost to protect it, and foremost, the trust you have in others already in possession of that private information. We will later formalize those relationships.

The second example comes from the work of Gupta *et al.* [23] extended by He *et al.* [22]. Gupta *et al.* [23] and He *et al*. [22] present a light weight cryptographic technique, a distributed authenticated key establishment scheme for WMN [22] and cellular-based heterogeneous wireless ad hoc networks [23] based on hierarchical multi-variable symmetric functions. They provide a method to generate a symmetric secret key to allow encrypted communication and authentication among any two nodes. Other interesting properties of their algorithm are: key independence, random generation, mobility, and handoff management. However, their algorithm uses a polynomial of degree *k* and is therefore *k*-secure [22-23]. This means that if *k* nodes do not protect their key, an attacker can break the security mechanism and have access to all encrypted communication. If that happens, there is no need for a node to encrypt the message because they can easily be decrypted. A rational node may even prefer not to encrypt because encrypting a message will have no other purpose than creating a useless and costly message overhead. This is just an illustration. In fact, several security mechanisms in distributed autonomous networks need the solidarity, cooperation and trust of several nodes to be implemented.

In the two examples above, the dilemma that each autonomous node faces when freely deciding to protect its private information or to participate in the security mechanism can be formalized as a game. This is a well known dilemma first presented by the French philosopher Jean-Jacques Rousseau. The dilemma is called the *stag hunt* game. We will first present the two players stag hunt game before generalizing it to *n* players.

A. The two nodes' trust game

Let us consider a network with two autonomous nodes, node *i* and node *j*. Each node has two strategies, protect (*P*) or not protect (*NP*). The network is secured or both nodes' privacy is protected if and only if both nodes choose *P*. The reward from security or privacy for each node is γ. The cost to a node to protect itself is β. We make the reasonable assumption that the reward from security or privacy exceeds the cost to protect itself. Thus, we have $\gamma > \beta > 0$. The nodes decide simultaneously. Table 7.1 represents the strategic form of this game.

TABLE 7.1: TWO NODES TRUST GAME

		Node *j*	
		Protect	Not Protect
Node *i*	Protect	$\{\gamma\text{-}\beta, \gamma\text{-}\beta\}$	$\{\text{-}\beta, 0\}$
	Not Protect	$\{0, \text{-}\beta\}$	$\{0; 0\}$

If both players play *P*, each player gets the payoff *γ-β* because they spend resources to protect themselves at the cost of *β* and get the reward *γ* from being protected. On the contrary, if both players play *NP*, each player gets the payoff of zero because no player spends any resource and no player is rewarded. Finally, if the two players adopt different strategy, the player that plays *NP* gets zero and the one that plays *P* gets the negative payoff *–β*. This is because playing *P* needs both players to be successful.

This game is the stag hunt game. Contrary to the Prisoners' dilemma game which has a unique Nash equilibrium, The stag hunt game presents two pure strategies Nash equilibrium, (*P, P*) and (*NP, NP*) and a mixed strategy equilibrium where each node plays *P* with probability $\frac{\beta}{\gamma}$ and plays *NP* with probability $1-\frac{\beta}{\gamma}$. The equilibrium (*P, P*) is Pareto efficient with both players protects and each player gets the payoff *γ-β*. The other equilibrium, (*NP, NP*) is less efficient, no node protects and each node gets zero. Recall that an allocation is Pareto efficient if there is no other allocation that can make at least one individual better off without making any other individual worse off. We can also say that the equilibrium (*P, P*) *payoff dominates* [35] the equilibrium (*NP, NP*) because $\gamma-\beta>0$.

Let u_i represents the utility function of node *i*. The *resistance* [35] of the equilibrium (*NP, NP*) against the equilibrium (*P, P*) is the largest number λ such that $0\le\lambda\le 1$ and

$$u_i\left(\left(\lambda P_j+(1-\lambda)NP_j\right),NP_i\right)\ge u_i\left(\left(\lambda P_j+(1-\lambda)NP_j\right),P_i\right),$$

$$\Rightarrow \lambda u_i\left(P_j,NP_i\right)+(1-\lambda)u_i\left(NP_j,NP_i\right)\ge \lambda u_i\left(P_j,P_i\right)+(1-\lambda)u_i\left(NP_j,P_i\right),$$

$\Rightarrow 0 \geq \lambda(\gamma - \beta) + (1 - \lambda)(-\beta),$

$$\Rightarrow \lambda \leq \frac{\beta}{\gamma}. \tag{1}$$

Taking the largest number λ, the resistance of (*NP, NP*) against (*P, P*) is $\lambda = \frac{\beta}{\gamma}$. Similarly, the resistance of the equilibrium (*P, P*) against the equilibrium (*NP, NP*) is $1 - \lambda = \frac{\gamma - \beta}{\gamma}$. The equilibrium (*NP, NP*) *risk dominates* the equilibrium (*P, P*) if and only if the resistance of (*NP, NP*) against (*P, P*) is greater than the resistance of (*P, P*) against (*NP, NP*). This means that $\frac{\beta}{\gamma} > \frac{1}{2}$. Also, $\frac{\beta}{\gamma}$ is called the *risk factor* for the equilibrium (*P, P*).

Another way to understand the risk factor and trust is as follows. Node *i* is uncertain about what node *j* will do but believes that some distribution over node *j*'s strategies predicts its behavior. Consequently, node *i* will choose its own strategy to maximize its own expected utility payoff. For instance, if node *i* assigns the probability π_{ij} to the event that node *j* will play *P*. We can say that π_{ij} is the trust that node *i* has for node *j*. The expected payoff of node *i* when playing *P* against node *j* is:

$$E[u_i(P_i)] = \pi_{ij}(\gamma - \beta) + \left(1 - \pi_{ij}\right)(-\beta). \tag{2}$$

On the other hand, node *i* always gets zero when playing *NP* against node *j*. Thus,

$$E[u_i(NP_i)] = 0. \tag{3}$$

Therefore, node *i*, acting rationally will maximize its expected utility and play *P* if and only if:

$$E[u_i(P_i)] \geq E[u_i(NP_i)] \Rightarrow \pi_{ij} \geq \frac{\beta}{\gamma}. \qquad (4)$$

Therefore, even though the equilibrium (*P, P*) payoff dominates the equilibrium (*NP, NP*) in this game, the two players may fail to protect the network or their privacy if one of them does not trust the others to play *P*. In other words, it is risky for a node to play *P* because that node may lose β if the other node does not also play *P*. For that reason, both nodes may end up playing *NP* resulting in the equilibrium (*NP, NP*). In short, the more efficient equilibrium where each node protects itself and consequently protects the entire network is reached only if the two nodes trust each other.

We acknowledge that the possibility of future interactions in other game models can promote cooperation and trusting behavior because nodes want to maximize the long term utility [1-20]. However, in the privacy game example, once players' privacy is compromised, they may not be interested in protecting it in the future. Other models of trust in the network use a reputation system [31-34]. A node reputation is calculated as a function of the past behavior. A node's trustworthiness is the expected value of its reputation. In our model, traditional mechanisms based on past or future interactions that create trust are not available. Nevertheless, trust can denote the disposition to engage in uncertain and risky interactions with others with the possibility of reward or lost.

B. The n nodes' trust game

There are several possible extensions of this basic two-node game to *n* nodes in an autonomous network. Let us start with the privacy protection game. We consider a subset of *n* nodes sharing private information. For the sake of simplicity, we suppose that the game only has two outcomes: full privacy if all agents sharing the private information play *P* and no privacy if one of them plays *NP*. As in the two-node game, nodes that play *P* receive a payoff *γ-β* if the *n-1* other nodes also play *P* and receive a payoff *–β* otherwise. Nodes that play *NP* always receive a payoff of zero.

Therefore, a node will prefer to play *P* if and only if it believes that *n-1* other nodes will also play *P*. Otherwise, a node prefers to play *NP*. To simplify the analysis, we present the symmetric case when the trust that the nodes have for each other is the same. We note the value of that trust π. Formally, after similar development as in (2), (3), (4) a node prefers to play *P* if and only if:

$$\pi^{n-1} \geq \frac{\beta}{\gamma}. \tag{5}$$

After the privacy game, let us analyze the security game. We consider *n* nodes in a distributed autonomous network. We also consider a security mechanism that requires the cooperation of a fraction of nodes to be successful. We note that fraction *α*, $0 < \alpha \leq 1$. Let us consider a *k*-secure algorithm in a network of *n* nodes. If $k \leq n$, the fraction of nodes that need to cooperate for the algorithm to be successful is $\alpha = \frac{n-k}{n}$. Similarly to the privacy game, it is better for a node to play *P* if and only if it believes that at least *αn-1* other nodes will also play *P*. If the belief on the cooperation of one node is π, the belief on the cooperation of *m* nodes among *n-1* other nodes is:

$$\binom{m}{n-1}\pi^{m}(1-\pi)^{n-m-1}. \tag{6}$$

Moreover, for a node, playing *P* is optimum if:

$$\sum_{m=\lceil \alpha n-1 \rceil}^{n-1} \binom{m}{n-1} \pi^{m}(1-\pi)^{n-m-1} \geq \frac{\beta}{\gamma}. \tag{7}$$

Or

$$\sum_{m=0}^{\lfloor \alpha n-1 \rfloor} \binom{m}{n-1} \pi^{m}(1-\pi)^{n-m-1} \leq \frac{\gamma-\beta}{\gamma}. \tag{8}$$

$\lfloor \alpha n - 1 \rfloor$ and $\lceil \alpha n - 1 \rceil$ are the Floor and Ceiling function respectively.

7.4 Evolutionary Game Theoretic Analysis of Trust

In the last Section, we saw that individual rational choices can result in trusting behavior. In this Section, we study how trusting and distrusting behaviors interact. We present in what condition trusting behavior can be the result of a long term evolutionary process. EGT is the application of game theory to evolution. Game theoretic concepts easily translate to evolution because the effectiveness of one animal's behavior depends on the proportion of other animals genetically programmed to use that behavior or other behaviors. The main difference between EGT and game theory is that EGT does not assume the rationality of the players. Evolution is driven by two main mechanisms: random mutation that provides diversity in the population and a selection mechanism that promotes some variety over others. Equivalently, EGT has two main solution concepts: ESS that deals with mutation and the replicator dynamic that examines the selection mechanism.

A strategy x is an ESS [73] if for every strategy $y \neq x$ there exist some $\bar{\varepsilon}_y \in (0,1)$ such that for all $\varepsilon \in \left(0, \bar{\varepsilon}_y\right)$, either

$$u(x, x) > u(y, x) \, \forall y, \tag{9}$$

Or

$$u(x, x) = u(y, x) \text{ and } u(x, y) > u(y, y) \, \forall y \neq x. \tag{10}$$

ESS resists against mutation. Moreover, from (9) and (10), strict Nash equilibrium are ESS and ESS are Nash equilibrium. However, some Nash equilibrium may not be ESS. Thus, ESS requires a strategy not only to be rational (Nash equilibrium) but also to be stable.

The replicator dynamic [74] gives the population dynamics for the population shares of x_i. The replicator dynamic is given by the differential equation:

$$\dot{x}_i = \left[u\left(e^i, x\right) - u(x, x)\right] x_i. \tag{11}$$

$u\left(e^i, x\right)$ is the expected payoff to pure strategy *i* (*i*= *P* or *NP*) at a random match.

$u(x, x)$ is the payoff of an individual drawn at random in the population or the average payoff per individual.

x_i is the proportion of individuals programmed to pure strategy *i* and its time derivative is $\dot{x}_i$.

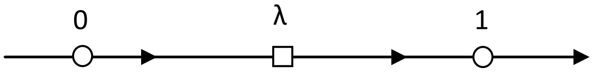

Figure 7.1: Dynamic of the population share of x_P

The replicator dynamic indicates that subpopulations programmed with better than average strategies growth whereas subpopulations programmed with worse than average strategies vanish.

We indicated before that our trust game of Table 7.1 has three Nash equilibria. Two are in pure strategies and one is in a mixed strategy. The mixed strategy Nash equilibrium is neither ESS nor stable in the replicator dynamic. The two pure strategies Nash equilibrium are strict Nash equilibrium and therefore ESS. Moreover, those two strict Nash equilibria are stable in the replicator dynamic. As represented in Fig. 7.1, the *basing of attraction* of the two pure strategies Nash equilibrium intersect at the mixed strategy Nash equilibrium.

Figure 7.1 shows that it is impossible to move from the low trust equilibrium (*NP, NP*) to the high trust equilibrium (*P, P*) through an evolutionary process. The final state depends entirely on the initial condition. Clearly, if starting with a low proportion of nodes playing P ($x_P < \lambda$), the final state will have all nodes playing *NP* ($x_P = 0$). The same way, if starting with a high proportion of nodes playing P ($x_P > \lambda$), the final state will have all nodes playing P ($x_P = 1$). The work in [77] analyzes the dynamic of the *n* players stag hunt game.

7.5 Simulation Results

In all these simulations, we fix $\lambda = \frac{\beta}{\gamma} = \frac{1}{2}$. Thus, in the two player trust game of Table 7.1, the high trust equilibrium (*P, P*) does not risk dominate the low trust equilibrium (*NP, NP*) and vice versa.

The replicator dynamic (11) is implemented in Fig. 7.2 and 7.3. In Fig. 7.2, the initial population is constituted of 51% of agent playing *P* and 49% of agent playing *NP*. Therefore, this initial composition is in the basing of attraction of the high trust equilibrium (*P, P*). We can see that the number of agents playing *P* increases whereas the number of agents playing *NP* decreases. The agents playing *NP* vanish after only twelve generations.

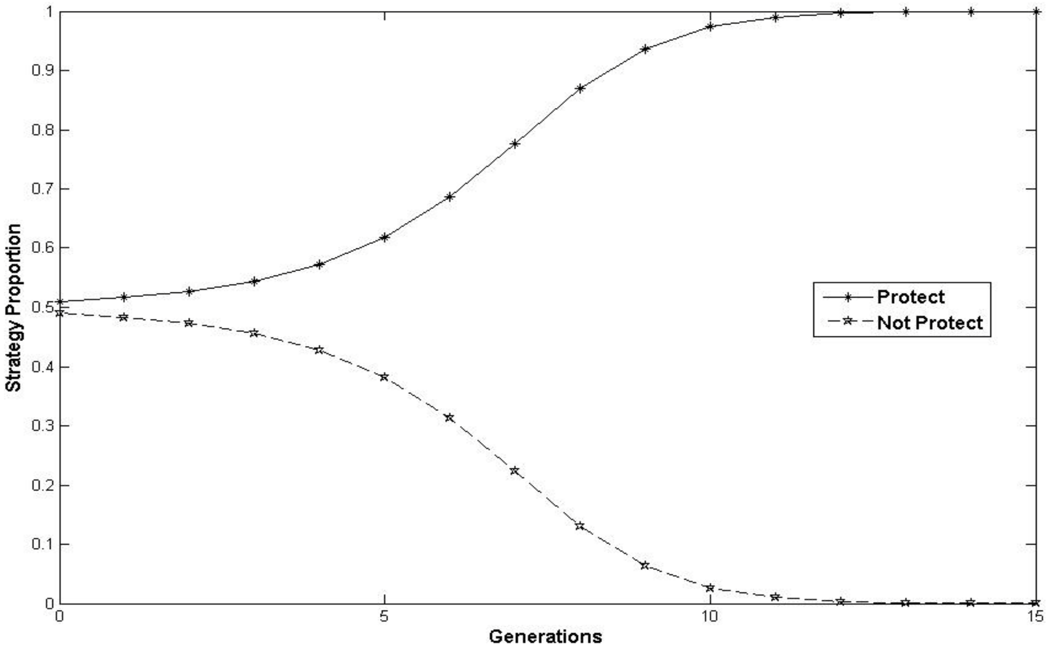

Figure 7.2: Evolution of trust in a two population's model when the initial condition favors the emergence of trust

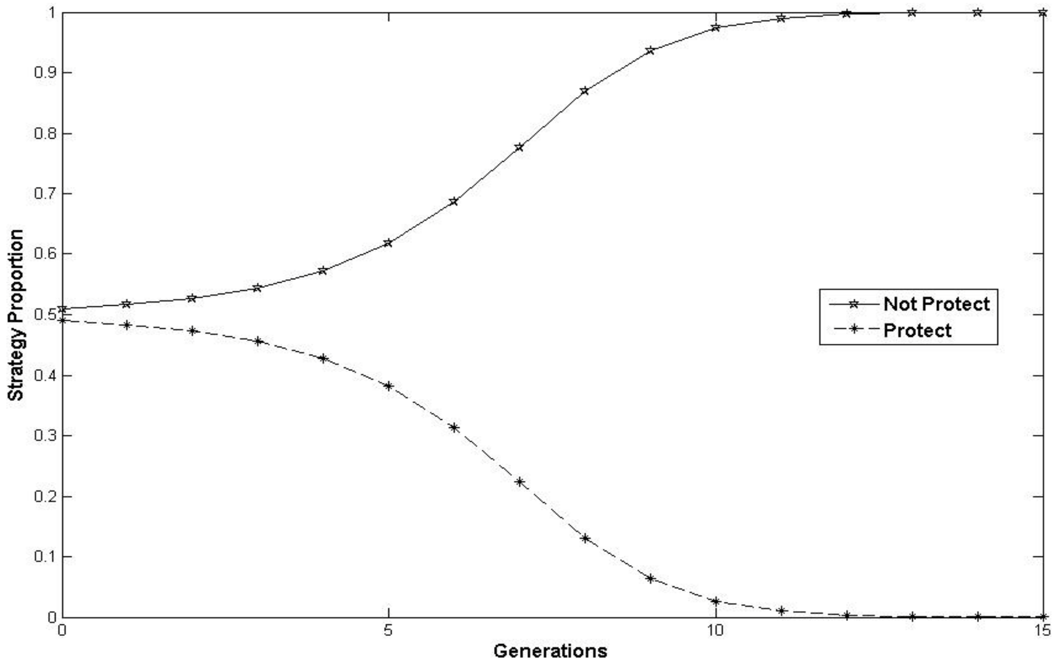

Figure 7.3: Evolution of trust in a two population's model when the initial condition does not favor the emergence of trust

In Fig. 7.3, we reverse the initial proportion of agents playing *P* and *NP*. As a consequence, the initial condition becomes in the basing of attraction of the low trust equilibrium (*NP, NP*). The number of agents playing *P* sinks.

Figure 7.4 represents the minimum trust requirement to protect your privacy as a function of the number of agents sharing the private information (5). From this graph, we can mention that, if a vast number of agents share private information, protection of that private information is cost effective if and only if there is a complete mutual trust among the agents (π close to 1).

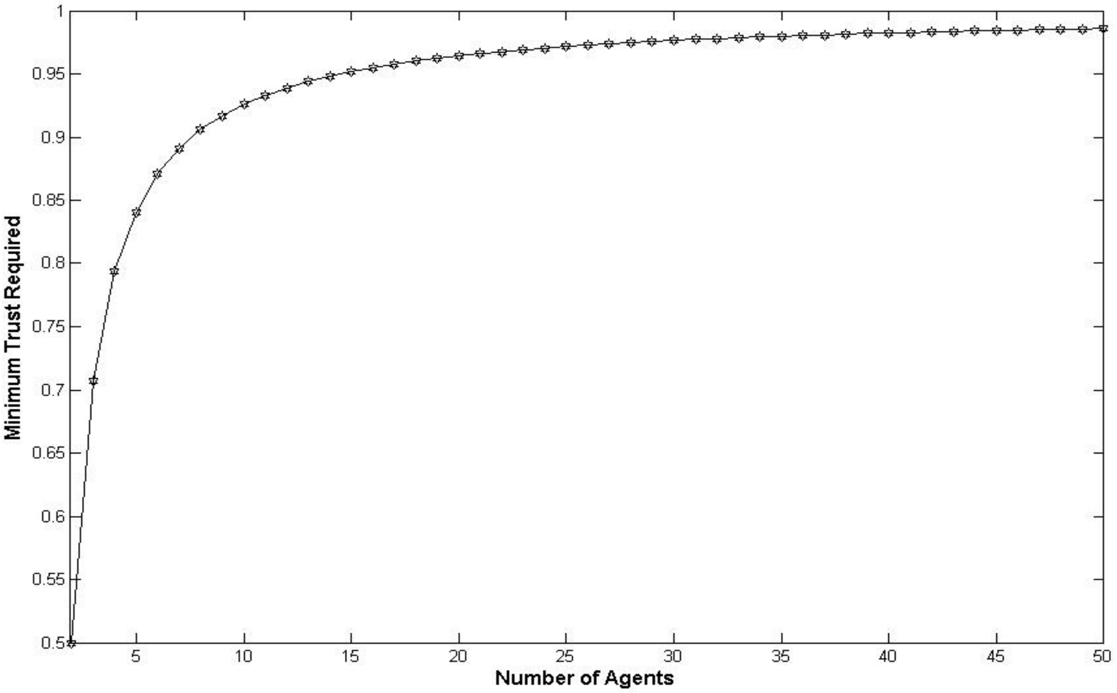

Figure 7.4: Minimum trust requirement to protect your privacy as a function of the number of agents sharing the private information

7.6 Conclusion

In this work, we investigated network security and privacy in autonomous multi-hop network as a game between the nodes in the network. Each node has the freedom to participate in the privacy protection or security mechanism to protect itself or not. We examined the applications that require the cooperation of others to be successful and modeled it as trust dilemma. We first performed an analysis of trust then captured the EGT implication. Both the game theoretic and EGT analysis indicated that mutual trust is necessary to cooperate and reach the efficient equilibrium where all nodes protect the network or protect their privacy. In the replicator dynamic model, the initial condition fully determines the final state (trusting behavior or not).

In the future, we will consider the case of incomplete information when each node payoff from security or privacy is not common knowledge but private information. We will also investigate other game theoretic models of trust in a network.

CHAPTER 8

GAME THEORETIC ANALYSIS OF USERS AND PROVIDERS BEHAVIOR IN NETWORK UNDER SCARCE RESOURCES

The demand on mobile data usage is exponentially increasing since the introduction of iPhones in 2007. The network became congested as millions of users tried to browse website and social networks, send e-mail, stream multimedia, and transfer file simultaneously. An immediate solution for the providers will be to change their pricing strategy with the goal to slow down heavy users and then decrease the bandwidth demand. From the users' standpoint, network providers must constantly upgrade their infrastructure to accommodate new applications and devises. However, upgrading the infrastructure will be costly for the provider. A provider would prefer a minimum investment to upgrade the network while attracting the maximum number of customers. On the other hand, without regular upgrade of the network from the provider, there may be more congestion, more delay and generally a low QoS at the user dissatisfaction. Moreover, users that experience bad connection will be tempted to switch providers. We analyze the dynamic communication market and the users and providers' interaction in the framework of repeated game theory. We consider noise in user's monitoring. We also compare two scenarios: individual and independent action of users as opposed to the collective action of users.

8.1 Introduction

In recent years, we have observed an increase of the number of smart phones, iPhones, PDAs, and other mobile devices. Moreover, the number of users is increasing in countries all over the world. Beside voice communication, those mobile devices support multimedia applications such as videos and TV that consume a tremendous amount of bandwidth. The number of new multimedia applications exponentially increases each year. Those multimedia applications become part of our everyday life. The users are more and more interested in new applications. However, the increase of the number of users and the number of applications coupled with the high bandwidth requirement of those applications tend to saturate the network. All network carriers are striving to keep their network capacity above the data demand. In fact, for the last four years, the demand in data network has been doubling each year. However, we do not observe a similar growth in network capacity. As a consequence, networks become more congested over the years as the demand in bandwidth from the users grows faster than network capacity. Network providers are slow to increase the network capacity because of the cost involved. Clearly, there is a conflict of interest between the network provider's profit and the user's satisfaction. We will model this conflict of interest in the framework of game theory. Recall that game theory is the branch of applied mathematics that models and analyzes strategic interaction among rational decision makers.

One cause of the high increase in bandwidth demand was the pricing mechanism practiced by the provider. Service provider used to practice a flat rate model instead of a usage-based one. As a consequence, for most operators, a very small percentage of users are responsible for the majority of traffic. For instance, in December 2009, the AT&T CEO confirmed that 3 percent of its Smartphone's users generate about 40 percent of its data traffic [78]. This is at the disadvantage of both the majority of users and the providers. The low traffic users in a flat price model subsidize the high traffic users.

Several providers quickly realized the drawback of a flat rate pricing mechanism and adopted a tiered pricing model. As an example, AT&T adopted a tiered pricing model in June 2010. Verizon followed. Sprint, and others will certainly follow soon. This is a strategic move on the providers' part. The benefits to a provider of tiered pricing model are three fold. First, the demand of bandwidth is reduced. In fact, higher price for higher tiers enforce low bandwidth usage. Second, the necessary investment to keep up with a good QoS is reduced and can even be eliminated. Last but not least, the providers can increase their profit.

There is rich literature on game theoretic modeling of user provider interactions. Hassan *et al* [46] show that the user can use a brinkmanship technique to provide credible threats to the provider and therefore compel the provider to allocate more resources to users. Sengupta *et al* [47] investigate a market in which multiple service providers compete to get a large portion of the spectrum and sell it to a maximum number of users. Other works analyzing provider price competition to attract users include [48-50]. The work in [51] examines one provider's Nash equilibrium price under asymmetric information. A comprehensive survey of wireless service providers and user's interactions can be found in [52]. An application of game theory to packet forwarding is found in [1-3].

This Chapter considers one provider and multiple users. User/provider interaction will be modeled as a repeated game. The threat available to a user is based solely on repeated interactions. The prospects of future loss of customers will oblige the providers to invest in the network. A customer will monitor the channel condition and QoS and leave a service provider that does not invest in the network to provide a good QoS. We will also consider the impact of noise on monitoring. It is possible that a service provider invests to improve the QoS while a customer still experience congestion and delay due to the non-reliability of the wireless channel.

We will consider two scenarios in our analysis. The first scenario deals with independent users that observe the provider's action (act) separately. The second case is about the users who observe and react collectively. We will show that collective monitoring is better in the sense that it increases the monitoring accuracy. Thus, a more efficient equilibrium is reached under collective monitoring. Moreover, acting collectively increases the bargaining power of users and can force the provider to make substantial investments and increase the supply of bandwidth. Certainly, the best solution to mobile data explosion should not be solely based on price increase, to decrease the demand on bandwidth, but also on adequate investment to increase the bandwidth supply. This research focuses on increasing the bandwidth supply.

The remainder of this Chapter is organized as follows. Section 8.2 is about the stage game model between a user and a provider. Section 8.3 analyzes repeated interactions under perfect monitoring. Section 8.4 considers noise in monitoring. Section 8.5 extends the one user model to *n* independent users. Section 8.6 analyzes the collective action of the *n* users. Section 8.7 presents the numerical results and Section 8.8 concludes the Chapter.

8.2 Stage Game Model

We consider a service provider with several users. The service provider owns the network infrastructure. The service provider is responsible for the network administration and maintenance. We assume that the demand in bandwidth from the users' devices and applications increases with time. Then, the users experience more congestion and delay if the provider does not increase the network capacity. Therefore, the service provider should periodically invest in the network infrastructure to accommodate the extra network traffic demand and maintain the same QoS. The new investment in network infrastructure can be in the form of new cell towers, change from cable to fiber optic to increase the capacity in the backbone of the network, or simply change in modulation techniques or software upgrades.

We assume that time is divided in periods. The length of a period is *T*. A period *T* can be a month to correspond with the billing cycle. At the end of each period, the service provider has two strategies or choices: Invest in the network (*I*) or cooperate or Not to Invest (*NI*) or defect. On the other hand, at the end of each period, a user can either Keep the Provider (*KP*) or cooperate if satisfied with the QoS or Change the Provider (*CP*) or defect if not satisfied. We consider that a user is free to change the provider at the end of each period if not satisfied. This means that there is no contract that prevents the user from switching the provider. We assume that there is no monopoly or there are several service providers that the users can choose from.

TABLE 8.1 STAGE GAME IN STRATEGIC FORM

		Provider	
		I	*NI*
User	*KP*	*A*, *a*	*C*, *b*
	CP	*B*, *c*	*D*, *d*

For the sake of simplicity, we start by presenting a two players' game between one user and a service provider. The n user's case will be the subject of Section 8.5. In this Section, we assume perfect monitoring and will consider imperfect monitoring in Section 8.4. Therefore, we presuppose that a user experiences a good QoS if the provider has invested and experiences a bad QoS otherwise. The user and the provider decide simultaneously. The strategic form of the user/provider game is represented in Table 8.1.

Let us examine the payoff structure of the game. The payoff or utility function characterizes the players' satisfaction given a profile of action. For the user, we need to establish an ordinal relation between the payoff A, B, C and D. The user prefers the provider to invest than not to invest, thus $A, B \geq C, D$. Also, if the provider chooses to invest (*I*), the user has a good QoS and prefers *KP* to *CP*. In other words, we have $A > B$. However, if the provider chooses *NI*, the user will prefer *CP* to *KP*. This means that $D > C$. Finally, we have for the user: $A > B \geq D > C$.

Without loss of generality, we assign a quantitative value to the user payoffs. We consider that when a user changes the provider, he is indifferent to the past provider's behavior and did not incur any gain or loss from that provider's action. Then, we have $B = D = 0$. Let f be the periodic fee the user pays to the provider. Let v be the value to the user of the service received from the provider. The two possible values of v will be $\overline{v}$ if the user receives a good QoS and $\underline{v}$ if the QoS is bad. We assume that $\overline{v} > f > \underline{v}$. Thus, we have: $A = \overline{v} - f > 0$ and $C = \underline{v} - f < 0$. When we normalize the payoffs to that of mutual cooperation (*KP,I*), we have $1 = A = \overline{v} - f > 0 > \frac{\underline{v}-f}{\overline{v}-f} = \frac{C}{\overline{v}-f} = -l$. Here, l represents the loss due to bad service.

We now analyze the provider payoff. If the user cooperates or keeps the provider (*KP*), the provider prefers not to invest (*NI*) to invest (*I*). This is obvious since investing is costly. Then, we have $b > a$. Similarly, if the user plays CP, the provider again prefers *NI* to *I*. Then, we have $d > c$. As we can see, for one-shot games, the provider always prefers *NI* to *I* regardless of what the user does. Therefore, cooperation (*I*) is dominated by defection (*NI*). We assume that the market conditions are such that the service provider prefers to invest in the network and keeps its user than not to invest and loose them. Thus, we must have $a > d$. Finally, we have for the provider: $b > a > d > c$.

As for the user, let us find the quantitative values of the provider's payoff. Let *e* be the average expense required to accommodate a user with a good QoS and *f* be the periodic fee the user pays to the provider. We assume that $f > e > 0$. Thus, we have: $f = b > a = f - e > 0 = d > c = -e$. When we normalize the payoffs to that of mutual cooperation (*KP, I*), we have: $g > 1 > 0 > 1 - g$ with $g = \frac{f}{f-e}$. Table 8.2 shows the strategic form of the game with normalized payoff. We can see that Table 8.2 represents an asymmetric game.

TABLE 8.2: STAGE GAME WITH NORMALIZED PAYOFFS

		Provider	
		I	*NI*
User	*KP*	*1, 1*	*-l, g*
	CP	*0, 1-g*	*0, 0*

This stage game has a unique Nash equilibrium. Mutual defection (*CP, NI*) is the Nash equilibrium. At the Nash equilibrium, the provider will not invest and the user will change provider. At a Nash equilibrium profile, no player can increase its payoff by a unilateral deviation. Moreover, each player plays a best response to the behavior of other players. However, the Nash equilibrium in this game is not Pareto efficient. Recall that an allocation is Pareto efficient if there is no other allocation that can make at least one player better off without making any other player worse off. The Pareto efficient outcome is mutual cooperation (*KP, I*) where provider invests and the user stays. We show in the next Section that the provider and the user can reach the Pareto efficient outcome when the interactions are repeated.

8.3 Repeated Game Model

There are two players: a user and a provider. The stage game in Table 8.2 is repeated over time. The provider discounts future payoff by a common discount factor δ and we have $0 \leq \delta < 1$. Player i ($i \in \{\text{User}, \text{Provider}\}$) sequence of stage game payoffs is $(u_i^t)_{t=1}^{\infty}$. The provider is a long-lived player [60] and wants to maximize the expected

δ-discounted average of its sequence of payoffs. Therefore, the provider's repeated game payoff is given by:

$$(1-\delta)\sum_{t=1}^{\infty}\delta^{t-1}u_i^t. \qquad (1)$$

The users are short-lived players and are concerned only with the payoff in the current period. This means that the user plays the myopic best reply to the provider's action. We consider the following strategy for the user: Start by cooperating and continue to cooperate until you observe a defection, defect forever after a defection. This simple strategy is called Grim Trigger strategy. The strategy is grim because it does not forgive. Also, it needs a trigger (a defection) to defect forever.

The main concept used to analyze a repeated game under perfect monitoring is the SPE [60]. A strategy profile is a SPE of the repeated game if it represents a Nash equilibrium after every history or subgame of the repeated game. SPE is a stronger criterion than the Nash equilibrium. SPE eliminates Nash equilibrium in which players' threats are not credible.

Theorem 1**:** The strategy profile where both the user and the provider use grim trigger is SPE if and only if:

$$\delta \geq \frac{g-1}{g}. \qquad (2)$$

Proof: A strategy profile is SPE if and only if there are no profitable one-shot deviations [60]. One –shot deviation of the provider is not profitable if:

$$1 \geq (1-\delta)\left[g + \sum_{t=1}^{\infty} \delta^{t-1} 0\right] = (1-\delta)g$$

$$\Rightarrow \delta \geq \frac{g-1}{g}.$$

Moreover, when the provider cooperates, the best response of the user is to cooperate. Also, when the provider defects, the best response of the user is to defect ■

8.4 Imperfect Monitoring Game

Up to now, we did not consider noise when a player monitors the opponent behavior. In fact, it is not realistic to assume that a user can perfectly monitor the provider behavior. Even if the provider invests in the in the network infrastructure, a user can still experience a low QoS due to the non-reliability of the wireless channel. Random events such as weather conditions can impose performance degradation on the wireless network. On the other hand, it is reasonable to assume that a provider knows for sure when a user decides to stay or to leave. Therefore, the user's action will be public while the provider's action will be private. However, there is a public signal $y_2 \in \{\overline{y}, \underline{y}\}$ indicating the past action of the provider. Thus, we have a game of imperfect public monitoring.

Let $a_1 \in \{KP, CP\}$ and $a_2 \in \{I, NI\}$ be the user and provider's action respectively. Also, ε denotes the error in user's monitoring. The distribution of the public signal y_2 is given by:

$$m_2(\overline{y}|a) = \begin{cases} 1-\varepsilon, & \text{if } a_2 = I \\ \varepsilon, & \text{if } a_2 = NI \end{cases} \tag{3}$$

We assume that $0 < \varepsilon < 1 - \varepsilon < 1$. This ensures that it is more likely that the user observes the high quality signal $\overline{y}$ when the provider plays *I* and observes the low quality signal $\underline{y}$ when the provider plays *NI*. As before, we consider that the user plays *KP* when observing the good signal $\overline{y}$ and plays *CP* forever when observing the bad signal $\underline{y}$.

Let $V_{a_1,a_2}(\varepsilon, \delta)$ represents the repeated game payoff of the provider given the action profile (a_1, a_2). We have the following Bellman equation [79],

$$V_{KP,I} = (1 - \delta) + \delta\big[(1 - \varepsilon)V_{KP,I} + \varepsilon V_{CP,NI}\big], \tag{4}$$

$$V_{KP,NI} = (1 - \delta)g + \delta\big[\varepsilon V_{KP,I} + (1 - \varepsilon)V_{CP,NI}\big]. \tag{5}$$

Equation (4) and (5) represent the repeated game payoff as a function of the initial payoff and future payoff. The future payoffs depend on the signal observed and take noise into account with the possibility of good and bad observations.

In any period, the provider prefers cooperation to defection if:

$$V_{KP,I} \geq V_{KP,NI} \Rightarrow$$

$$(1 - \delta) + \delta\big[(1 - \varepsilon)V_{KP,I} + \varepsilon V_{CP,NI}\big] \geq (1 - \delta)g + \delta\big[\varepsilon V_{KP,I} + (1 - \varepsilon)V_{CP,NI}\big] \Rightarrow$$

$$V_{KP,I} \geq V_{CP,NI} + \frac{(1 - \delta)(g - 1)}{\delta(1 - 2\varepsilon)}. \tag{6}$$

Equation (6) shows that the provider is willing to cooperate if the payoff of mutual cooperation $V_{KP,I}$ exceeds that of mutual defection $V_{CP,NI}$ by at least $\frac{(1-\delta)(g-1)}{\delta(1-2\varepsilon)}$. This difference in payoff vanishes as noise decreases and the discount factor becomes close to

1. As a consequence, the provider will invest in the network at any given period under two conditions. First, the provider must be sufficiently patient. Future gain will have almost the same value as current payoff. Second, the provider must be convinced that the users are able to accurately monitor its behavior. For instance, the provider will find its investment in the network useless if that investment does not translate into a better QoS to a vast majority of users.

Let us evaluate the maximum value of $V_{KP,I}$. From (4), the maximum value of $V_{KP,I}$ is achieved when $V_{CP,NI}$is maximum. Then, taking (6) with equality, we have:

$$V_{CP,NI} = V_{KP,I} - \frac{(1-\delta)(g-1)}{\delta(1-2\varepsilon)}. \tag{7}$$

Replacing (7) into (4) gives us:

$$V_{KP,I_{Max}} = 1 - \frac{\varepsilon(g-1)}{(1-2\varepsilon)}. \tag{8}$$

A necessary condition for the provider to cooperate is to have $V_{KP,I_{Max}} \geq 0$. This means that:

$$\varepsilon \leq \frac{1}{1+g}. \tag{9}$$

Equation (9) indicates that monitoring must be more precise when the temptation to defect or *g* increases.

8.5 Extention to *N* Independent Users

We consider a service provider with n users in a dynamic market. Each user independently monitors the channel condition to infer the provider's action. A user leaves the provider forever after experiencing a bad QoS or observing a low quality signal $\underline{y}$. A user stays with the provider otherwise. Let *x* denotes the total investment required from the provider to accommodate the *n* users and *f* the periodic fee paid by each user.

In a given period, when the provider chooses to invest in the network, the expected revenue of the provider in the next period will be:

$$E\left\{\sum_{k=0}^{n}\binom{k}{n}(1-\varepsilon)^{k}\varepsilon^{n-k}kf\right\}=(1-\varepsilon)nf. \tag{10}$$

k is the number of users observing a good signal. $0 \leq k \leq n.$

Equation (10) takes into account the noise level and all combination of subsets of users that may experience good or bad signal. The k users that observe the good signal will stay with the provider and pay the fee *f* in the next period. On the other hand, *n-k* users will quit and not pay any fee. This translates into a binomial distribution and the corresponding expected value is represented in the right hand side of (10).

Similarly, when the provider chooses not to invest in the network, its expected revenue in the next period will be:

$$E\left\{\sum_{k=0}^{n}\binom{k}{n}\varepsilon^{k}(1-\varepsilon)^{n-k}kf\right\}=\varepsilon nf. \tag{11}$$

Therefore, the provider's cooperation can be enforced if:

$$-x + \delta(1 - \varepsilon)nf \geq \delta\varepsilon nf \qquad (12)$$

$$\Rightarrow \delta \geq \frac{x}{(1 - 2\varepsilon)nf}. \qquad (13)$$

If the total investment x is proportional to the number of users and e is the average investment per user. Then (13) becomes:

$$\delta \geq \frac{ne}{(1 - 2\varepsilon)nf} = \frac{e}{(1 - 2\varepsilon)f}.$$

Consequently, when noise in monitoring vanishes and the revenue per users *f* exceeds the required investment per user *e*, there exists $\underline{\delta}, 0 < \underline{\delta} < 1$ such that the provider is better off cooperating when its discount factor is $\delta \geq \underline{\delta}$.

Note that the number of users can fluctuate from one period to the next due to the acquisition or loss of customers. Thus, we did not include the total sequence of payoffs. However, this analysis captures the essence of the game and players' interaction.

8.6 Collective Action of the *N* Users

In the last Section, each user independently monitors the channel condition to infer the provider action. Moreover, the users decide to stay or to quit the provider independently. In this Section, we investigate what should happen if the users coordinate their actions.

We consider a tamperproof resistant database that helps to coordinate the users' action and observation. At the end of each period, each user sends its observation to the database.

There are only two possible observations a user can report depending on the QoS: a good observation for a good QoS and a bad observation otherwise. The users agree to collectively leave the provider if the ratio of users experiencing a good QoS in a given period is below a pre-established cutoff score. Let us call that cutoff score α, $0 \leq \alpha \leq 1$.

Similar reasoning as in the last Section indicates that the provider cooperates when

$$-x + \delta n f E\left\{\sum_{k=\alpha n}^{n} \binom{k}{n} (1-\varepsilon)^{k} \varepsilon^{n-k}\right\} \geq \delta n f E\left\{\sum_{k=\alpha n}^{n} \binom{k}{n} \varepsilon^{k} (1-\varepsilon)^{n-k}\right\}. \quad (14)$$

The difference here compared to the last Section is that the provider will receive the full revenue *nf* or zero depending on if *αn* users report a good signal or not. Assuming that the provider has a large number of users, which is generally the case, the Binomial distribution can be approximated by a Normal distribution. Then, (14) becomes:

$$-x + \delta n f\left\{1 - \Phi\left(\frac{\alpha n - n(1-\varepsilon)}{\sqrt{n\varepsilon(1-\varepsilon)}}\right)\right\} \geq \delta n f\left\{1 - \Phi\left(\frac{\alpha n - n\varepsilon}{\sqrt{n\varepsilon(1-\varepsilon)}}\right)\right\}$$

$$\Rightarrow \delta \geq \frac{x}{\left\{\Phi\left(\frac{\alpha n - n\varepsilon}{\sqrt{n\varepsilon(1-\varepsilon)}}\right) - \Phi\left(\frac{\alpha n - n(1-\varepsilon)}{\sqrt{n\varepsilon(1-\varepsilon)}}\right)\right\} n f}. \quad (15)$$

Φ represents the cumulative distribution function of the Normal distribution.

Generally, a solution that enforces cooperation with the lowest discount factor is preferable. The discount factor measures the patience of the provider. Also, considering the interest rate per period *r*, the discount factor will be $\delta = \frac{1}{1+r}$. Thus, a high interest rate per period implies a low discount factor. Therefore, in a competitive wireless communication market, we want even the least patient provider or those with a high interest rate per period to invest in the network for the collective benefit of its users. To achieve that, we need to maximize the denominator of (15).

A straightforward analysis shows that, for $\varepsilon \neq 0$ the difference

$$\left\{\Phi\left(\frac{\alpha n - n\varepsilon}{\sqrt{n\varepsilon(1-\varepsilon)}}\right) - \Phi\left(\frac{\alpha n - n(1-\varepsilon)}{\sqrt{n\varepsilon(1-\varepsilon)}}\right)\right\}$$

is maximized when $\alpha = 0.5$.

The message is that when α is too big, the provider will find it risky to invest. In fact, few users may report a bad QoS due to noise and cause the provider to lose its entire customer base when the users act collectively. On the other hand, if the ratio α is too small, the provider may not invest and expect few users to still report a good QoS. Recall that there is noise in users' observation. As a result, a fair value of one half is adequate.

When we take $\alpha = 0.5$, we have:

$$\Phi\left(\frac{\alpha n - n\varepsilon}{\sqrt{n\varepsilon(1-\varepsilon)}}\right) - \Phi\left(\frac{\alpha n - n(1-\varepsilon)}{\sqrt{n\varepsilon(1-\varepsilon)}}\right)$$

$$= \Phi\left(\frac{n(1-2\varepsilon)}{2\sqrt{n\varepsilon(1-\varepsilon)}}\right) - \Phi\left(-\frac{n(1-2\varepsilon)}{2\sqrt{n\varepsilon(1-\varepsilon)}}\right)$$

$$= \Phi\left(\frac{n(1-2\varepsilon)}{2\sqrt{n\varepsilon(1-\varepsilon)}}\right) - \left\{1 - \Phi\left(\frac{n(1-2\varepsilon)}{2\sqrt{n\varepsilon(1-\varepsilon)}}\right)\right\}$$

$$= 2\Phi\left(\frac{n(1-2\varepsilon)}{2\sqrt{n\varepsilon(1-\varepsilon)}}\right) - 1.$$

Therefore, when we take the average investment per user, (15) becomes:

$$\delta \geq \frac{e}{\left\{2\Phi\left(\frac{n(1-2\varepsilon)}{2\sqrt{n\varepsilon(1-\varepsilon)}}\right)-1\right\}f}. \tag{16}$$

8.7 Numerical Results

In this Section, we compare the provider's behavior when the users act independently to when the users act collectively. We consider a service provider with a large number of customers; say 10,000 users in a local area, a ZIP code for instance. Users are divided in subsets. Subsets of users belong to the same area with similar channel conditions. We also assume that the provider uses 20% of its revenue in administrative and operational cost. Thus,

$$\frac{e}{f} = 0.8.$$

Figure 8.1 compares the minimum discount factor that enforces cooperation in two scenarios: individual and independent decision and collective decision. In the case of independent user, the minimum discount factor required for the provider cooperation equals 1 when the noise level ε reaches 0.1. Therefore, it becomes impossible for the users to enforce the provider cooperation with a probability of observation error above 0.1. The result is a mutual defection which is not an efficient outcome. The incentive to cooperate is lost due to noise.

However, when the users observe and decide collectively, the provider still has an incentive to cooperate with a noise level of 0.49. This is clearly in the advantage of both the provider and the users. Introducing a database that records all users' observations has, as a consequence, the increase of the monitoring accuracy. Moreover, the database gives more bargaining power to users. The n+1 player's game becomes strategically equivalent to a two player game. The provider is one of the players and the *n* users represent the second player.

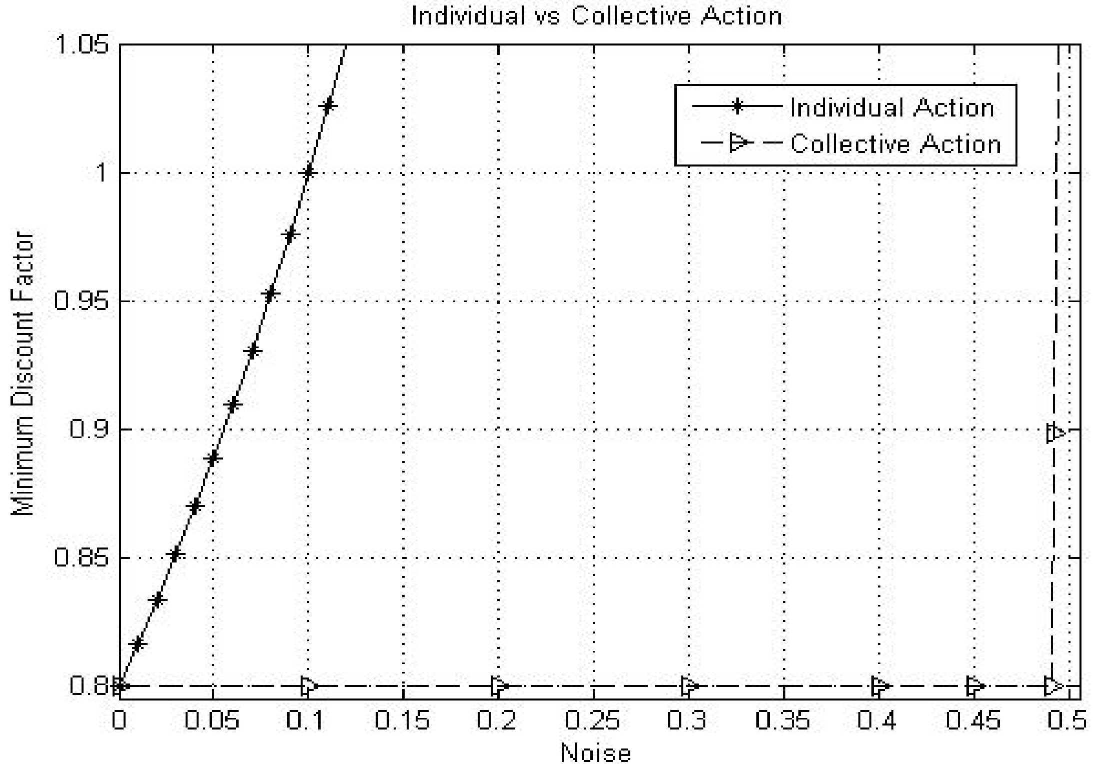

Figure 8.1: Comparison between individual and collective actions of users. α=0.5

Figure 8.2 shows what will happen if the users are intransigent and request a cutoff score of 95%. The provider will cooperate under perfect monitoring or when the noise level is below 5%. Therefore, a fair cutoff score of 50% is better that the 95%. This is because the provider's investment becomes unprofitable under noise.

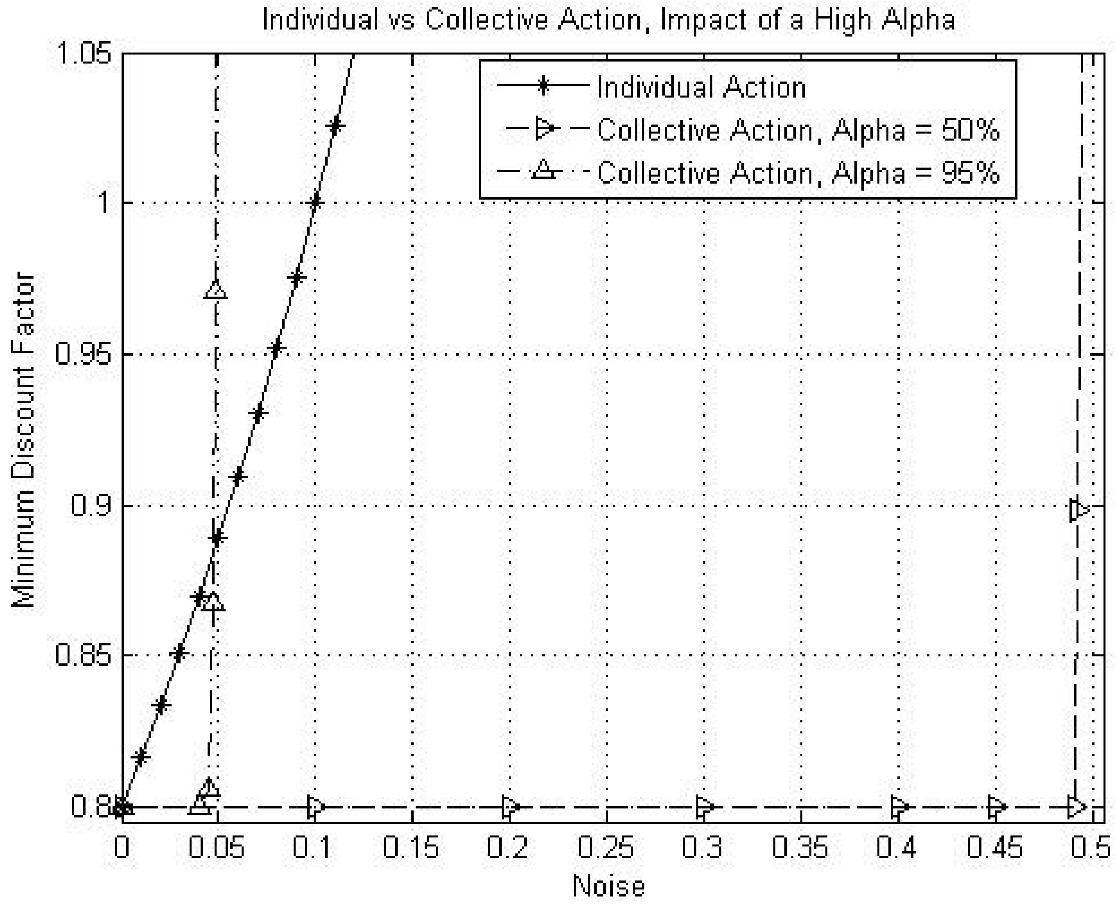

Figure 8.2: Comparison of high and fair values of Alpha

Figure 8.3 analyzes the provider's action under noise and several other cutoff scores. We can see that the provider's behavior is symmetric on a cut off score of 0.5. Therefore, recommending that 75% of users report a good QoS is identical to a request of 25%.

A straightforward statistical analysis indicates that, for a 3 standard deviation confidence, we must have:

$$n\varepsilon + 3\sqrt{n\varepsilon(1-\varepsilon)} \leq \alpha n \leq n(1-\varepsilon) - 3\sqrt{n\varepsilon(1-\varepsilon)} \Rightarrow$$

$$\varepsilon + 3\sqrt{\frac{\varepsilon(1-\varepsilon)}{n}} \leq \alpha \leq (1-\varepsilon) - 3\sqrt{\frac{\varepsilon(1-\varepsilon)}{n}}. \qquad (17)$$

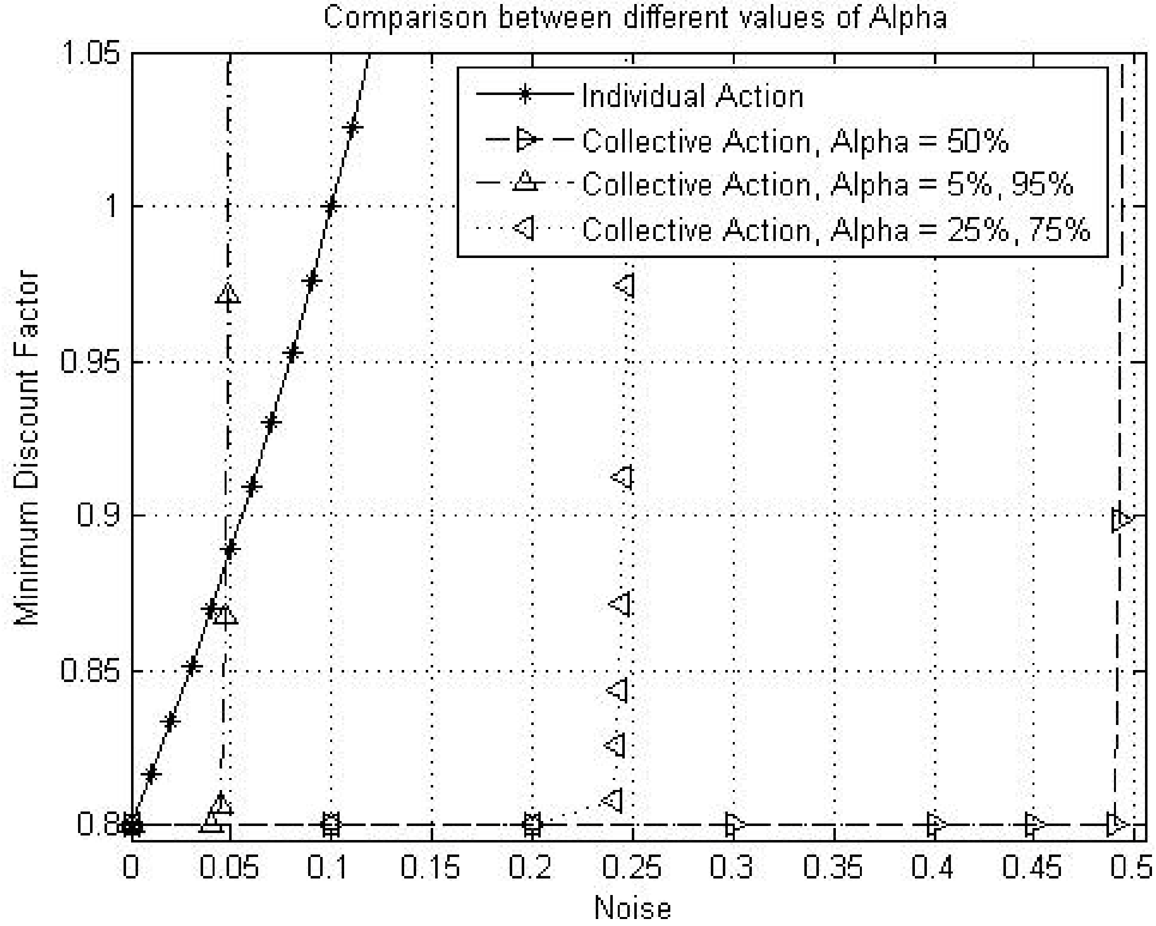

Figure 8.3: Comparison between different values of Alpha

8.8 Conclusion

This analysis shows that recording the provider's behavior in a database can have a valuable effect for users. The provider is forced to cooperate even with a high observation error (close to 0.5!). Collective observation increases the accuracy in monitoring and the bargaining power of users. Previous researches [47-50] have emphasized the competition between service providers to guarantee market efficiency. This Chapter focuses on the power of repeated interactions and recording the provider's action.

Similar results can be obtained if the user does not use the grim trigger strategy presented here, but instead decides based on the provider's reputation. The provider's reputation will depend on the history of percentage of user satisfaction. The mechanism described here is easy to implement. An application on a Smartphone can automatically monitor the QoS, report it at the end of each period to the database, read other users' observations, and recommend to switch the provider or not, depending on its past behavior or reputation.

We believe that this recording mechanism combined with the change of pricing strategy from flat rate to tiered pricing should avoid network congestion. The tiered pricing should decrease the demand in data traffic, whereas recording the provider's behavior should increase the network capacity.

All along this Chapter, we made the implicit assumption that the provider uses flat rate pricing. In the future, we will consider different pricing strategies.

CHAPTER 9

CONCLUSION AND FUTURE WORKS

This Chapter concludes this dissertation. Our contributions are recapitulated and the directions for future research are proposed.

9.1 Conclusion

This dissertation addresses several concerns that are universal in social science. Those enigmas include rationality, cooperation, selfishness, trust, security, privacy, as well as independent and collective decision. Game theory and EGT have been used in this work to propose novel solutions to those problems with rigorous mathematical analyses. Those dilemmas will become more consequential as nodes in a network have more freedom, autonomy, independence, intelligence, and learning capability.

Repeated interaction has been used to model different scenarios in wireless networks including packet forwarding at the routing layers, network security, user and service provider interaction. When dealing with packet forwarding, we have presented the advantage of repeated interaction over reputation mechanisms and virtual currency systems. We have considered both rational and irrational nodes as well as perfect and imperfect monitoring.

With respect to security mechanisms in our analysis, the trust in a central authority, like a certificate authority, has been replaced with trust among nodes. Nodes need to trust each other to secure the network.

The major contributions of this dissertation are summarized as follow:

1. Mitigation of selfish behavior in autonomous ad hoc networks considering both rational and irrational nodes.

2. Use of repeated, stochastic, and evolutionary games to design distributed cooperation algorithm.

3. Use of community enforcement mechanisms to compel node cooperation under non-uniform traffic load distribution.

4. Analysis of node cooperation in ad hoc network under noise and imperfect private monitoring.

5. Investigate network security and its connection to trust in wireless networks.

6. Interpret users' and service providers' optimal behavior alongside an exponential increase in bandwidth demand.

To place our work in a wider context, it is worth mentioning that the strategies developed from Chapter 3 to Chapter 6 can also apply to file sharing in peer-to-peer networks and to inter-domain routing.

9.2 Future Works

The first part of this PhD research mainly involves enforcing cooperation to forward packets at the routing layer. A comprehensive model for an autonomous network can then be developed. Clearly, cross layer optimization techniques in the framework of game theory and EGT can be the subject of future research. The incorporation of information from other layers in the models developed in this work would be an improvement. For instance, MAC and transport layer information can be aggregated to consider network congestion, packet loss, and packet retransmission. Also, our model extension without a database in Section 5.3 needs further investigation. It is still not clear how the network topology will influence the game. Moreover, a general framework for packet forwarding games should have both imperfect private monitoring and private payoff information. However, as mentioned in Section 5.5, even though belief-free equilibrium strategies are robust to imperfect private monitoring, they may not resist private payoff information.

Additionally, introducing incomplete information to the network security game introduced in Chapter 7 can be the subject of future research. In fact, each node payoff, from security or privacy, is not common knowledge but private information.

Finally, in Chapter 8, the users switch to a new service provider if the current service provider does not invest sufficient resources to improve the quality of service for the users. However, the behavior dynamic of other service providers has not yet been considered. Some users may leave a provider while new users join that provider. A complete market analysis can be done to investigate the optimum user and service provider behavior. It is well known that the market model proposes a simple and almost optimum solution to complex resource allocation problems for large scales. Market interaction consists of several agents with limited knowledge of the system involved in decentralized interaction. Each agent makes its decision in a distributed way. The market's interactions converge to equilibrium that is, for the most part, close to the optimum solution. Moreover, there is a strategic effect in market interaction. The behavior of any agent affects the payoff and the behavior of others. Therefore, market interaction can be modeled as a game. Furthermore, compared to centralized resource allocation schemes, distributed mechanisms based on market principles are robust against strategic manipulations from selfish users.

APPENDIX

GAME THEORY AND EVOLUTIONARY GAME THEORY OVERVIEW

According to Myerson [35], game theory can be defined as the study of mathematical models of conflict and cooperation between intelligent rational decision-makers. A rational player makes decisions to satisfy his or her self interest. A game can be represented in different forms. Most of the time, it is represented in extensive form or in strategic (or normal) form. The extensive form of a game is used to formalize sequential action of players. In those games, the order in which players act in the game is important. On the other hand, a strategic form of a game is formalized as:

$$\Gamma = (N, (S_i)_{i \in N}, (u_i)_{i \in N}).$$

N is the set of players of game *Γ*, i is a player in *N*, S_i is the set of pure strategies that players i can choose from, u_i is the utility function of player *i*. A mixed strategy is a random combination of two or more pure strategies. A strategy profile is a combination of strategies that the players can choose. The set of all possible strategy profiles is $S = X_{j \in N} S_j$. u_i is a function defined from the set of strategy profiles S to the set of real numbers R. At any strategy profile, the utility function associates the expected utility payoff that player i would get. A game in strategic form can also be represented by a matrix as in table A.1.

For two strategies A and B, A strictly dominates B if A always earns a higher payoff than B. A weakly dominates B if A never earns a lower payoff than B and A is superior to B for at least one strategy.

A. 1. Nash Equilibrium

A strategy profile is a Nash equilibrium if and only if no player can gain by changing its strategy when the other players do not change. Moreover, in a Nash equilibrium, each player's equilibrium strategy is a best-response to other player's equilibrium strategies.

Definition 1: A mixed strategy profile σ* is a Nash Equilibrium if, for all players i

$$u_i(\sigma_i^*, \sigma_{-i}^*) \geq u_i(s_i, \sigma_{-i}^*) \text{ for all } s_i \in W_i \qquad (1)$$

We have a strict Nash equilibrium if inequality (1) is strict.

Theorem 1: Given any finite game Γ in strategic form, there exists at least one Nash equilibrium.

Proposition 1: The Nash equilibrium is not always unique.

Theorem 2: A strictly dominated strategy is never a Nash equilibrium.

This comes from the fact that a strictly dominated strategy can never be optimal.

An allocation is Pareto efficient if there is no other allocation that can make at least one individual better off without making any other individual worse off.

Proposition 2: A Nash equilibrium is not always Pareto efficient.

We will illustrate proposition 2 in the following Subsection.

A.2. Prisoners’ Dilemma Game

Table A.1 shows the Prisoners’ Dilemma game in strategic form. In this game, if both players cooperate, they get the reward price R. They get the punishment price P if both defect. But if one defects when the other cooperates, the defector gets the temptations price T while the cooperator gets the suck price S. We also have: T>R>P>S. For the one stage game, player can choose to cooperate or to defect. Cooperate is strictly dominated by defect. The only Nash equilibrium is both players defect. The 2 players get the punishment prize P. However, if the 2 players choose to cooperate, the 2 players will get the Pareto efficient outcome R. We can see that when rational intelligent players play the game, they will get a non efficient outcome. This illustrates that the Nash equilibrium is not always Pareto efficient. We see in the next Section that when the game is played repeatedly, cooperation among players can emerge. Players could cooperate expecting future gain and reach an efficient equilibrium while all players cooperate.

TABLE A.1: PRISONNERS’ DILLEMA GAME IN STRATEGIC FORM

		Player 2	
		Cooperate	Defect
Player 1	Cooperate	R,R	S,T
	Defect	T,S	P,P

A.3. Repeated Game

A repeated game is some number of repetitions of a base game called the stage game. A repeated game is of perfect monitoring if at the end of each period, each player observes the strategies chosen by other players. A history of the repeated game at period *t* h^t is the list of all players' actions in all periods before *t.* A pure strategy for a player in the repeated game is a mapping from all possible histories to a stage game strategy. The number of repetitions can be finite or infinite. When the end of the game is fixed and known to all players, a backward induction argument shows that it is optimum to play a Nash equilibrium strategy of the stage game in all periods even though that may not be Pareto efficient. For the infinitely repeated game, the payoff of player i is the infinite sequence of payoffs

$(u_i^0 + u_i^1 + u_i^2 + u_i^3 \dots)$.

The average δ-discounted payoff of that sequence is:

$$(1-\delta)\sum_{t=0}^{\infty}\delta^t\, u_i^t. \tag{2}$$

δ is the discount factor. $0 \le \delta < 1$.

The payoff of a pure strategy profile σ is defined as:

$$U_i(\sigma) = (1-\delta)\sum_{t=0}^{\infty}\delta^t\, u_i\left(a^t(\sigma)\right). \tag{3}$$

$a^t(\sigma)$ is the pure action profile in period *t.*

Definition 2: A strategy profile is a subgame perfect equilibrium if it represents a Nash equilibrium of every subgame of the original game.

Proposition 3: A strategy profile σ is a subgame perfect Nash equilibrium if and only if there is no profitable one – shot deviation.

This proposition is the one – shot deviation principle.

SPE is a stronger criterion than the Nash equilibrium in repeated game. SPE imposes the behavior to be sequentially rational. This means that the behavior must be optimum even out of the Nash equilibrium path.

The minmax payoff of a player is a payoff that minimizes his maximum possible loss.

Theorem 3: For every pure action profile whose payoff strictly dominates the pure action minmax payoff, there exists a SPE of the repeated game in which that action profile is played in every period if the players are sufficiently patient.

This theorem is called the folk theorem. Player's patience means that the discount factor is large. It is one of the most important results in repeated game. It implies that Pareto efficient payoff can be achieved in repeated game equilibrium.

A.4. Stochastic Game

Stochastic games are also called dynamic games. They are generalizations of repeated game. In the repeated game, the same stage game is repeated in all periods. However, in stochastic games, the stage game can change randomly or deterministically from period to period depending on the history for a fix set of players. A game state of the stochastic game is one of the possible stage games. A description of the stochastic game specifies how the game states vary from period to period. In general, the probability of the current state depends on the previous state and players' actions. In the special case that the current state is independent of the previous state and players' actions, we have a repeated game with random states that is the base of our model in Chapter 4.

A.5. Game of Imperfect Monitoring

Recall that a repeated game is of perfect monitoring if at the end of each period, each player observes the strategies chosen by other players. In perfect monitoring game, each player precisely observes the past action of all other players without any ambiguity. When the action of other players cannot be absolutely observed, players have only noisy signal about past plays. The signal observed is randomly correlated to the actions of other players. We then have a game of imperfect monitoring. There are two classes of imperfect monitoring game: imperfect public monitoring game and imperfect private monitoring game. A game is of imperfect public monitoring if the signal of past plays, however imprecise and noisy are invariably observed by all players [60]. However, in imperfect private monitoring game, the signal of past plays observed by a player is not observable by others. A detailed presentation of game theory is done by Myerson [35]. Mailath and Samuelsson [60] emphasize repeated games and stochastic games.

A.6. Evolutionary Game Theory

EGT has its foundation in game theory and evolutionary biology. Evolutionary biology studies the processes of change in populations of organisms. Selection and mutation are two important factors of change in Evolutionary biology. Mutation provides diversity in the population while selection promotes some variety over others. The assumption of rationality used in game theory is relaxed in EGT. Unlike game theory where players are intelligent rational and choose strategies, in EGT, players' are programmed to some strategies in the game. Players' are randomly and repeatedly drawn from large populations. They are not assumed to have common knowledge of the game. The payoff is the individual fitness or expected number of surviving offsprings. Two key concepts of EGT are ESS and the replicator dynamic.

A.7. Evolutionary Stable Strategy

Suppose that the individuals in the initial population are programmed to play the same pure or mixed incumbent strategy *x.* Then, a small group of mutants in proportion $\varepsilon \in (0,1)$, all programmed to play the same pure or mixed strategy *y,* appear in the population. Pairs of individuals are repeatedly drawn with equal probability to play the game. Therefore, if an individual is drawn to play the game, the probability that the opponent will play the strategy y is ε, and the probability that the opponent will play the strategy x is 1- ε. As a result, the payoff in a match is equivalent to that in the match where an individual plays the mixed strategy $w = \varepsilon\, y + (1-\varepsilon)x.$ It follows that the payoff of the incumbent strategy is $u(x,w)$ or $u[x, \varepsilon\, y + (1-\varepsilon)x\,]$, and that of the mutant strategy is $u(y,w)$ or $u[y, \varepsilon\, y + (1-\varepsilon)x\,]$. A strategy x is an ESS if and only if:

$$u[x, \varepsilon\, y + (1-\varepsilon)x\,] > u[y, \varepsilon\, y + (1-\varepsilon)x\,]. \tag{4}$$

for a small proportion of mutant ε.

Definition 3**:** A strategy x is an evolutionary stable strategy if for every strategy $y \neq x$ there exists some $\bar{\varepsilon}_y \in (0,1)$ such that inequality (4) holds for all $\varepsilon \in \left(0, \bar{\varepsilon}_y\right)$.

Considering the linearity of u, equation (4) can be written as:

$$(1-\varepsilon)u(x,x) + \varepsilon u(x,y) > (1-\varepsilon)u(y,x) + \varepsilon u(y,y). \qquad (5)$$

For small values of ε, equation (4) is equivalent to either

$$u(x,x) > u(y,x) \ \forall\, y, \qquad (6)$$

Or

$$u(x,x) = u(y,x) \text{ and } u(x,y) > u(y,y) \ \forall\, y \neq x, \qquad (7)$$

as first formulated by John Maynard Smith in [73].

Proposition 4: If x is an ESS then x is a NE

This is because otherwise, inequality (6) should not hold. ESS is a stronger criterion than Nash equilibrium.

Proposition 5: Any strict Nash equilibrium is an ESS.

This is because strict Nash equilibrium always satisfies equation (6)

Proposition 6: An ESS does not always exist in a game.

A population of individuals programmed to play an ESS resist against mutation. This means that such a population cannot disappear in the long run evolution.

A.8. Replicator Dynamic

As described before, ESS focuses on mutation. Actually, the replicator dynamic highlights the role of selection. This model was first proposed by Taylor and Jonker [74]. Assume we have a large but finite population of individuals. Assume also that all individuals are programmed to a pure strategy $i \in K$ in a symmetric two player's game. Let *u* be the payoff function. Let $p_i(t) \geq 0$ be the number of individuals programmed to pure strategy $i \in K$ at time *t*. Thus, the total population is $p(t) = \sum_{i\in K} p_i(t) > 0$. Let the proportion of individuals programmed to pure strategy *i* be:

$$x_i(t) = p_i(t)/p(t). \tag{8}$$

The associated population state is defined as the vector $x(t) = [x_1(t), x_2(t), \dots, x_k(t)\ ]$. Therefore, when the population is in state *x*, the expected payoff to pure strategy *i* at a random match is $u(e^i, x)$ and the payoff of an individual drawn at random in the population is $u(x, x) = \sum_{i=1}^{k} x_i u(e^i, x)$. Assume that payoffs are individual's fitness representing the number of offsprings per unit of time and that strategy is inherited by a continuous time reproduction. We obtain the following population dynamic using time derivative:

$$\dot{p}_i = p_i u(e^i, x). \tag{9}$$

Moreover, equation (8) gives us $p(t). x_i(t) = p_i(t)$ and taking the time derivative on both sides gives:

$$\dot{p}x_i + p\dot{x}_i = \dot{p}_i \Rightarrow p\dot{x}_i = \dot{p}_i - \dot{p}x_i.$$

Also, if we derive $p(t) = \sum_{i \in K} p_i(t)$ on both side and use $u(x, x) = \sum_{i=1}^{k} x_i u(e^i, x)$ we get, after iteration,

$\dot{p} = pu(x, x)$.

Thus, $p\dot{x}_i = \dot{p}_i - \dot{p}x_i \Rightarrow p\dot{x}_i = p_i u(e^i, x) - pu(x, x)x_i$,

and dividing both side by p we finally obtain the population dynamics for the population shares x_i

$$\dot{x}_i = [u(e^i, x) - u(x, x)]x_i. \tag{10}$$

Equation (10) gives the replicator dynamic, this system of differential equation allow us to draw an important conclusion. Subpopulations programmed with better than average strategies grow whereas supbpopulations programmed with worse than average strategies sink.

Let ξ (t, x^0) represent the population state at time t when the initial state is x^0.

Proposition 7: If a pure strategy i is strictly dominated, then $\xi_i\,(t, x^0)_{t \to \infty} \to 0$ for any x^0.

Properly, strictly dominated strategies perish in the replicator dynamic.

Weibull [45] provides a more detailed presentation of EGT.

A.9. Conclusion

In short, we presented in this Section three important concepts; Nash equilibrium, ESS, and the replicator dynamic. We emphasized their similarities and differences. Nash equilibrium is associated with rational players; on the other hand, ESS and the replicator dynamic consider programmed players without any assumption of rationality. However, all three eliminate strictly dominated strategies. Only Nash equilibrium strategies can pass ESS criterions. Also, in the long run, the stationary and stable states in the replicator dynamic correspond to aggregate Nash equilibrium behavior. Thus, a population or a strategy resists an evolutionary process only if it is rational. Finally, EGT and game theory deliver comparable results with different assumptions.

REFERENCES

1. C. A. Kamhoua, N. Pissinou, S. K. Makki “Game Theoretic Analysis of Cooperation in Autonomous Multi-hop Networks: The Consequences of Unequal Traffic Load” in proceedings of IEEE Globecom 2010. Miami, Florida, USA. December 2010.

2. C. A. Kamhoua, N. Pissinou” Mitigating Selfish Misbehavior in Autonomous Wireless Multi-hop Networks Using Stochastic Game Theory” in proceedings of the 35th IEEE Conference on Local Computer Networks (LCN 2010) Denver, Colorado, USA. October 2010.

3. C. A. Kamhoua, N. Pissinou, A. Busovaca,K. Makki "Belief-free Equilibrium of Packet Forwarding Game in Ad Hoc Networks under Imperfect Monitoring" 29th IEEE international performance computing and communications conference (IEEE IPCCC 2010), pp 315-324. Albuquerque, New Mexico, USA. December 2010.

4. C. A. Kamhoua, N. Pissinou, J. Miller, S. K. Makki "Mitigating Selfish Misbehavior in Autonomous Wireless Multi-hop Network Using Evolutionary Game Theory" in proceedings of IEEE Globecom 2010 Workshop, Miami, Florida, USA. December 2010.

5. G. Crosby, N. Pissinou, “Evolution of Cooperation in Multi-Class Wireless Sensor Networks” 32nd IEEE Conference on Local Computer Networks, 2007.

6. M. Felegyhazi, J-P. Hubeaux, L Buttyan “Nash Equilibria of Packet Forwarding Strategies in Wireless Ad Hoc Networks” IEEE Transaction on Mobile Computing, vol.5, NO.5, May 2006.

7. V. Srivastava, P. Nuggehalli, C.F. Chiasseriniand, R. R. Rao. “Cooperation in Wireless Ad Hoc Networks”, IEEE Infocom, San Francisco, CA, USA, 2003.

8. Y. E. Sagduyu, A. Ephremides, “A game-theoretic look at simple relay channel” Wireless Network 12:545–560, 2006.

9. A. Jade, S. K. Madria, M. Linderman, ” IncentiveBased Routing Protocol for Mobile Peer to Peer Networks” Tenth International Conference on Mobile Data Management: Systems, Services and Middleware, 2009.

10. L. Yan, “Cooperative Packet Relaying in Wireless Multi-hop Networks” International Conference on Advanced Information Networking and applications Workshops, IEEE Computer Society. 2009.

11. L. Yan and S. Hailes, “Designing Incentive Packet Relaying Strategies for Wireless Ad Hoc Networks with Game Theory” in IFIP International Federation for Information Processing, Volume 264; Wireless Sensor and Actor Networks II; Ali Miri; (Boston: Springer), pp. 137-148, 2008.

12. L. Yan, S. Hailes,”Cooperative Packet Relaying Model for Wireless Ad hoc Networks”, ACM International Workshop on Foundations of Wireless Ad Hoc and Sensor Networking and Computing (FOWANC’08), Hong Kong, China, 2008.

13. V. Srivastava and L. A. DaSilva, “Equilibria for Node Participation in Ad Hoc Networks – An Imperfect Monitoring Approach,” IEEE Intl. Conf. On Communications (ICC 06), Istanbul, Turkey, June 2006.

14. Z. Ji, W. Yu, K. J. R. Liu, "A Belief Evaluation Framework in Autonomous MANETs under Noisy and Imperfect Observation: Vulnerability Analysis and Cooperation Enforcement," IEEE Transactions on Mobile Computing, vol. 9, no. 9, pp. 1242-1254, April 2010.

15. Z. Ji, W. Yu and K. J. R. Liu, "Cooperation enforcement in autonomous MANETs under noise and imperfect observation", IEEE Secon 2006, pp. 460-468, 2006.

16. W. Wang, M. Chatterjee, K. Kwiat "Cooperation in ad hoc networks with noisy channels" 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks. SECON '09. pp. 547-555 , 2009.

17. W. Yu, K. J. R. Liu, “Secure Cooperation in Autonomous Mobile Ad-Hoc Networks Under Noise and Imperfect Monitoring: A Game-Theoretic Approach” IEEE Transactions on Information Forensics and Security, Vol. 3, No. 2, June 2008.

18. S. Brama, “Spectrum Sharing and Service Pricing in Dynamic Spectrum Access Networks” PhD Dissertation, University of Central Florida, May 2011.

19. K. Komathy, P. Narayanasamy “Trust-based evolutionary game model assisting AODV routing against selfishness” Journal of Network and Computer Applications, 2008.

20. V. Srivastava, J. Neel, A. B. MacKenzie, R. Menon, L. A. DaSilva, J. E. Hicks, J. H. Reed, and R. P. Gilles, "Using game theory to analyze wireless ad hoc networks," in IEEE Communications Surveys and Tutorials, vol. 7, pp. 46-56, 2005.

21. C. Kamhoua, N. Pissinou, K. Makki " Game Theoretic Modeling and Evolution of Trust in Autonomous Multi-hop Networks: Application to Network Security and Privacy" in proceedings of the IEEE international conference on communications (IEEE ICC 2011). Kyoto, Japan. June 2011.

22. B. He, S. Joshi, D. P. Agrawal, D. Sun.” An Efficient Authenticated Key Establishment Scheme for Wireless Mesh Networks” IEEE Globecom 2010. Miami, Florida, USA. December 2010.

23. A. Gupta, A. Mukherjee, B. Xie, and D. P. Agrawal, “Decentralized key generation scheme for cellular-based heterogeneous wireless ad hoc networks,” J. Parallel Distrib. Comput., vol. 67, no. 9, pp. 981–991, 2007.

24. G. Crosby, L. Hester, N. Pissinou, “ Location-aware, Trust- based Detection and Isolation of Compromised Nodes in Wireless Sensor Networks,” International Journal of Network Security,Vol.12, No.2, PP.107–117, Mar. 2011. 107.

25. G. V. Crosby, N. Pissinou, J. Gadze “A Framework for Trust-based Cluster Head Election in Wireless Sensor Networks”, in the proceedings of the Second IEEE

Workshop on Dependability and Security in Sensor Networks and Systems (DSSNS ‘06), April 2006, Columbia, Maryland, USA.

26. G. Crosby, N. Pissinou, “Cluster based Reputation and Trust for Wireless Sensor Networks” in the proceedings of the IEEE Consumer Communications and Networking Conference, Las Vegas, Nevada, USA, January 2007.

27. C. Kamhoua, N. Pissinou, K. Makki " Game Theoretic Analysis of Users and Providers Behavior in Network under Scarce Resources” under submission

28. L. Buttyan, J-P. Hubaux “Nuglets: a Virtual Currency to Stimulate Cooperation in Self-Organized Mobile Ad Hoc Networks” Technical Report DSC/2001/001, Swiss Federal Institute of Technology-Lausanne, Department of communication systems, January 2001.

29. S. Zhong, J. Chen, Y. Yang “Sprite: A Simple, Cheat-Proof, Credit-Based System for Mobile Ad-Hoc Networks” IEEE INFOCOM 2003.

30. A. Josang R. Ismail “The Beta Reputation System” In Proceedings of the 15th Bled Electronic Commerce Conference, June 2002.

31. S. Ganeriwal and M.B. Srivastava, ”Reputation-based Framework for High Integrity Sensor Networks” presented at ACM Workshop on Security of Ad Hoc and Sensor Networks(SASN’04),Washington, D.C.,USA, October 25, 2004.

32. S. Marti, T.J. Giuli, K. Lai, M Baker. “Mitigating routing misbehavior in mobile ad hoc networks.” In processing of the sixth annual international conference in mobile computing and networking, pages 255-265. ACM press,2000.

33. P. Michiardi, R. Molva, “CORE: A COllaborative REputation mechanism to enforce node cooperation in Mobile Ad Hoc Networks.” Communication and Multimedia Security, September, 2002.

34. S. Buchegger and J.-Y. Le Boudec. “Performance Analysis of the CONFIDANT Protocol (Cooperation Of Nodes- Fairness In Dynamic Ad-hoc NeTworks)”. Proceedings of MobiHoc 2002, Lausanne, CH, June 2002.

35. R. Myerson “Game theory: analysis of conflict” Harvard University Press, 1997.

36. M. J. Neely, “Optimal Pricing in a Free Market Wireless Network” Wireless Network 15:901–915, 2009.

37. P. Marbach, Y. Qiu, "Cooperation in Wireless Ad Hoc Networks: A Market-Based Approach," IEEE/ACM Transactions on Networking, Vol.13, pp 1325-1338, 2005.

38. K. C. Nguyen, T. Alpcan, T. Basar” Stochastic games for security in networks with interdependent nodes”. Proc. Of Intl. Conf. on Game Theory for Networks (GameNets), 2009.

39. W. Sun, X. Kong, D. He, X. You. “Information security problem research based on game theory”. International Symposiumon Publication Electronic Commerce and Security, 2008.

40. J. Jormakka and J. V. E. Molsa “Modelling information warfare as a game”. Journal of information warfare; vol.4(2), 2005.

41. Y. Liu, C. Comaniciu, H. Man “ A Bayesian game approach for intrusion detection in wireless ad hoc networks”. ACM International Conference Proceeding Series; vol. 199, 2006.

42. S. Shiva, S. Roy, H. Bedi, D. Dasgupta, Q. Wu. “A Stochastic Game with Imperfect Information for Cyber Security” 5th International Conference on i-Warfare & Security (ICIW), 2010.

43. A. Agah, S. K. Das, K. Basu, M. Asadi, "Intrusion Detection in Sensor Networks: A Non-Cooperative Game Approach," nca, pp.343-346, Network Computing and Applications, Third IEEE International Symposium on (NCA'04), 2004.

44. S. Roy, C. Ellis, S. Shiva, D. Dasgupta, V. Shandilya, W. Qishi “A Survey of Game Theory as Applied to Network Security” 43rd Hawaii International Conference on System Sciences (HICSS). Honolulu, HI, USA. March 2010.

45. J. Weibull “Evolutionary Game Theory” MIT Press, 1997.

46. J. A. Hassan, M. Hassan, S. K. Das, “A Brinkmanship Game Theory Model for Competitive Wireless Networking Environment” in proceedings of the 35th IEEE Conference on Local Computer Networks (LCN 2010). Denver, Colorado, U.S.A. October 2010.

47. S. Sengupta, M. Chatterjee, S. Ganguly, “An economic framework for spectrum allocation and service pricing with competitive wireless service providers," in DySPAN, 2007, pp. 89-98.

48. J. Acharya, R. D. Yates, “Service provider competition and pricing for dynamic spectrum allocation," in GameNets, 2009, pp. 190-198.

49. A. Zemlianov, G. de Veciana, “Cooperation and decision-making in a wireless multi-provider setting," in INFOCOM, vol. 1, 2005, pp. 386-397.

50. J. Jia, Q. Zhang, “Competitions and dynamics of duopoly wireless service providers in dynamic spectrum market," in MobiHoc, New York, NY, USA, 2008, pp. 313-322.

51. J. Musacchio J. Walrand, “WiFi access point pricing as a dynamic game," IEEE/ACM Transactions on Networking, vol. 14, no. 2, 04 2006/04/01.

52. V.Gajic, “Game Theory in Communications: A Study of Two Scenarios” PhD Dissertation, Ecole Polytechnique Federale de Lausane, September 2010.

53. E. Hyytiä, J. Virtamo ” On Traffic Load Distribution and Load Balancing in Dense Wireless Multihop Networks “EURASIP Journal on Wireless Communications and Networking, Volume 2007, 2007.

54. P. P. Pham and S. Perreau “Performance Analysis of Reactive Shortest Path and Multi-path Routing Mechanism with Load Balance” Proceedings of IEEE Infocom, 2003.

55. B. D. Johnson and D.A. Maltz,”Dynamic Source Routing in Ad Hoc Wireless Networks,” Mobile Computing, Vol. 353, Kluwer Academic Publishers, 1996.

56. E. Hyytia, P. Lassila, J. Virtamo” Spatial Node Distribution of the Random Waypoint Mobility Model with Applications” IEEE Transactions on Mobile Computing, VOL. 5, NO.6, June 2006.

57. C. Bettstetter and C. Wagner “ The Spacial Node Distribution of the Random Waypoint Mobility Model” Proceedings First German Workshop Mobile Ad Hoc Netwoek, WMAN 2002, pp 41-58, 2002.

58. D. M. Lazo, A. T. Sherman “An Exact Formula for the Expected Wire Length Between Two Randomly Chosen Terminals” Technical Report TR CS-94-08, Computer Science Department, University of Maryland Baltimore County, July 20,1994.

59. M. Kandori”Social Norm and Community enforcement” The Review of Economic Studies, Vol. 59, No 1 (Jan.,1992), pp.63-80.

60. G. Mailath, L.Samuelson “Repeated Games and Reputations, Long-run relationships” Oxford university press, 2006.

61. J. Ely, J. Valimaki “A Robust Folk Theorem for the Prisoners Dilemma” Journal of Economic Theory 102, p:84-105.January 2002.

62. J. Ely, J. Hörner, W. Olszewski, “Belief-Free Equilibria in Repeated Game,” Econometrica, Econometric Society, vol. 73(2), pp. 377-415, 03, 2005.

63. M. Kandori “Repeated Games, Entry in The New Palgrave Dictionary of Economics, 2nd Edition" January 2006.

64. W. Olszewski,"A Simple Exposition of Belief-Free Equilibria in Repeated Game," Economic Bulletin, Vol. 3, No.58 pp.1-16, November 2007.

65. Y. Yamamoto " Efficiency results in *N* player games with imperfect private monitoring" Journal of Economic Theory Vol. 135, Issue 1 pp 382-413, July 2007.

66. Y. Yamamoto "A Limit Characterization of Belief-Free equilibrium Payoffs in Repeated Games," Journal of Economic Theory, Vol. 144, Issue 2, p: 802-824, March 2009.

67. S. Takahashi "Community Enforcement when Players Observe Partners' Past Play." Journal of Economic Theory 145 (1), pp. 42-62, January 2010.

68. V. Bhaskar, I. Obara, "Belief-Based Equilibria in the Repeated Prisoners' Dilemma with Private Monitoring" Journal of Economic Theory, Elsevier, vol. 102(1), p: 40-69, January 2002.

69. S. Tanachaiwiwat, P. Dave, R. Bhindwale, A. Helmy,"Location-centric Isolation of Misbehavior and Trust Routing in Energy-constrained Sensor Networks," Presented at 2004 IEEE international conference on performance, computing, and communications, 2004.

70. M. Nowak, K. Sigmund "A strategy of win-stay, lose-shift that outperforms tit-for-tat in the Prisoner's Dilemma game" Nature vol.364, 1July 1993.

71. M. Boerlijst, M. Nowak, K. Sigmund "The Logic of Contrition" Journal of Theoretical Biology, Volume 185, Number 3, 1997, pp. 281-293(13).

72. Robert Axelrod, "The Evolution of Cooperation". New York: Basic Books, 1984.

73. J. M. Smith, G. R. Price, "The logic of animal conflict". Nature 246: 15-18, 1973.

74. P. Tailor, L. Jonker "Evolutionary Stable Strategies and Game Dynamic," Mathematical Biosciences, 5:455-484, 1978.

75. http://www2.lifl.fr/IPD/applet-evolution.html

76. W. Stallings “Network Security Essentials: Applications and Standards” Prentice Hall; 4 edition, 2010.

77. J. M. Pacheco, F. C. Santos, M. O. Souza,B. Skyrms. “Evolutionary dynamics of collective action in N-person stag hunt dilemmas” proceeding of the royal society B 22, vol. 276 no. 1655 315-321. 2009.

78. Fierce Wireless, “Is usage-based pricing inevitable?” [online]. Available: http://www.fiercewireless.com/story/usage-based-pricing-inevitable/2010-02-03.

79. D. Bertsekas, “Dynamic Programming and Optimal Control”, vol. 1,2, Athena Scientific, Belmont, MA, Second edition, 2001.

VITA

CHARLES ALEXANDRE KENMOGNE KAMHOUA

EDUCATION

2008 M.S., Telecommunications and Networking
Florida International University
Miami, Florida

1999 B.S., Electronics
University of Douala/ENSET
Douala, Cameroon

AWARDS

2011 National Academies Research Associateship award at the
Air Force Research Laboratory

2008 FAEDS Teacher Awards

WORK EXPERIENCE

2008-2011 Instructor, Research/Teaching assistant
Florida International University
Electrical and Computer Engineering Department
Miami, Florida

2008-2011 Research Mentor
Florida International University
NSF/REU at Telecommunications and Information Technology Institute
Miami, Florida

2007-2008 Mathematics Teacher
Miami Dade County Public Schools
Miami Carol City Senior High School
Miami, Florida

2006-2007 Mathematics Tutor
Miami Dade College
Miami, Florida

2000-2005 Electronics Teacher/Head of Department
Government Technical High School of Batcham
Batcham, Cameroon

PUBLICATIONS AND PRESENTATIONS

1. Charles Kamhoua, Niki Pissinou, Kia Makki " Game Theoretic Modeling and Evolution of Trust in Autonomous Multi-hop Networks: Application to Network Security and Privacy" in proceedings of the IEEE international conference on communications (IEEE ICC 2011). Kyoto, Japan. June 2011

2. Charles Kamhoua, Niki Pissinou, Alan Busovaca, Kia Makki "Belief-free Equilibrium of Packet Forwarding Game in Ad Hoc Networks under Imperfect Monitoring" in proceedings of the 29th IEEE international performance computing and communications conference (IEEE IPCCC), pp 41-50, Albuquerque, New Mexico, USA. December 2010.

3. Charles Kamhoua, Niki Pissinou, Kami Makki "Game Theoretic Analysis of Cooperation in Autonomous Multi-hop Networks: The Consequence of Unequal Traffic Load" in the proceedings of IEEE Globecom Workshop 2010, pp 2036-2041, Miami, Florida, USA. December 2010.

4. Charles Kamhoua, Niki Pissinou, Jerry Miller, Kami Makki "Mitigating Selfish Misbehavior in Autonomous Wireless Multi-hop Network Using Evolutionary Game Theory" in the proceedings of IEEE Globecom Workshop 2010, pp 2020-2025, Miami, Florida, USA. December 2010.

5. Charles Kamhoua, Niki Pissinou "Mitigating Selfish Misbehavior in Autonomous Wireless Multi-hop Networks Using Stochastic Game Theory" in proceedings of the 35th Annual IEEE Conference on Local Computer Networks (LCN 2010), pp 236-239, Denver, Colorado, USA. October 2010.

6. Charles Kamhoua, Niki Pissinou, Kia Makki " Game Theoretic Analysis of Users and Providers Behavior in Network under Scarce Resources" under submission.

7. Charles Kamhoua, Niki Pissinou, Kia Makki "Game Theoretic Modeling of Trust and Security in Wireless Networks" To be submitted at IEEE transaction on Communication.

8. Charles Kamhoua, Niki Pissinou, Kia Makki "A Survey of Packet Forwarding Game in Multi-hop Networks" To be submitted at IEEE communication and tutorials.