GAME THEORY MANET ROUTING FOR JAMMING ENVIRONMENT

by

Yi Zhu

A thesis submitted to the Faculty of the University of Delaware in partial fulfillment of the requirements for the degree of Master of Science in Electrical and Computer Engineering

Summer 2014

ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346

GAME THEORY MANET ROUTING FOR JAMMING ENVIRONMENT

by

Yi Zhu

Approved: _____

Stephan Bohacek, Ph.D.
Professor in charge of thesis on behalf of the Advisory Committee

Approved: _____

Kenneth E. Barner, Ph.D.
Chair of the Department of Electrical and Computer Engineering

Approved: _____

Babatunde A. Ogunnaike, Ph.D.
Dean of the College of Engineering

Approved: _____

James G. Richards, Ph.D.
Vice Provost for Graduate and Professional Education

ACKNOWLEDGMENTS

I would like to express my deep appreciation to my advisor Dr. Stephan Bohacek, for introducing me to this exciting and challenging thesis, for his constant encoragement and acdemic advice. He has been extremely helpful to me. I must thank my colleague Jonghyun Kim for his sincere help in simulation experiments.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

A Mobile Ad Hoc Network (MANET) is a set of self-organizing wireless mobile nodes, which communicate with each other without any existing network infrastructure or centralized network management. Wireless communication over MANET follows a multi-hop manner, in which each node can be a transceiver or a router. MANET has spurred considerable research interests and applications in the felids of vehicle network, sensor network and tactical communication network. Tactical communication network works in a highly dynamic environment with the kinds of interferences and jammings, which can cause low packet delivery rates, long delays or even interruptions. Moreover, the effects may spread across multiple network protocol stacks, typically physical (PHY) layer and media access control (MAC) layer. Because of the interruptions and jammings, MANET routing algorithms must be robust enough to give reliable network services. Furthermore, since the end-to-end communications of MANET are achieved via multi-hop relays, the routing protocol is an essential factor that affects the overall performance of MANET. The Optimized Link State Routing (OLSR) is a routing protocol for MANET, which is built on the classical link state algorithm with multi-point relays (MPRs). OLSR has been shown to be suitable for large-scale dense wireless networks. In this thesis, we developed and implemented a game theory routing protocol (GTR) based on conventional OLSR for jamming environment. With game theory, GTR makes probabilistic decision according to the jamming conditions. GTR have been put in place over OLSR in Qualnet network simulation tool. Impacts of jamming on GTR and OLSR are studied. The results show that GTR has better overall performance, specifically lower end-to-end delay and higher packet delivery rate, over OLSR.

Chapter 1

INTRODUCTION

In this section, an introduction of Mobile Ad Hoc Network (MANET) is given following by applications and challenges. Technologies that are suitable for MANET are presented. Routings over MANETs is introduced with a discussion of challenges and design issues. An overview of routing protocols is given.

1.1 MANET

A MANET is a system of wireless mobile nodes that autonomously self-organize a temporary network without using any preexisting network infrastructure [1]. Without network infrastructure, MANET, unlike wireless local area network or a cellular communication network, do not have centralized administration, where nodes communicate with each other directly, instead of transmitting via a base station. If the destination node is out of transmission range of the source node, other intermediary nodes will act as routers or relay nodes to assist the transmission in a multi-hop manner [2], as shown in Figure 1.1. Compare to wired networks, wireless channels are time-various caused by multipath fading and interference. Furthermore, with the mobility of nodes in MANET, the network topologies and link connectivities are dynamically changed, which makes the multi-hop transmission very challenge. As a result, routing protocol is a main factor that affect the overall performance of MANET.

1.1.1 Applications

Many wireless mobile applications require for self-organize and self-management MANET networking. One of typical applications is wireless body area networks (WBANs) [3]. Nowadays, lots of health-care and athletics training applications are

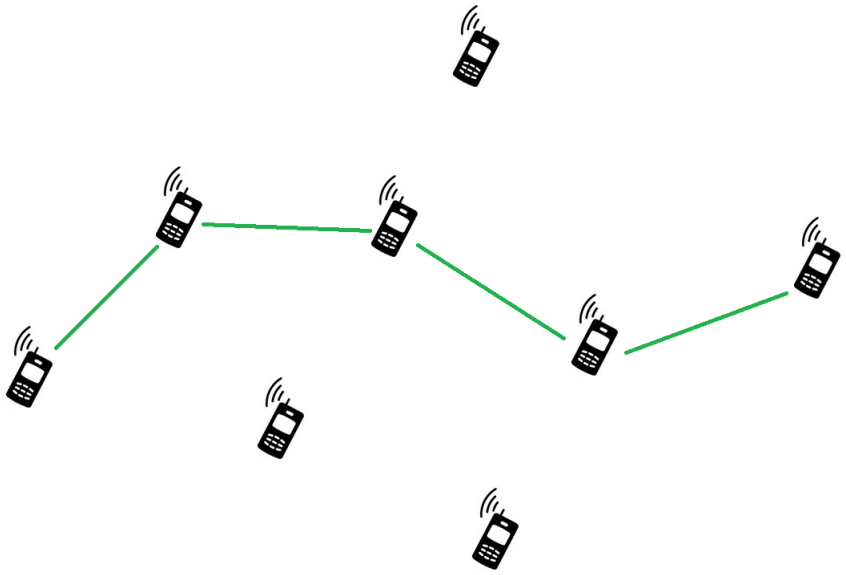

Figure 1.1: Multi-hop transmissions of MANETs.

built on wearable devices. For instance, a private health-care system may consist of several wireless sensor node to collect physiological signals from the patient. In this scenario, there is no network infrastructure and centralized network management. In addition, devices are heterogeneous. Also, the system is required to send out physiological signals to other networks. Another application of MANET is Vehicular Ad Hoc Networks (VANETs). Another application of MANET is Vehicular Ad Hoc Networks (VANETs) [4]. VANETs provide real-time traffic information, such as traffic flows and traffic accidents, which are helpful for drivers to avoid traffic congestions and make the driving safer. Wireless sensor network (WSN) is built on MANET [4]. It is utilized to sense the environment of a target area. Usually, it contains two kinds of node, which are sensor nodes and actuators. Sensor nodes gather information and send it to actuators. Actuators gather information from sensors and send it to a base station. Besides, Internet access and multimedia services can be provided through VANETs. Communications on the battlefield is another typical sinuation that is suitable for MANET,

since there is no network infrastructure available. And this environment is highly dynamic with jammings. MANETs are also applicable for temporarily established mobile applications, such as rescue robots and cooperating industrial robots.

1.1.2 Challenges

As promising wireless networks in future mobile applications, MANET has spurred considerable interests from research and industrial communities. However, it has also brought many technical and theoretical challenges [5]. One of the main challenges is the energy constraint. The power source of most of the wireless terminals is battery. There are four energy-consuming components of a node in MANET: RF front-end, signal processing back-end, sensors. These parts highly constrain the system life-time of MANET. Limited transmission range is another challenge. Because the coverage area of the source node cannot contain the destination node. The transmission must be helped by relay nodes between them to form a multi-hop path. Discovering this path is a difficult for MANET. In many sinuations, such as VANETs, nodes in the network are dense. Moreover, they have the feature of high mobility. These factors make finding path a tricky problem. Limited bandwidth is a constraint for high data rate data transmission, although it extends the coverage area.

1.1.3 Technologies

There are various technologies and standards that are suitable for MANETs. Bluetooth is part of them [6]. Operating band is 2.56. It has the characteristic of high data rate transmission with low energy cost, which makes it suitable to provide data transmission among energy-constrained mobile devices and sensor nodes. The transmission range is up to 100 meters. Another feature of Bluetooth is self-organized. Bluetooth-enabled devices can identify each other and setup networks themselves, which exactly meet the demands of MANETs. ZigBee is a specification for personal area networks [7]. The transmission range is up to 100 meters. It supports 250 kbps

data rate, which is suitable for a sensor node to transmit data to a base station periodically. In addition, it can work on low cost devices. The transmission is secured by 128 bit symmetric encryption keys. Devices based on ZigBee can build a network without network infrastructure and centralized management. WiFi is another option for MANETs. It is a commonly used standard for wireless local area networks, which are based on IEEE 802.11[6]. The operation band is 2.4GHz and 5GHz. The transmission range is approximately 20 meters. It is particularly used to support high speed reliable wireless Internet access. Moreover, it is widely supported by daily used devices, such as a desktop, laptop, tablet and mobile phone. Security protocols, WPA or WPA2, are employed to protect the data communication.

1.2 Routing

A routing protocol can be seen as a model that responsible for computing the route between the source and the destination. Usually, MANETs are built for temporary used without any existing infrastructures. All nodes collaborate together to fulfill tasks. The motions of nodes in MANETs are random, independent and unpredictable, which lead to frequent network topology changes. MANETs should be capable of discover and maintain route dynamically.

1.2.1 Challenges

Routings for MANETs is very challenging, because of characteristics of MANETs and additional requirements for the traditional routing in wired networks. First of all, wireless channels are time vary due to shadowing and fading, which lead to unstable channel conditions or even channel disconnection. Reflecting to routing, it causes frequent network topology changes, which are the base for routing computing. To tackle this problem, MANETs routing protocol must sense network topology more frequently than that of the traditional routing protocol via sending and broadcasting link detection messages and link information messages, which decrease the utility of the network. Moreover, the wireless bandwidth is very limited. And sometimes, it is not sufficient

for data transmission. Sending and broadcasting link information messages make it worse. Because these messages are still sent, even though there is no data traffic in the network. In addition, it consumes extra energy, which is restricted in mobile devices battery. Furthermore, applications with the feature high mobility, such as vehicular networking, make the rate of topological changes much higher compared to that of wireless networks, such as WLAN. It results in less reliable routes increasing the route maintaining and updating process, which results in more control packets overhead and end-to-end delay.

Another challenge for MANETs routing protocol is unidirectional links. Compare with bidirectional links, where two nodes can hear each other, unidirectional links are a phenomenon that one node can hear the other; but the opposite is not true. However, link detection is based on handshaking between two nodes, which only work for bidirectional links. Unidirectional links can never be detected based on the present routing protocols. The life-time of a path is another design issue of MANETs routing protocol. Recognizing paths with short life-time benefits for MANETs a lot. Because these paths are unreliable and frequently disconnected, which is expected to result in data loss. Also, when a path is disconnected, a new path must be built, which brings extra bandwidth and energy consumption. Mobility is one of the main factors that affect paths life-time. Nodes with high mobility may move outside the transmission range of another node, which leads to link failure. Another consequence of high mobility is fast various channels. Besides to mobility, the energy of nodes in the path can also affect the paths life-time. For a link in a path, two nodes must be included in each others transmission range. Otherwise the link will be disconnected. The energy of a node is a main factor that determines its transmission range. If a node with low energy is included in a path, that path has high risk to be interrupted.

1.2.2 Classifications

Depending on the management of the routing tables and what and how informations are acquired and maintained to compute the routing tables, the routing protocols

can be categorized into reactive routing protocols, proactive routing protocols, and hybrid routing protocols[8].

1.2.2.1 Proactive Routing Protocol

A proactive routing protocol is also referenced as table-driven routing protocol. With proactive routing protocol, nodes in the network regularly exchange network topology information with each other in order to maintain up-to-date forwarding table whether there is data traffic or not. Besides, proactive routing protocol attempts to compute routes between all possible pairs of source and destination according to the network topology information. Therefore, a source node can get a routing path immediately if it needs one.When a network topology change occurs, network topology information must be broadcast across the network to notify the change.

Most of proactive routing protocols have been designed based on the protocols for wired networks with some modifications in order to adapt to MANETs. Compare with conventional protocols for wired networks, proactive protocols reduce the amount of traffic of control messages while preserving their basic functionalities.

The problem of proactive routing protocol is that periodical exchanging control messages are bandwidth-consuming as well as energy-consuming, especially when it was applied to MANETs, where bandwidth and energy are constraints; and where the network topology changes frequently. Moreover, maintaining updated network topology information results in considerable overheads, especially for large-scale MANETs.

The Optimized Link State Routing (OLSR)[9] is a proactive link state routing protocol, where nodes exchange link state information and compute the shortest path to other nodes. Multi-Point Relay (MPR) nodes are used to optimize the flooding of link state information. Destination-Sequenced Distance Vector (DSDV) is a proactive protocol based on the classical Bellman-Ford mechanism with sequence numbers to guarantees loop free routes[10].

DSDV is regarded as a typical example to explain the characteristics of proactive protocol. In order to maintain consistent and up-to-date routing tables, DSDV requires

each node to periodically advertise its routing information to its immediate neighbors. There are two types of routing information: full dump packets and incremental packets. Full dump packets contain all entries in the routing table. Incremental packets contain the changed routing entries, since the last full dump packets. Incremental packets are broadcasted more frequently than full dump packets.

Each node holds a routing table that stores the next hop toward all possible destinations. Each routing entry has the total number of hops required to reach the destination and the unique sequence numbers to distinguish up-to-date route and out-of-date route. The route updates can be either time-driven or event-driven. Specifically, for time-driven, the route table is refreshed periodically; for event-driven, it is updated, when there is a change in routing information.

One of the major features of DSDV is to use a sequence number distinguishing latest routes from out-of-date routes, ensuring loop-free routing and solving count-to-infinity problem. It is generated from the destination. The principal is route with a higher sequence number is newer. For two identical routing entries with different sequence numbers, it discards the one with less sequence number, since that entry is not the latest one. For two identical routing entries with identical sequence numbers, the one with the least total number of hops is chosen.

The advantage of DSDV is that there is no transmission delay if the route of the destination is already in the routing table. The disadvantage is that DSDV is not scalable for large network. Because broadcasting routing information frequently consumes lots of bandwidth and results in substantial overhead.

1.2.2.2 Ractive Routing Protocol

A reactive routing protocol is also called on-demand routing protocol. In a reactive routing protocol, a routing path is created in the presence of data transmission in need of a route. There are two procedures for path management: path discovery and path maintenance.

A route discovery is invoked when a source has needs to transmit data to the destination; and there is no route entry to that destination. It floods a route request packet throughout the network. Route reply packet is sent back if the destination itself or a node with route to the destination is eventually reached. Once the route between the source and destination has been established, the data could be transmitted through the selected route. The discovery procedure terminates when either a route has been found or no route available after examination for all route permutations. Because of the mobility and dynamic environment, routes may be disconnected. Route maintenance is an important operation to cope with this problem.

Compared to proactive routing protocols, the advantage of reactive routing protocols is that the control overhead is low, since it does not maintain topology information. Therefore, it is scalable for large-scale network with high mobility. The disadvantage is that the route discovery procedure before data transmission results in high delay.

One of the representative reactive routing protocol is Dynamic Source Routing protocol (DSR) using a source routing instead of using the routing table[11]. Ad hoc On-Demand Distance Vector routing protocol (AODV) is another routing protocol that belongs to the group of reactive routing protocol[12]. It is a type of single path and loop-free distance vector protocol.

AODV is regarded as a typical example to explain the characteristics of reactive protocol. It inheritances the sequence number from DSDV to maintain up-to-date routes and to prevent loops. Its route discovery mechanism is taken from DSR. Unlike DSR, instead of using source routing, AODV uses routing based on routing table.

The route discovery process is triggered by the source when it is required to send a packet to the destination; and it has no routing information in its routing table. The source floods a Route Request message (RREQ). If the node does not get a route to the requested destination, it broadcasts the RREQ. If a node has a route entry for the destination, it responds by sending a Route Reply message (RREP) back to the source node. The reverse path is built through all nodes back to the source. It utilizes

the previous hop of the RREQ as the next hop on the reverse path. When the source receives a RREP, the route is established. The data transmission is started through the discovered path. Figure 1.2 illustrates an example of AODV, where node 1 is the source; node 2 is the destination. Blue arrow is the propagation of RREQ. Red arrow is the reverse path.

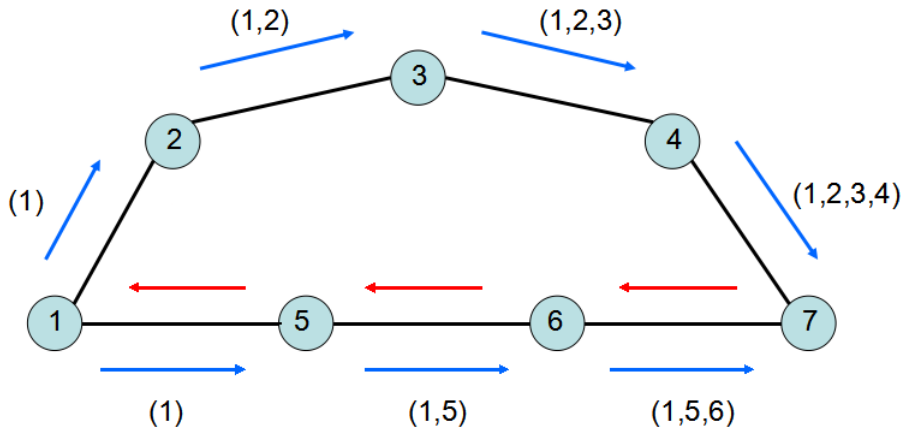

Figure 1.2: An example of AODV routing

Route maintenance and neighbor discovery are based on broadcasting HELLO messages periodically. For neighborhoods discovery, after a neighbor receives a HELLO message, it will check whether a neighbor exists in its routing table. If so, it increases the life-time of that route entry. Otherwise, it adds the neighbor to its routing table. For neighborhoods discovery, when a link failure occurs, a node will send an Unsolicited Route Reply packet to other nodes on the path and delete all broken route entries When the source receive that packet, it will flood RREQ to initiates a new path discovery procedure.

The strength of AODV is that there is no overhead of topology information maintaining and updating, which saves bandwidth and energy. The disadvantage is that AODV need to discover the path when there is a need. That causes relatively long delay before the transmission.

1.2.3 Routing Metric

Routing metric is a parameter of the route computed by the routing protocols. It shows the quality of a route. Wireless networks suffer from routing variations due to shadowing and fading the propagation environment as well as the mobility of terminals in the network. The aim of routing metric is to guide the routing protocol to find a path with the minimum cost. Routing metric is a critical factor that affects the overall performance of MANETs.

1.2.3.1 Link Metric

Expected transmission count (ETX) metric was among the first attempts to improve the link quality estimation in high rate wireless networks[13]. ETX estimates the expected total number of transmissions needed to successfully deliver a packet through a link. Two parameters are involved in ETX, which are forward delivery ratio and reverse delivery ratio. Forward delivery ratio, d_f, is the error probability of the packet received by the destination. Reverse delivery ratio, d_r, is the error probability of acknowledgment of the packet received by the source. The formular of ETX is demonstrated in 1.1.

$$\frac{1}{d_f \cdot d_r} \tag{1.1}$$

The advantage of ETX is that it truly reflects the throughput of the network, since it is based on message delivery ratios. Besides, it is a bidirectional link quality estimation, which is able to recognize unidirectional links. The problem of ETX is that it requires transmitting link prob messages periodically, which is bandwidth and energy consuming.

An other problem of EXT is that it does not take into consideration the data rate of the link[14]. Most of the wireless protocols support multi-rate transmission.

ETT modified EXT by involving the size of the packet, S, and the data rate, R. The formular of ETT is shown in 1.2.

$$ETT = ETX \cdot \frac{S}{R} \tag{1.2}$$

The advantage of ETT is that it considers the date rate a link. However, it is still bandwidth and energy consuming.

To save bandwidth and energy while speeding up the estimation time, some metrics from PHY layer are considered. Received signal strength indicator (RSSI) is a member of the basic metrics, which is measured from radio hardware. A typical approach to measure RSSI is to average energy of eight symbols of packets. Another metric is link quality indication (LQI), which is a measure of chip error rate. It is the correlation value of eight symbols ranging from 0 to 255. LQI is higher related with link quality than RSSI[15].

1.2.3.2 Path Metric

The total number of hops needed to route from the source to the destination is the most commonly used path metric for MANETs routing protocols. It is sensible, since longer path consumes more bandwidth and energy and has a higher packet loss probability. The merit is in fact probably the simplest metric to implement. The drawback is that it only considers the length of the path ignoring the impacts of link qualities. One path with a smaller number of hops, but bad link qualities, may have higher packet loss probability than the path with a larger number of hops, but good link qualities.

Some methods intent to use the minimum number of hops strategy with link qualities. One of them is to remove links with low link qualities before computing the path[16]. This method is based on the fact that links with very low qualities highly degrade the qualities of a path. On other words, they can be considered as noises interfering route computing. The problem with this method is that it still not able to differ from the path with high quality and the path with low quality. In the stage of

route computing, it deletes bad links. However, in the stage of route selection, it does not take into account path qualities.

The packet delivery rate of a path can be invoked as a path metric[17]. The log-scaled packet delivery rate is used in a link. The sum of log-scaled packet delivery rates of all links in the path is the packet delivery rate of that path. It takes qualities of all links into consideration, regardless of the length of the path. Also the computational accuracy is an element that affects the estimation of that path metric.

1.3 Jamming

Jamming is identified as the act of intentionally directing electromagnetic energy towards a communication system to disrupt or prevent signal transmission[18]. For MANETs, it is built on a shared medium, which makes it easy for adversaries to launch jamming attacks. Jamming can severely interfere with the normal operation of MANETs. For applications with QoS requirements, a disruption of the proper data transmission may lead to disastrous results.There are four ways to classify jamming[19].

1.3.1 Classifications

Jamming can be classified based on jamming methods. One method is that the jammer keeps emitting powerful noise to occupy the communication bandwidth. The other method is to disrupting of networks proper functions by resembling network traffic. Jamming can also be classified by the layers it targets at. Basically, there are two kinds of jamming: Radio jamming and Link-layer jamming. Different kinds of jamming have different spectrum characteristics. Spot jamming directs all its transmitting power on a single frequency. Sweep jammings full power shifts rapidly from one frequency to another. Barrage jamming jams a range of frequencies one time. Deceptive jamming mislead the network by flooding fake data in a sole frequency or in a set of frequencies completing the damage without leaving any trace. Jamming can also be classified by jamming behaviors. Constant jamming emits continuous radio

signals. Random jamming sleeps for random time and jams for a random time. Reactive jamming sleeps when the channel is idle and jams when the channel is active. Constant jammers and reactive jammers are effective jammers in that they can cause the packet delivery ratio to fall to zero or almost zero, if they are located within a suitable distance from the victim nodes.

1.3.2 Jamming detection

Jamming detection is about differentiating a jamming scenario from various network conditions, such as data transmission, congestions, and failures at the sender side. Signal strength is a case in measurement of jamming. The rationale behind using this measurement is that the signal strength distribution may be influenced by the presence of a jammer. There are two basic strategies The first approach uses either the average signal value or the total signal energy over a window of several signal strength measurements. The second strategy aims to capture the shape of the time series by representing its spectral behavior. Carrier sensing time can also be utilized to detect jamming. It is very natural for one to keep track of the amount of time it spends waiting for the channel to become idle, i.e. the carrier sensing time, and compare it with the sensing time during routine traffic operations to determine whether it is jammed. Packet delivery ratio is another indicator reflecting the presence of jamming. Since a jamming attack will degrade the channel quality surrounding a node, the detection of a radio interference attack essentially boils down to assess whether the communication node can send or receive packets in the way it should have had the jammer not been present.

1.4 Thesis Outline

In this thesis, we develop a proactive game theory routing protocol for MANETs builds upon OLSR in jamming environment. The organization of this thesis is as follows: Section 2 is the description of OLSR followed by the game theory in Section

3. We present proposed routing protocol in Section 4, and in Section 5 the simulation results are demonstrated. Finally this thesis is concluded in Section 6.

Chapter 2

OLSR

In this section, OLSR routing protocol is presented in details, which is the foundation of the proposed game theory routing.

2.1 Introduction

OLSR[20] is a proactive routing protocol built upon the link state algorithm for wired networks. In order to adapt it to MANETs, it reduces the traffic of control messages by introducing multipoint relays (MPRs), which are selected nodes which generate and forward broadcast messages during the flooding process. Compare with OLSR, traditional link state algorithm floods among all nodes in MANETs. Because of that optimization, it is scalable for large-scale MANETs. The transmission of link state informations does not require to setup reliable links. They are delivered via User Datagram Protocol(UDP) periodically. Sequence number is used to maintain the order of control messages. According to these link state informations, OLSR select the shortest path with minimum total number of hops as the route for all possible pairs of the source and the destination.

In [20], the core functionalities, which are mandatory for implementations, are defined. A collection of auxiliary functionalities are describe for optional implementations. The core functionalities are listed below.

- Packet Format and Forwarding: The formats of packets are specified, which are encapsulated in UDP datagram. And the forwarding mechanism is defined.
- Link Sensing: Link Sensing is achieved via periodically transmit HELLO messages.
- Neighbor detection: Based on the results of link sensing, neighbors are detected.

- MPR Selection: A set of neighbors are selected as MPRs.
- Topology Control Message Diffusion: Up-to-date topology information is exchanged and maintained at each node.
- Route Computing: Routes for all possible destinations are computing, according to the topology information.

2.2 Protocol Format

In this section, the formats of OLSR packet and message are given with explanations of each entry.

2.2.1 Packet Format

Figure2.1 shows the format of OLSR's packet.

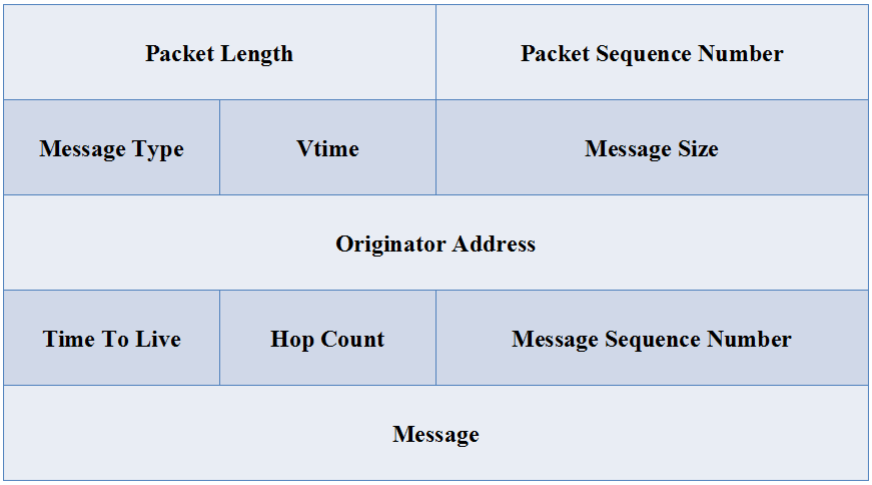

Figure 2.1: The format of a OLSR packet

The definitions of each field are listed below.

- Packet Length: Total number bytes in the packet.
- Packet Sequence Number: It will be increased by one, when a new packet is generated.
- Message Type: It can be HELLO-messages, TC-message or users self-defined message.

- Vtime: The packet is valid within this period of time.
- Message Size: Total number bytes in the message.
- Originator Address: The address of the node generating the packet.
- Time To Live: It will be decreased by one after relaying each time.
- Hop Count: It will be increased by one after relaying each time.
- Message Sequence Number: It will be increased by one, when a new message is generated.
- Message: The content of the message.

2.2.2 Message Format

Figure2.2 shows the format of a HELLO message.

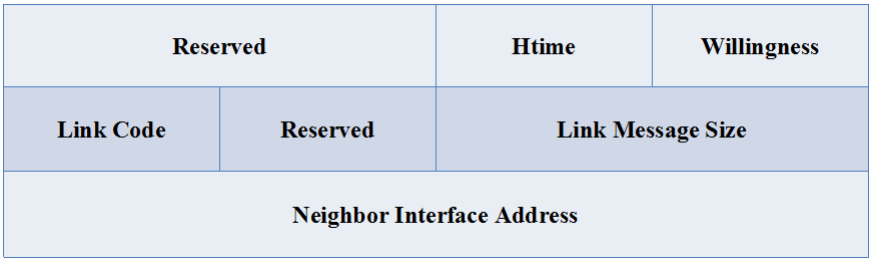

Figure 2.2: The format of a HELLO message

The definitions of each field are listed below.

- Reserved: This field is setted to zero.
- HTime: The period of sending each HELLO message.
- Willingness: The willing of forwarding data for other nodes, which can be DEFAULT, ALWAYS or NEVER.
- Link Code: The type of the neighbor and the type of the link.
- Link Message Size: The size of the link message.
- Neighbor Interface Address: The address of an interface of a neighbor node.

Figure2.3 shows the format of a TC message.

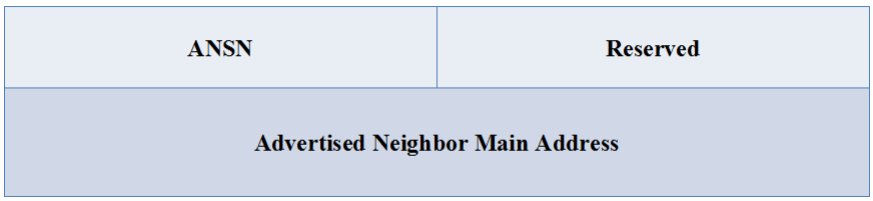

Figure 2.3: The format of a TC message

- Advertised Neighbor Sequence Number (ANSN): When a node detects a change in its advertised neighbor set, ANSN will be increased by one.
- Reserved: This field is setted to zero.
- Advertised Neighbor Main Address: This field contains the main address of a neighbor node.

2.2.3 Link, Neighbor and Topology Information

In this section, the informations needed to maintain up-to-date routing table are described. There are three basic informations: link information, neighbor information and topology information. All of them will be presented sequentially.

Link informations, regarded as link set, contains informations about links to neighbors. The element in the link set is called link tuple, which is shown below.

$$(L\_local\_iface\_addr, L\_neighbor\_iface\_addr, L\_SYM\_time, L\_ASYM\_time, L\_time)$$

- $L\_local\_iface\_addr$: The address of the node itself.
- $L\_neighbor\_iface\_addr$: The address of its neighbor.
- L_SYM_time: The time interval in which the link is symmetric.
- L_ASYM_time: The time interval in which the link is asymmetric.
- L_time: The time interval in which the link is valid.

Neighbor informations, regarded as neighbor set, contains informations about neighbors of the node. There are four kinds of elements in neighbor informations: neighbor set, 2-hop neighbor set, MPR and MPR selector.

Neighbor set contains the neighbor that is one-hop away from the node, which is shown below.

$$(N\_neighbor\_main\_addr, N\_status, N\_willingness)$$

- N_neighbor_main_addr: The main address of the neighbor.
- N_status: It can be either NOT_SYM or SYM.
- N_willingness: The willingness relay traffic of other nodes.

2-hop neighbor set contains the neighbor that is two-hop away from the node with symmetric links, which is shown below.

$$(N\_neighbor\_main\_addr, N\_2hop\_addr, N\_time)$$

- N_neighbor_main_addr: The main address of the neighbor.
- N_2hop_addr: The main address of the 2- hop neighbor.
- N_time: The time interval in which the 2- hop neighbor is valid.

MPR set contains the list of addresses of the nodes MPRs, which are selected to retransmit the flooding control messages.

MPR selector set contains the list of addresses and valid time of its neighbors, which select the node as MPR.

Topology informations, regarded as topology set, contains informations about network topologies. The element in the topology set is called topology tuple, which is shown below.

$$(T\_dest\_addr, T\_last\_addr, T\_seq, T\_time)$$

- T_dest_addr: The address of the destination.
- T_last_addr: The address of next-hop that can reach the destination.
- T_seq: The sequence number of the topology tuple.
- T_time: The valid time of the topology tuple.

2.3 Link Sensing

In link sensing, each node broadcasts control messages periodically to its neighbors. Specifically, nodes exchange the link information with their neighbors via HELLO message. Finally, all nodes get its up-to-date symmetric and asymmetric links. Basically, there are two operations: link information generating and link information processing.

2.3.1 Link Information Generating

Each node broadcasts its link set to its neighbors periodically. A typical value of the period is 2 seconds. The link set is encapsulated in a HELLO message. And the HELLO message is encapsulated in OLSRs packet. The flow diagram of link information generating is shown in Figure2.4.

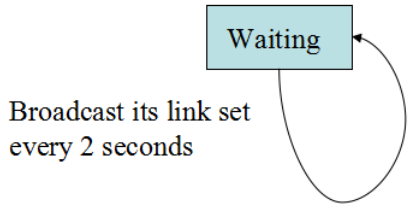

Figure 2.4: The flow diagram of link information generating

2.3.2 Link Information Processing

The link set is updated according to the received HELLO message. There are three cases. In case 1, if the link tuple does not exist, it will be added as an asymmetric link. In case 2, if the link tuple exists; and it is a asymmetric link, it will be changed to a symmetric link. In case 3, if the link tuple exists; and it is a symmetric link, the valid time will be extended. For all links, if the valid time of a link is expired, it will be deleted. The flow diagram of link information processing is shown in Figure2.5.

2.4 Neighbor Detection

In neighbor detection, it is based on the results of link sensing. Specifically, after link sensing, the link set is updated. Each link is related to a neighbor. According

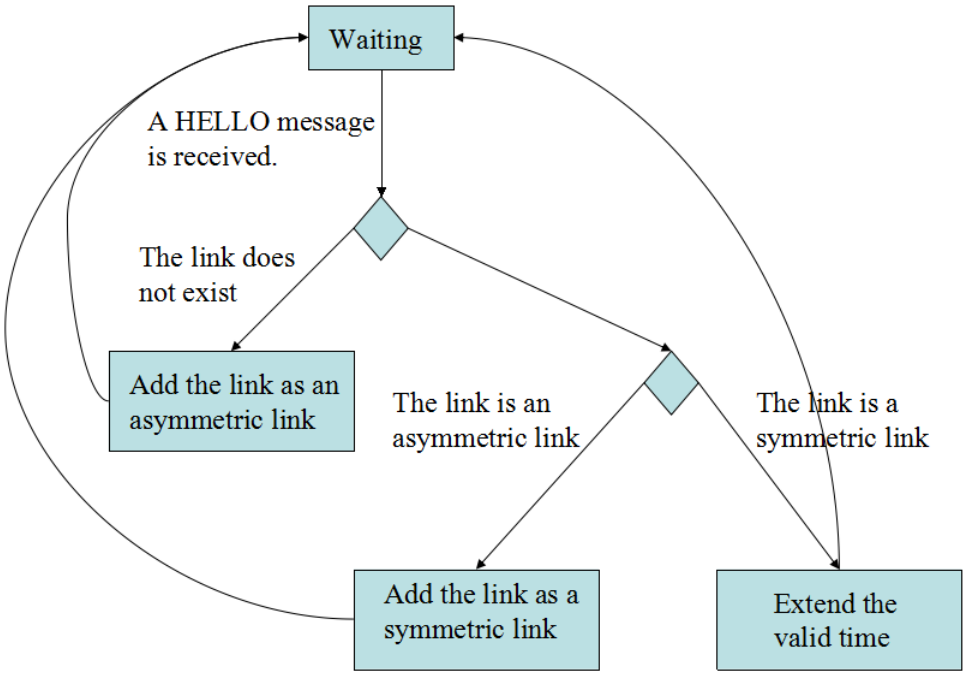

Figure 2.5: The flow diagram of link information processing

to these relationships, neighbor set is updated. Finally, all nodes get its up-to-date symmetric and asymmetric neighbors.

2.4.1 Neighbor Information Updating

The neighbor set is updated according to the update of link set. There are four cases. In case 1, if a new asymmetric link is added, then a new asymmetric neighbor will be added. In case 2, if a new symmetric link is added, then a new symmetric neighbor will be added. In case 3, if a link is dropt, then the corresponding neighbor will be dropt. In case 4, if the valid time of a link is updated, then the corresponding valid time of the neighbor will be updated. The flow diagram of neighbor information processing is shown in Figure2.6.

2.4.2 2-hop Neighbor Information Updating

The 2-hop neighbor set is updated according to the update of neighbor set. If a symmetric neighbor is added, the link set of that neighbor is checked. If the symmetric

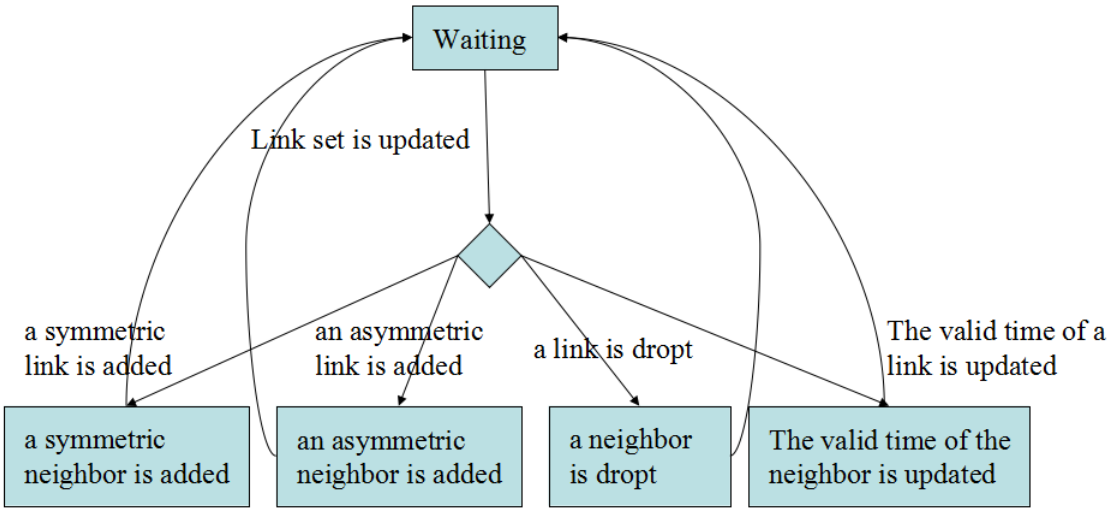

Figure 2.6: The flow diagram of neighbor information updating

neighbors of that neighbor are not in the 2-hop neighbor set, then add them. If they are already in the set, then expend the valid time. All 2-hop neighbor with invalid time are dropt. The flow diagram of 2-hop neighbor information updating is shown in Figure2.7.

2.5 MPR selection

MPR is a set of symmetric neighbors that generates and floods the control messages. It reduces overhead of flooding mechanism in conventional link state routing protocols. Each node has its own MPR set. The MPR set will be updated, whenever there is a change in neighbor set or 2-hop neighbor set. The neighbors in MPR set must fulfill two requirements. The first one is that it must be one of the symmetric neighbors. The second one is that it must be the symmetric neighbor of all strict 2-hop neighbors. Strict 2-hop neighbor is defined as a node that 2-hop away from the node; it is not the node itself; and it is not the nodes neighbor. The flow diagram of MPR selection is shown in Figure2.8.

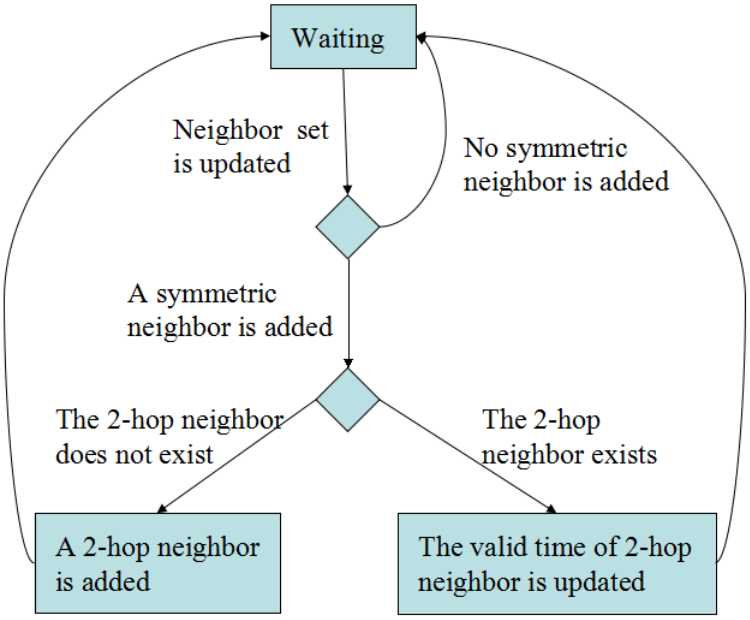

Figure 2.7: The flow diagram of 2-hop neighbor information updating

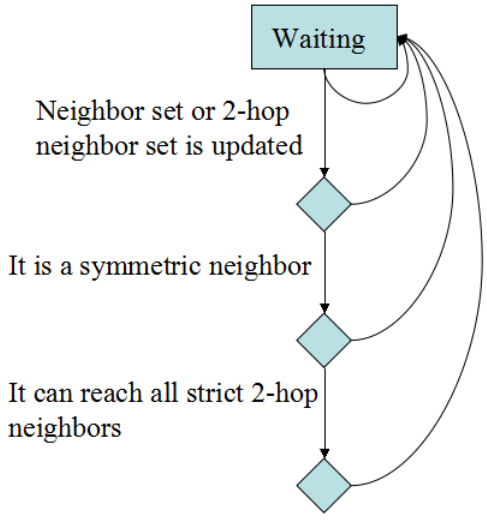

Figure 2.8: The flow diagram of MPR selection

2.6 Topology Discovery

Topology discovery depends on neighbor detection and MPR selection. The results of neighbor detection are distributed throughout the entire network by an optimized flooding mechanism with MPRs, which makes OLSR scalable for large-scale MANETs. Each node encapsulates its link set to the TC message and broadcast it. MPRs flood TC message in order to let every node in MANETs obtained consistent and up-to-date network topology information.

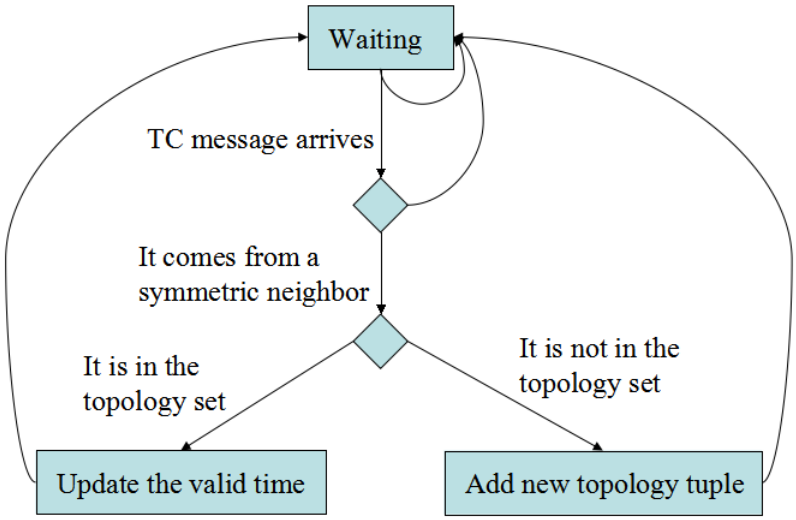

Figure 2.9: The flow diagram of topology information updating

The topology set will be updated, when a new TC message comes. First, it checks whether the last hop is a symmetric neighbor or not. If its not, discard the message. If so, it will check whether the topology tuple is already in the topology set or not. If so, the valid time is updated. If its not, a new topology tuple will be added. The flow diagram of topology information updating is shown in Figure2.9.

2.7 Route Computing

After obtaining network topology, the shortest path algorithm can be used to calculate the routing table for all possible destination. Dijkstra's algorithm is one of methods to get the shortest path. In this thesis, we develop a multi-path game theory

routing to deal with the problem of routing in jamming environment. It employs OLSRs link sensing, neighbor detection and topology discovery. Compare with single-path OLSR, it is a multi-path routing protocol. Besides, the routing computing algorithm is develop based on game theory.

Chapter 3

GAME THEORY ROUTING

The performance of OLSR is highly degraded in jamming environment due to the fact that it is not able to fast detect and fast react to the presence of jammer, which causes the intermittent of communications. Game theory routing solves this problem by computing the routes that are less likely caught by the jammer. The purpose of this section is to describe the basic theory and implementation of the Game Theory Routing, which is designed to compute routing in jamming environment.

3.1 Game Theory

An introduction of game theory and its applications is given. Zero-sum game is presented with an example. Finally, the method for solving a zero-sum game is introduced.

3.1.1 Introductions and Applications

Game theory studies the decision making processing in the multiple player games[21]. In a game, each player chooses its own strategies or actions. The outcomes of the game depends on what kinds of strategies chosen by all players. There are four key elements of a game. First, the total number of players in the game. Second, the descriptions of all strategy can be utilized in the game. Third, the descriptions of all informations used in the decision making processing. Fourth, the outcomes for each player for all possible strategies other players have chosen. Game theory has been applied to wide range of areas. In economics, it is used to model the competition among business entities. Also the voting system can be depicted by game theory model. In sociology, Game theory help researcher to explain the behaviors of people in a social

network. In biology, some of the phenomenas in the processing of evolutionary can be describe by game theory. Generally, there are three kinds of games: cooperative game, non-cooperative game and hybrid game. Cooperative game is played by several groups of team. In each team, team members collaborate with each other to achieve the goals. The decision making processing is collective. On the other side, non-cooperative game is the game among individuals. Everyone has their own strategies and make their own decisions independently. In a hybrid game, it contains both of cooperative games and non-cooperative games.

3.1.2 Zero-sum Game

A zero-sum game may have two players or more than two players. Each play chooses its own strategies. The most important property of zero-sum game is that, for all possible choices of strategies, the total sum of gain equals to the total sum of loss. Zero-sum games , shown in 3.1, are a special case of constant-sum games, where the sum of each players outcome is a constant. For zero-sum games, the sum is zero.

$$\sum_{i=0}^{N} u_i = c \tag{3.1}$$

where N is the total number of players and u_i is the outcome of each player.

For example, there are two players and two strategies. All possible outcomes are shown in Table 3.1. From that table, for all possible chooses strategies, the gain of one play alway equals to the loss of the other play. For instance, the entry in the first row and second column is $-20, 20$, which means, in the case that player1 chooses strategy2 while player2 chooses strategy1, the outcome of player1 is -20; the outcome of player2 is 20.

Table 3.1: The Outcomes of a Zero-sum Game

	strategy1	strategy2
strategy1	10, -10	-20, 20
strategy2	-30, 30	40, -40

3.1.3 Solving Zero-sum Game

The method of solving zero-sum game will be presented with the following example. Consider a two-player zero-sum game. Assume player2 can predict which strategy player1 will choose and take action with the most profitable strategy. The gain matrix of player1 is shown in Table 3.2.

Table 3.2: The Gain Matrix of Player1

	strategy1	strategy2	strategy3
strategy1	10	20	10
strategy2	-10	-10	10
strategy3	10	-20	-10

If player1 chooses strategy1, player2 will pick up the strategy with maximum profits, which is strategy1 or strategy3. If player1 chooses strategy2, player2 will pick up strategy1 or strategy2 with 10 profits. If player1 chooses strategy3, player2 will pick up strategy2 with 20 profits. Let $u_{i,j}$ be the j^{th} element in i^{th} row of the gain matrix. For player1, the best strategy is to choose the strategy with maximum profits among minimum profits of each strategy. The profits of that strategy is formulated in 3.2. This value is defined as pure-strategy security level. For player2, it is formulated in 3.3. The pure-strategy security level is -10, when strategy1 or strategy3 is chosen.

$$u_{play1} = max_i min_j \; u_{ij} \tag{3.2}$$

$$u_{play2} = max_j min_i \; - u_{ij} \tag{3.3}$$

Lets consider another case. Assume both players get more informations during decision making processing. Specifically, they know the probability of each strategy their opponent will choose. For player1, suppose he or she has n strategies, the probability vector for each strategy is $\mathbf{p} = (p_1, ..., p_n)$. For player1, suppose he or she has m strategies, the probability vector for each strategy is $\mathbf{q} = (q_1, ..., q_m)$. If player1 chooses strategy1, the expectation value of player1s gain is $\sum_{j=1}^{m} q_j u_{1,j}$. The most secure strategy for player1 is in 3.4. The most secure strategy for player2 is in 3.5.

$$
\begin{aligned}
u_{play1} &= max_{\mathbf{p}} min_{\mathbf{q}} \sum_{i=1}^{n} \sum_{j=1}^{m} p_i q_j u_{i,j} \\
&= max_{\mathbf{p}} min_{\mathbf{q}} \mathbf{p} \cdot U \cdot \mathbf{q}
\end{aligned} \tag{3.4}
$$

$$
\begin{aligned}
u_{play2} &= max_{\mathbf{q}} min_{\mathbf{p}} \sum_{j=1}^{m} \sum_{i=1}^{n} -p_i q_j u_{i,j} \\
&= max_{\mathbf{q}} min_{\mathbf{p}} - \mathbf{p} \cdot U \cdot \mathbf{q} \\
&= -min_{\mathbf{q}} max_{\mathbf{p}} \mathbf{p} \cdot U \cdot \mathbf{q}
\end{aligned} \tag{3.5}
$$

In order to solve this zero-sum game, the equation 3.6 must be solved. In this solution, each play choose its own security level strategy.

$$
\begin{aligned}
u_{play1} - u_{play2} &= 0 \\
u_{play1} &= u_{play12} \\
max_{\mathbf{p}} min_{\mathbf{q}} \mathbf{p} \cdot U \cdot \mathbf{q} &= min_{\mathbf{q}} max_{\mathbf{p}} \mathbf{p} \cdot U \cdot \mathbf{q}
\end{aligned} \tag{3.6}
$$

This is called minimax problem can be solved by linear programming, as shown in 3.7 and 3.8.

$$
\begin{aligned}
&max\ u_{play1} \\
&subject\ to \\
&\mathbf{p} \cdot U - u_{player1} \cdot I \geq 0 \\
&\mathbf{p} \geq 0,\ \mathbf{p} \cdot I = 1
\end{aligned} \tag{3.7}
$$

$$
\begin{aligned}
&min\ u_{play2} \\
&subject\ to \\
&\mathbf{p} \cdot U - u_{player2} \cdot I \leq 0 \\
&\mathbf{q} \geq 0,\ \mathbf{q} \cdot I = 1
\end{aligned} \tag{3.8}
$$

3.2 Game Theory Routing

In this section, the model of data routing over a network is introduced. The game between router and attacker is presented.

3.2.1 Routing over Networks

To begin with, the key element of network routing is listed below.

- **Nodes:** The network consists of N nodes.
- **Links:** The link from node i to node j is denoted as $l_{i,j}$.
- **Data Rate:** The data rate of the link from node i to node j is denoted as $r_{l_{i,j}}$.
- **Transmission Time:** The transmission time of the link from node i to node j is denoted as $t_{l_{i,j}}$.
- **Path:** The path from node i to node j is denoted as $Path_{i,j} = \{l_{i,1}, l_{1,2}, \ldots, l_{k,j}\}$, which contains a set of connected links.

Routing from node i to node j is defined as finding $Path_{i,j}$ based the set of links in the network.

3.2.2 Router-attacker Game

There are two players in the game: router and jammer. The router intends to find a path in order to transmit data through it. The jammer intends to jam the path in order to intercept the data transmission. That game can be treated as a zero-sum game[22], in which the router aim to minimize the data transmission time while the jammer aim to maximize that time. Assume the game is off-line game, which means

the jammer select a set of links to jam before the data transmission; the router does not know which links are going to be jammed. The objective function for both of the jammer and the attacker is in 3.9. The router intend to minimize objective function. On the other hand, the jammer intend to maximize it. The router is more partial to shorter path, as the increase of ϵ. If $\epsilon = 0$, all path are the same to the route. If $\epsilon \to \infty$, the router will choose the shortest path.

$$J(\epsilon) = \begin{cases} (1+\epsilon)^{t-1}, \ t^{th} \text{ hop is jammed} \\ 0, \ \text{otherwise} \end{cases} \tag{3.9}$$

The probability of routing from node i to node j is denoted as $p_{i,j}$. One unique feature of game theory routing is that the route is not only depends on the destination, but also depends on the source. For node i, with a contain pair of a source and a destination, the probability $p_{i,j}$ satisfies the principle in 3.10. L is the set of all links of the network.

$$\sum_{l_{i,k} \in L} p_{i,k} = 1 \tag{3.10}$$

For the router, a set of routing strategies is defined as 3.11.

$$P = \left\{ p_l : \sum_{l_{i,k} \in L} p_{i,k} = 1, \forall i \in N, l \in L \right\} \tag{3.11}$$

For the jammer, a set of jamming strategies is defined as 3.12.

$$Q = \left\{ q_l : \sum_{l \in L} q_l = 1, l \in L \right\} \tag{3.12}$$

The security level strategy of a router is 3.13.

$$u_{jammer} = min_P max_Q J(\epsilon) \tag{3.13}$$

The security level strategy of a jammer is 3.14.

$$u_{router} = min_Q max_P - J(\epsilon)$$
$$= -max_Q min_P J(\epsilon) \tag{3.14}$$

It has been proved in [23] that the solution of this problem is 3.15

$$J^*(\epsilon) = \frac{1}{max_\mu max_{x_l \in [0,1], A(\epsilon)x + \mu c = 0} \mu}$$
$$p^*_{l_{i,j}} = \frac{x^*_{l_{i,j}}}{\sum_{l_{i,k} \in L} x^*_{l_{i,k}}},\ l_{i,j} \in L \tag{3.15}$$

$A(\epsilon)$ is an appropriately defined matrix, and c is an appropriately defined vector. When $\epsilon = 0$, the security level strategy for router is to maximize the flow from a certain pair of the source and destination. At this point, solving this two-player zero-sum game is equivalent to solving maximum flow problem.

3.2.3 Solving Maximum Flow

Maximum flow problem is defined as designing a network in order to maximize the flow from the source to the destination[24]. In a maximum flow problem, there are only one source and one destination. There is no flow coming into the source. The source only outputs flows. There is no flow coming out of the destination. The destination only absorbs flows. Maximum flow problem has three important constrains. The total amount of output flow from the source must equal to the total amount of input flow into the destination. The flow on each arc cannot excess the capacity of that arc. At each node, the total amount of input flow must equal to the total amount of output flow.

An example is given to demonstrate how to solve maximum flow problem via linear programming. The network is illustrated in Figure 3.1. In this example, there are seven nodes in the network. The number on the link is its capacity. The source is node1. The destination is node6. The maximum flow problem is to design that

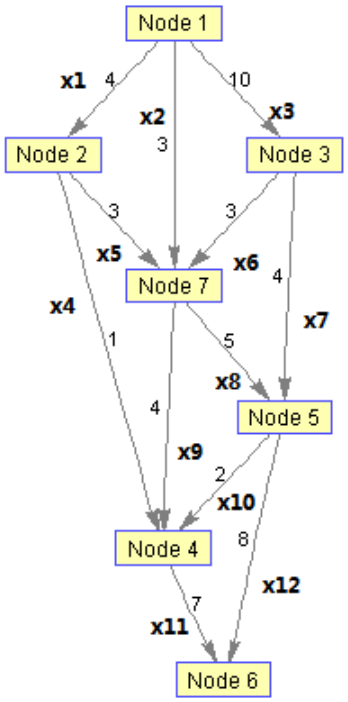

Figure 3.1: The network for maximum flow

network so that the total amount of output flow from node1 and the total amount of input flow into node6 are maximized.

The objective function is 3.16, which means to maximize the total amount of output flow from node1.

$$f = x_1 + x_2 + x_3 \tag{3.16}$$

All of the capacity of a link must be greater than or equal to zero, as shown in 3.17.

$$x_i \geq 0,\ i = 1...12 \tag{3.17}$$

All of the weight of a link must be greater than or equal to the capacity of that link, as shown in 3.18.

$$\begin{aligned} x_1 \leq 4,\ x_2 \leq 3,\ x_3 \leq 10,\ x_4 \leq 1,\ x_5 \leq 3,\ x_6 \leq 3 \\ x_7 \leq 4,\ x_8 \leq 5,\ x_9 \leq 8,\ x_{10} \leq 2,\ x_{11} \leq 7,\ x_{12} \leq 8 \end{aligned} \tag{3.18}$$

The total amount of output flow node1 must equal to the total amount of input flow into node6, , as shown in 3.19.

$$x_1 + x_2 + x_3 = x_{11} + x_{12} \tag{3.19}$$

For each link, the flow on it must be smaller than the capacity of it, as shown in 3.20.

$$\begin{aligned} x_1 \leq 4,\ x_2 \leq 3,\ x_3 \leq 10,\ x_4 \leq 1,\ x_5 \leq 3,\ x_6 \leq 3 \\ x_7 \leq 4,\ x_8 \leq 5,\ x_9 \leq 8,\ x_{10} \leq 2,\ x_{11} \leq 7,\ x_{12} \leq 8 \end{aligned} \tag{3.20}$$

For each node, the total amount of input flow must equal to the total amount of output flow, as shown in 3.21.

$$\begin{aligned} x_1 = x_4 + x_5,\ for\ node2 \\ x_{10} = x_3 + x_4,\ for\ node3 \\ x_4 + x_9 + x_{10} = x_{10},\ for\ node4 \\ x_7 + x_8 = x_{10} + x_{12},\ for\ node5 \\ x_2 + x_5 + x_6 = x_8 + x_9,\ for\ node7 \end{aligned} \tag{3.21}$$

This is a linear programming problem. There are various algorithms to solve this problem. Simplex algorithm is a typical one[24]. The result is shown in Figure 3.2.

3.3 Implementation of Game Theory Routing

The performance of OLSRv2 is highly degraded in jamming environment due to the fact that it is not able to fast detect and fast react to the presence of jammer, which causes the intermittent of communications. Game theory routing solves this problem by computing the routes that are less likely caught by the jammer.

3.3.1 Concepts and Techniques

Game theory routing protocol consists of a three major functions. The first part is topology information gathering from OLSRv2. The seconde part is route computing. The third part is probabilistic routing based on the probability forwarding table.

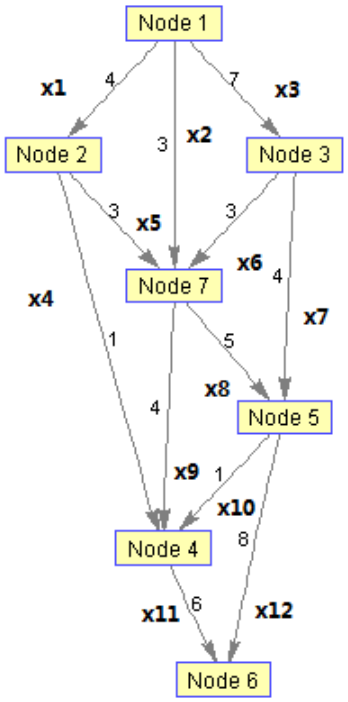

Figure 3.2: The solution of maximum flow

Three critical concepts are needed to be explained. First, jamming probability is the probability that the link is caught by the jammer. Second, probability forwarding table is the able consists of destination, source, next hop, and jamming probability. Third, probabilistic routing means that each pair of destination and source has several routing entries with different next hops and jamming probabilities. Probabilistic routing determines the next hop by generating a random number and comparing it with jamming probabilities of next hops.

Game theory routing is implemented in Qualnet 5.0.2. The development environment is Microsoft Visual Studio 2008. Game theory routing is built upon OLSRv2. GNU Linear Programming Kits 4.52 is employed the development.

3.3.2 System Architecture

Figure 3.3 depicts the high-level system architecture. The main components of the system are listed below.

- **Jamming Detector:** The module for jamming probability estimation.
- **Jamming Probability:** The module stores jamming probability.
- **OLSRv2:** The module of OLSRv2 routing protocol.
- **Game Theory Routing:** The module for computing probability forwarding table.
- **Probability Forwarding Table:** The module stores probability forwarding table.

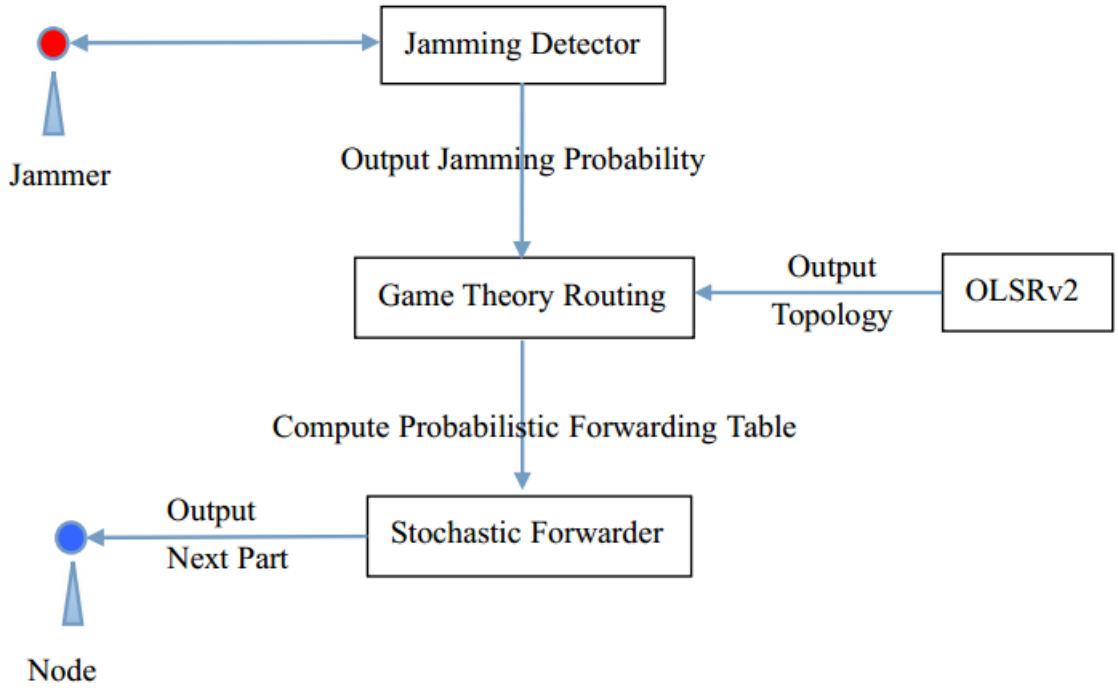

Figure 3.3: The system architect of game theory routing

3.3.3 System Operations

The flow diagram of a typical sequence of system operations that occur during jamming is shown in Figure 3.4. Each number on it is explained below.

- 1. Execute OLSRv2 topology discover
- 2. Write topology information

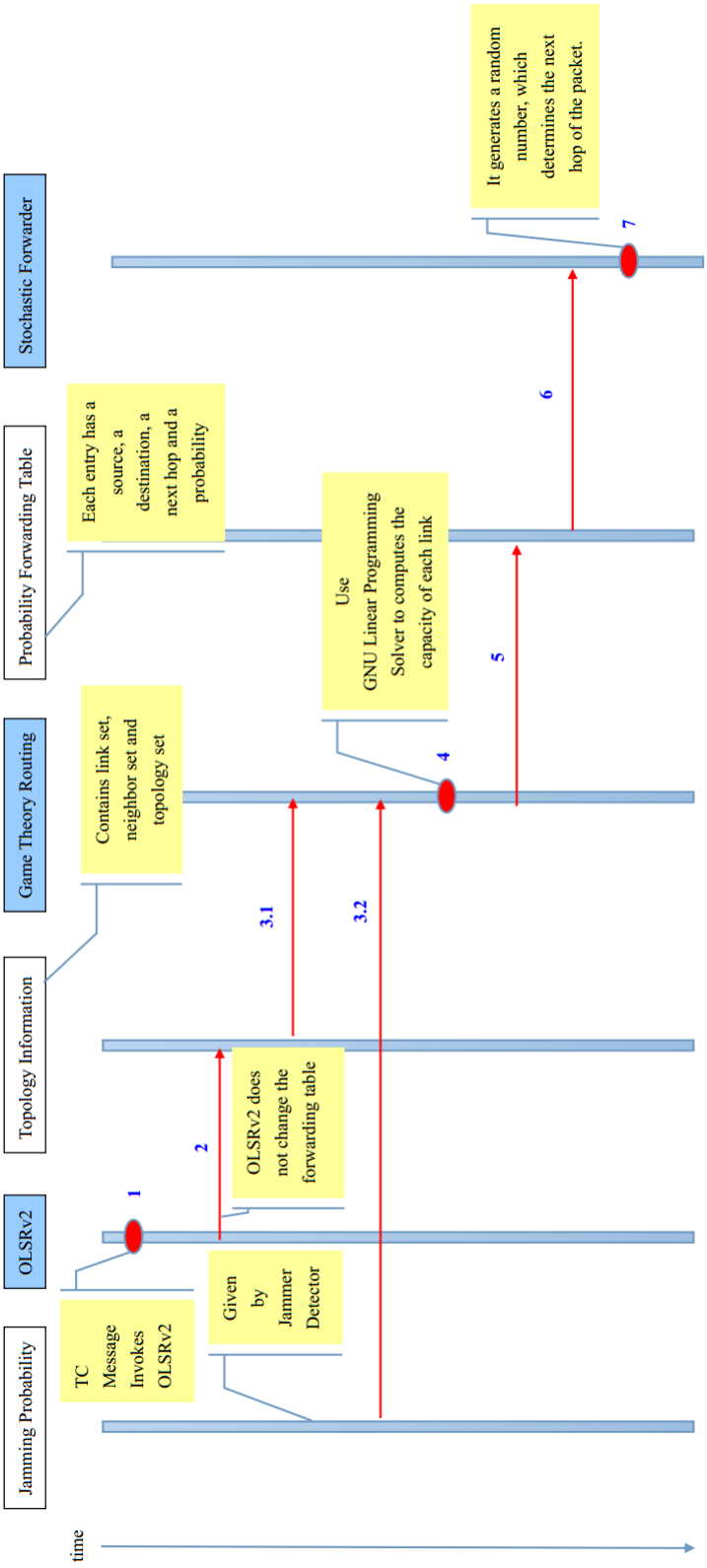

Figure 3.4: The flow diagram of system operations

- 3.1. Read topology information
- 3.2. Read jamming probability
- 4. Execute game theory routing
- 5. Write probability forwarding table
- 6. Read probability forwarding table
- 7. Execute stochastic forwarding
- 8. Write next hop

The flow diagram of a typical sequence of probabilistic forwarder subsystem operations that occur during jamming is shown in Figure 3.5. Each number on it is explained below.

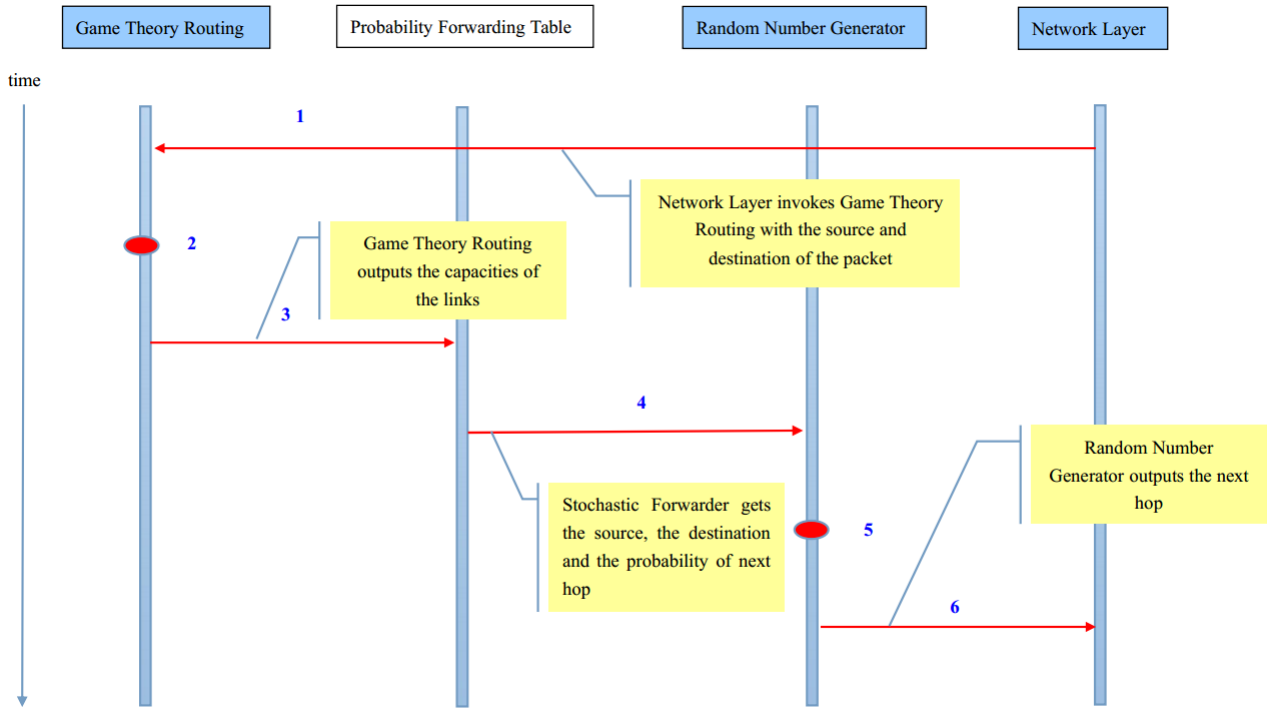

Figure 3.5: The flow diagram of probabilistic forwarder subsystem operations

- 1. Invoke Game Theory Routing
- 2. Execute game theory routing

- 3. Write probability forwarding table
- 4. Read probability forwarding table
- 5. Execute Random Number Generator
- 6. Read the outputted next hop

3.3.4 System Flow Chart

The system flow chart is shown in Figure 3.6. Each step on it is explained below.

- Step 1: OLSRv2 gets the jamming probability from the jammer detector.
- Step 2: OLSRv2 outputs the topology information to the Game Theory Routing without changing the forwarding table.
- Step 3: Game theory routing gets the topology information from OLSRv2and jamming probability invoking the linear programming solver. The solver computes the capacity of each link and output them to the probabilistic forwarding table.
- Step 4: Stochastic Packet Forwarder reads the probabilistic forwarding table and determines the next hop.

The flow chart of probabilistic forwarder subsystem is shown in Figure 3.7. Each step on it is explained below.

- Step 1: Network layer invoke the game theory routing and outputs the source and the destination of the packet.
- Step 2: The Game Theory Routing outputs the packet loss probability to the the stochastic forwarder.
- Step 3: The stochastic forwarder reads the packet loss probability, the source of the packet and the destination of the packet.
- Step 4: The random number generator generates a random number, which determines the next hop and outputs it to network layer.

3.3.5 Use Case

In this section, two use cases are provided to demonstrate the application of game theory routing.

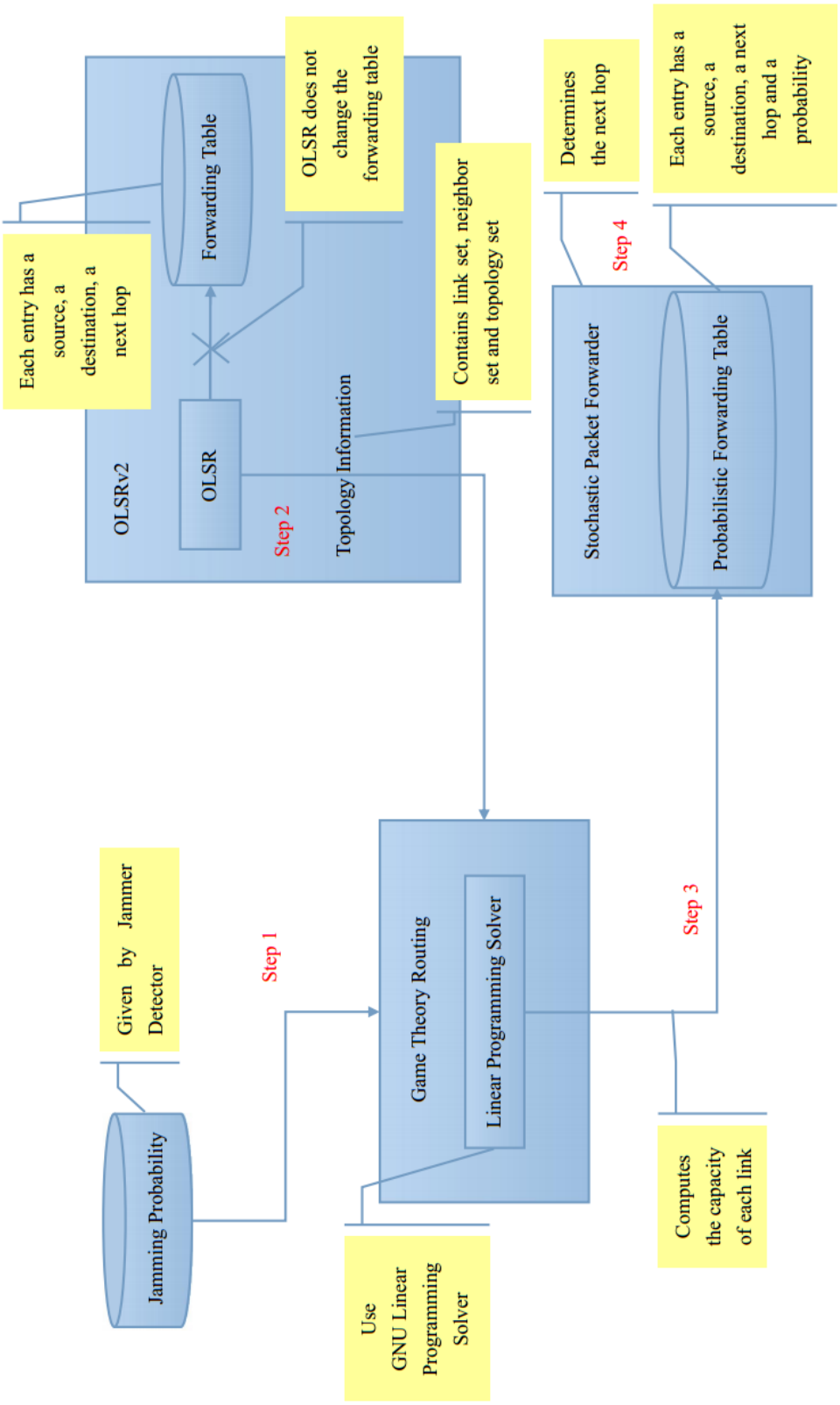

Figure 3.6: The system flow chart

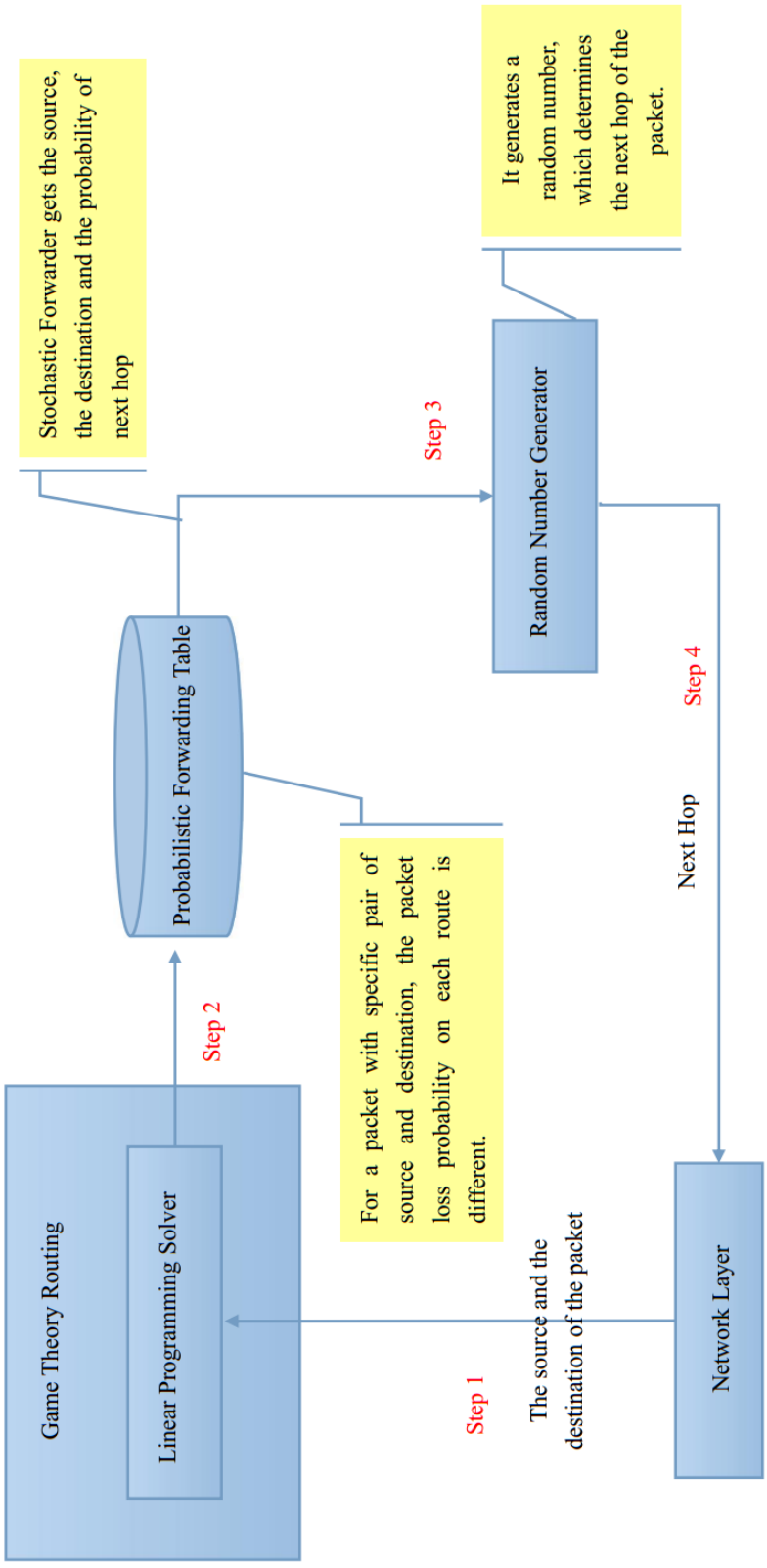

Figure 3.7: The flow chart of probabilistic forwarder subsystem

3.3.5.1 Case 1

The ad hoc network consists of ten nodes, one source, one destination and several intermediary nodes, as what is shown in Figure /reff19. The source tries to transmit data to the destination. At this time, a jammer tries to cut the transmission via random emitting high power over the communication band. Suppose, the source and the destination communicate through the upper link, before the presence of the jammer. When the jammer presents, the communication will be cut for a while, since the conventional routing protocol cannot fast react to the jamming.

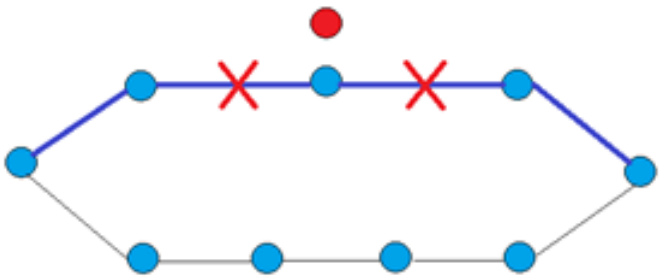

Figure 3.8: The network of case 1.

Game theory routing takes advantages of the fast jamming detection. Specifically, it knows the probabilities of links being caught by the jammer. In this scenario, the nodes are able to predict the probabilities of the presence of the jammer in advance. There are two links in the upper route, whose jamming probability is 0.8, which means a jammer is likely to present near there, as what is shown in Figure 3.9. Based on the jammer probabilities, our routing protocol will compute a new route based on game theory, which chooses a route that is least likely caught by the jammer. In this scenario game theory routing will choose the lower links for the transmission, since the probabilities of the lower links being caught is much lower than the upper links. The throughput of link is calculated by game theory routing, as what is shown in Figure 3.10.

In sum, in order to preserve the data communication in the presence of jamming,

game theory routing will swiftly compute a new route that is not affected by the jammer, when the jammer presents.

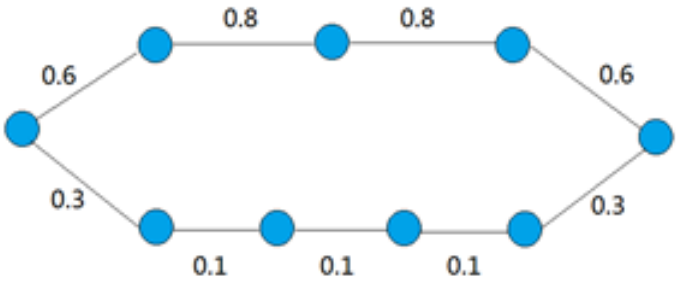

Figure 3.9: The jamming probabilities of each link.

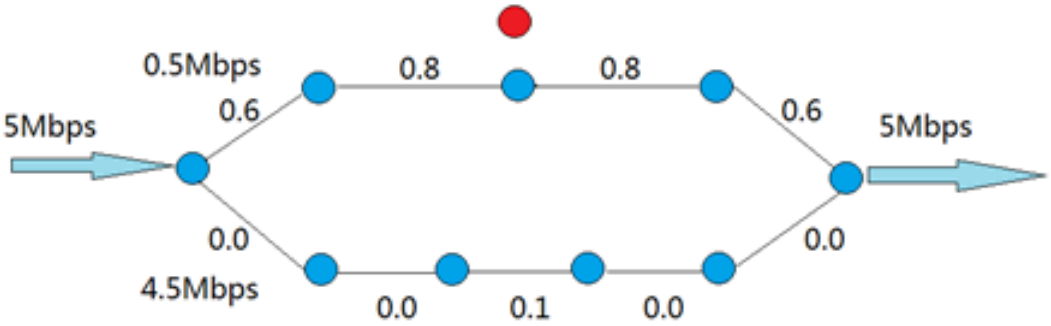

Figure 3.10: The throughput calculated by game theory routing.

3.3.5.2 Case 2

Game Theory Routing divided flow by the probability of the links. In Case 2, shown in Figure 3.11,the total input flow is 10Mbps, which equals to the total output flow. There are four paths. Three of them are on the upper side. And one of them is on the lower side. The sum of probabilities of the upper three paths is 1.5. The sum of probabilities of the lower one path is 0.5. The flow is divided proportionally to the probability. As a result, the flow of the upper three paths is 7.5Mbps. The flow of the lower one path is 2.5Mbps.

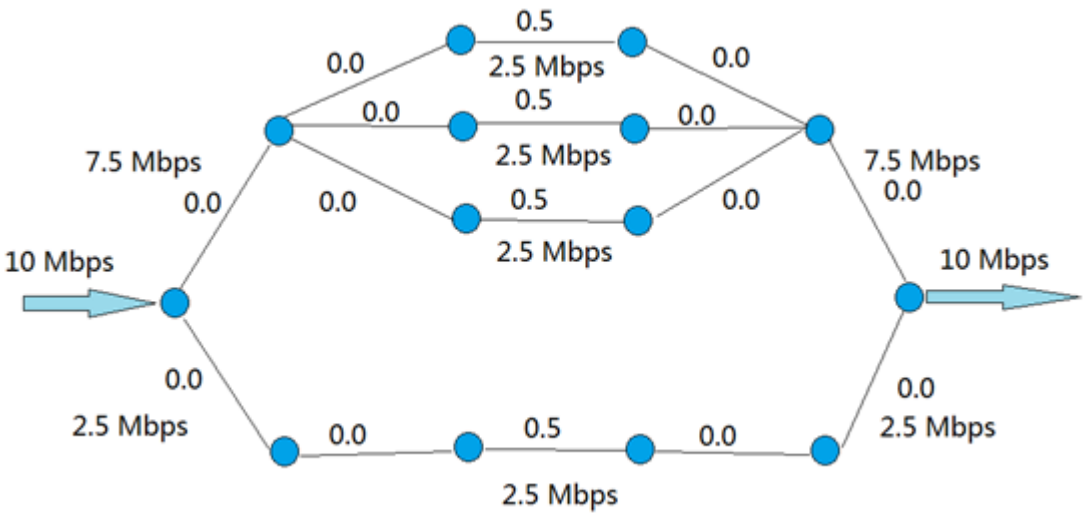

Figure 3.11: The jamming probability and the throughput calculated by game theory routing.

Chapter 4

SIMULATIONS

In this section, the simulation studies of the performances of the jammer and game theory routing are presented. Qualnet 5.0.2 is utilized as the simulation platform. The following metrics are considered in the simulation experiments. Packet delivery ratio is the ratio of the total number of packets originated by the source and the total number of packets received by the destination. It describes the loss probability of the network.Average end-to-end delay is the average length of the time interval between a packet sending from the source and that packet receiving at the destination. This delay includes transmission delay, propagation delay, processing delay and queuing delay.

4.1 Performance of Jammer

4.1.1 Simulation Parameters

Table 4.1 presents main simulation parameters for the study of the performance of the jammer.

4.1.2 Simulation Scenarios

Figure 4.1 shows the simulation scenarios. Node1 is the source. Node12 is the destination. There are two paths, which are upper path and lower path. Nodes on different path cannot communicate with each other. Each node is only able to transmit data with there one-hop neighbors. The jammer can isolate node5 from the network. However, it does not affect other nodes in the network.

Table 4.1: The Simulation Parameters

Parameter	Value
Number of Nodes	12
Simulation Time	300sec
Simulation Area	5000*1500m
Number of Channel	1
Pathloss Model	Free Space
PHY Layer	IEEE 802.11a
Data Rate	12Mbps
Antenna Model	Omnidirectional
MAC Layer	IEEE 802.11a
Network Layer	IPv4
Routing Protocol	OLSRv2 NIGATA
Application	CBR

4.1.3 Constant Jammer

Constant jammer keeps emitting signal over the networks operation band. The tracing data of the network is summarized below. Suppose jammer start to jam at 20 second.

- [0, 20]s: CBR packets were sending through the path 1-2-3-4-5-6-12.
- [20, 28]s: CBR packets were sending through 1-2-3. Node3 dropt the packets and sent ICMP packets back through 3-2-1.
- [28, 33]s: CBR packets were sending through 1-2-3. Node3 sent it back to node2. And node2 sent it to node3 again. Packets were sending between node2 and node3 several times. Finally, node3 dropt the packet and sent ICMP packets back through 3-2-1.
- [33, -]s: After 33 second, the network switched path. The packets were sending through 1-7-8-9-10-11-12.

In conclusion, with constant jammer, the network will be blocked in the first 13 secondes, after the start of the jamming. After 13 secondes, it will find a new route and transmit data over it. Therefore, in this case, the time needed to switch path is 13 secondes.

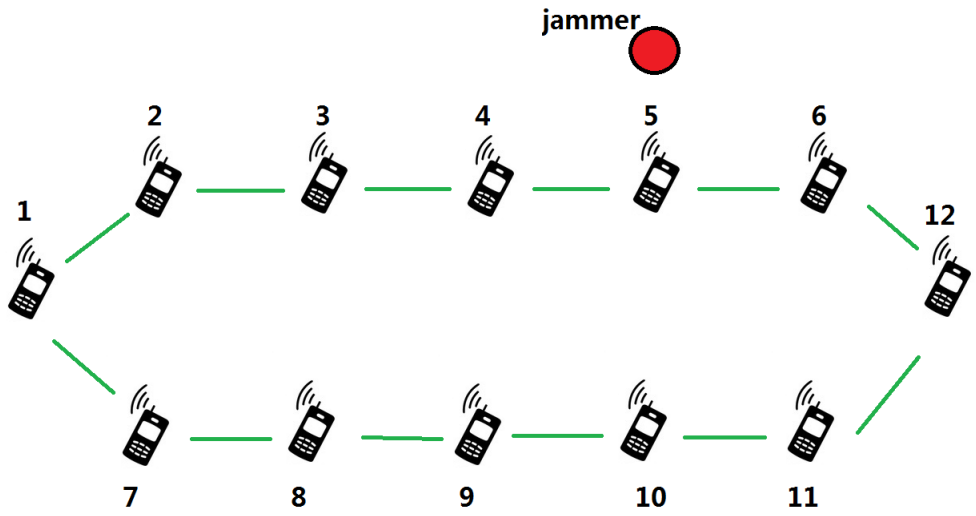

Figure 4.1: Simulation scenarios

4.1.4 Pulse Jammer

Pulse jammer emits a pulse signal over the networks operation band. The width of the pulse or the jamming period varies. The effects of pulse jammings with different width are simulated. The tracing data of the network is summarized below. Suppose jammer start to jam at 20 second. There are three case. First, the width varies from 1 second to 5 second. 3 second is chosen as a typical value. Second, the width varies from 6 second to 10. 7 second is chosen as a typical value. Third, the width varies from 11 second to infinity. 30 second is chosen as a typical value. In each case, the responses of the network are similar. Three typical values are selected to show the networks response for each case.

The tracing data of the first case is summarized below.

- [0, 20]s: CBR packets were sending through the path 1-2-3-4-5-6-12.
- [20, 23]s: CBR packets were sending through 1-2-3. Node3 dropt the packets and sent ICMP packets back through 3-2-1.
- [23, -]s: CBR packets were sending through the path 1-2-3-4-5-6-12.

In conclusion, in this case, the network wont realize the existence of the jammer. It kept transmitting over one path.

The tracing data of the second case is summarized below.

- [0, 20]s: CBR packets were sending through the path 1-2-3-4-5-6-12.
- [20, 27]s: CBR packets were sending through 1-2-3. Node3 dropt the packets and sent ICMP packets back through 3-2-1.
- [27, 32]s: CBR packets were sending through 1-2-3. Node3 sent it back to node2. And node2 sent it to node3 again. Packets were sending between node2 and node3 several times. Finally, node3 dropt the packet and sent ICMP packets back through 3-2-1.
- [32, 37]s: The network switched path. The packets were sending through the path 1-7-8-9-10-11-12.

  [37, -]s: The network switched path. The packets were sending through the path 1-2-3-4-5-6-12.

In conclusion, in this case, the network detected the disconnected links caused by the jammer. It switched the path after jammer period. And it finally got back to the original path.

The tracing data of the third case is summarized below.

- [0, 20]s: CBR packets were sending through the path 1-2-3-4-5-6-12.
- [20, 28]s: CBR packets were sending through 1-2-3. Node3 dropt the packets and sent ICMP packets back through 3-2-1.
- [28, 33]s: CBR packets were sending through 1-2-3. Node3 sent it back to node2. And node2 sent it to node3 again. Packets were sending between node2 and node3 several times. Finally, node3 dropt the packet and sent ICMP packets back through 3-2-1.
- [33, 50]s: The network switched path. The packets were sending through 1-7-8-9-10-11-12.
- [50, 63]s: The network switched path. The packets were sending through the path 1-2-3-4-5-6-12.

In conclusion, in this case, the network detected the disconnected links caused by the jammer. Because the jamming time is long enough for the network to switch to the other path within jamming period. After jamming period, it took 13 seconds for the network to switch back to the original path.

4.1.5 Periodic Jammer

Periodic jammer emits pulse signal over the networks operation band periodically. For example, the jamming schedule shown in Figure 4.2 is an particular periodic jamming. The jammer period, the red block in Figure 4.2, is 4 seconds; and the idle period is 2 seconds.

Figure 4.2: The jamming schedule of a periodic jammer

The simulation results of periodic jamming is shown in Figure 4.3. The jamming time varies from 1 seconds to 10 seconds. The idle time varies from 1 seconds to 15 seconds.

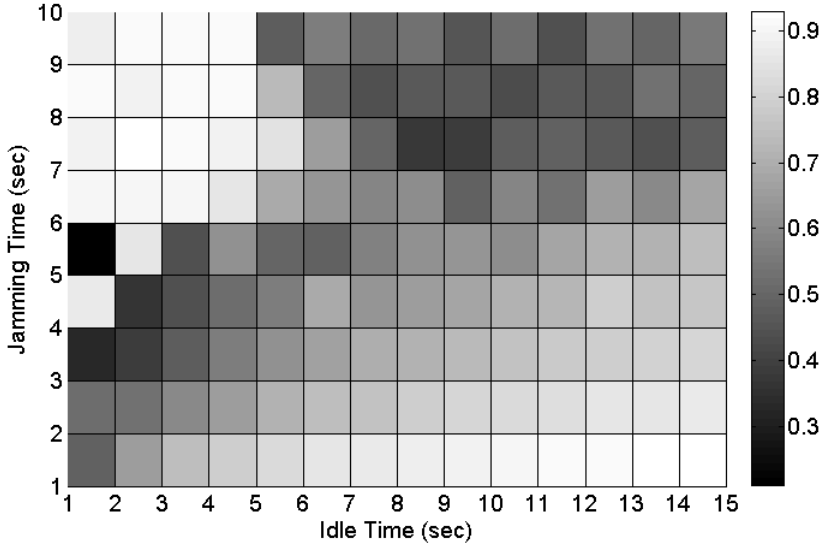

Figure 4.3: The packet delivery rate of periodic jammer

An intelligent jammer does not make the network switch path during jamming period. There are two different jamming strategies. One is fast periodic jamming. The other one is slow periodic jamming. The design of fast periodic jamming is explained first. According to simulation results of pulse jammer, the network will not realize the existence of the jammer, if the jamming period is shorter on 6 seconds. No packets

can go through the network during jamming. After jamming, packets can be believed immediately. The different between pulse jammer and periodic jammer is that pulse jammer never cares about the idle time. However, the length of idle time is very important to periodic jammer. Because short idle time lets the network detect the jammer and switch path. But long idle time makes the jammer inefficient. Therefore, finding the optimal idle time is a crucial task. The criterion is to minimize the idle time subject to the condition that the network is not able to switch path. The results are shown in Table 4.2 and Figure 4.4. From these results, the jammer with 5 seconds jamming time and 1 seconds idle time is the most optimal design for fast periodic jammer.

Table 4.2: The Results for Fast Periodic Jammer

Jam Time(sec)	**Idle Time(sec)**	**Packet Delivery Rate(%)**
1	1	48
2	1	52
3	1	32
4	2	36
5	1	21

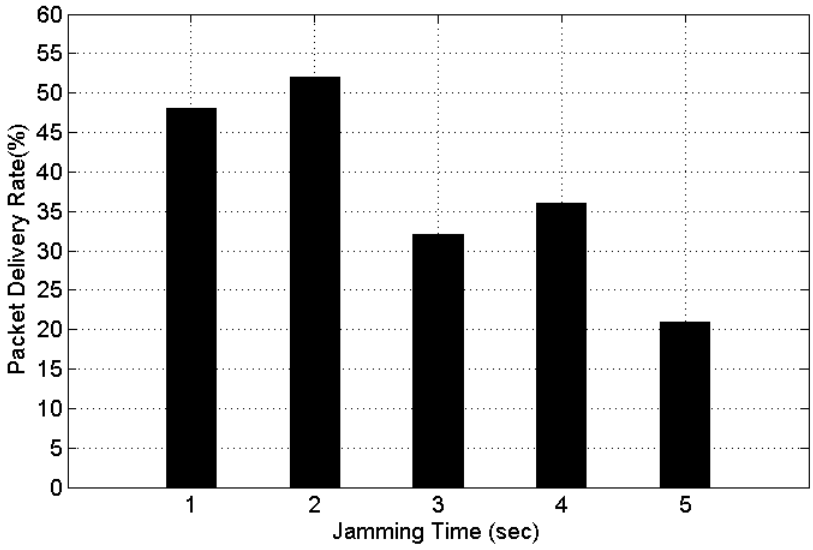

Figure 4.4: The results of fast periodic jammer

The design of slow periodic jamming will be explained in the followings. According to simulation results of pulse jammer, the network will realize the existence of the jammer, if the jamming period is longer than 6 seconds. No packets can go through the network during jamming. After jamming, the network is blocked for a short period of time. And then a new path will be utilized. Finally, it will switch back to use the original path. If the idle time is too short, the network will use a new path. So, it must be long enough to let the network switch back to use the original path. However, it cannot be too long. Because it makes the jammer inefficient. To design an optimal slow period jammer, the criterion is to minimize the idle time subject to the condition that the network is able to switch back to the original path. The results are shown in Table 4.3 and Figure 4.4. In conclusion, the slow periodic jammer is not efficient, since it requires for a relatively long idle time compared with that of fast periodic jammer.

Table 4.3: The Results of Slow Periodic Jammer

Jam Time(sec)	**Idle Time(sec)**	**Packet Delivery Rate(%)**
6	10	49
7	9	37
8	11	42
9	12	44
10	14	53

4.2 Performance of Game Theory Routing

In this section, the comparative simulation studies of the performance of game theory routing and that of the OLSR are presented. Qualnet 5.0.2 is utilized as the simulation platform. Packet delivery ratio and average end-to-end delay are the metrics considered in the simulation experiments. Table 4.1 presents main simulation parameters. Figure 4.1 shows the simulation scenarios. Game theory routing and OLSR are tested with fast periodic jammer and slow periodic jammer. The results are shown in Table 4.4, Table 4.5, Figure 4.6 and Figure 4.7. In conclusion, game theory routing is more robust in jamming environment that OLSR. Because it has higher packet delivery ratio and comparable average end-to-end delay.

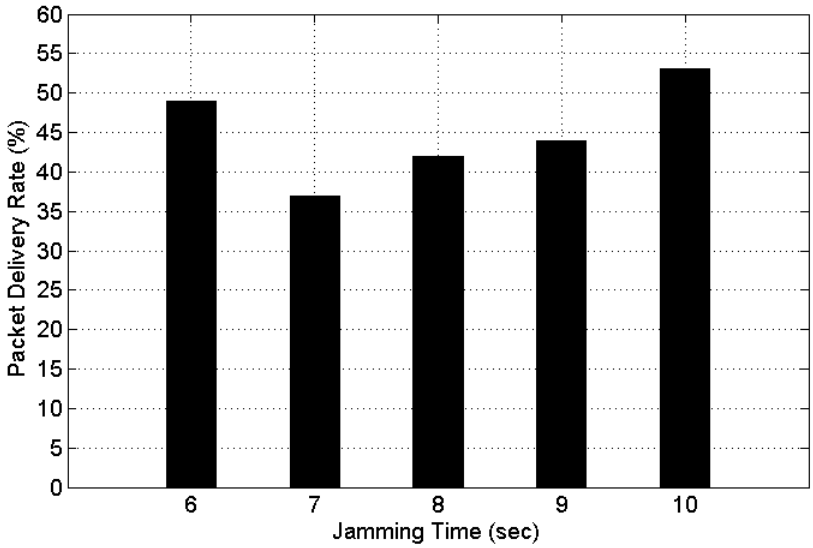

Figure 4.5: The results of slow periodic jammer

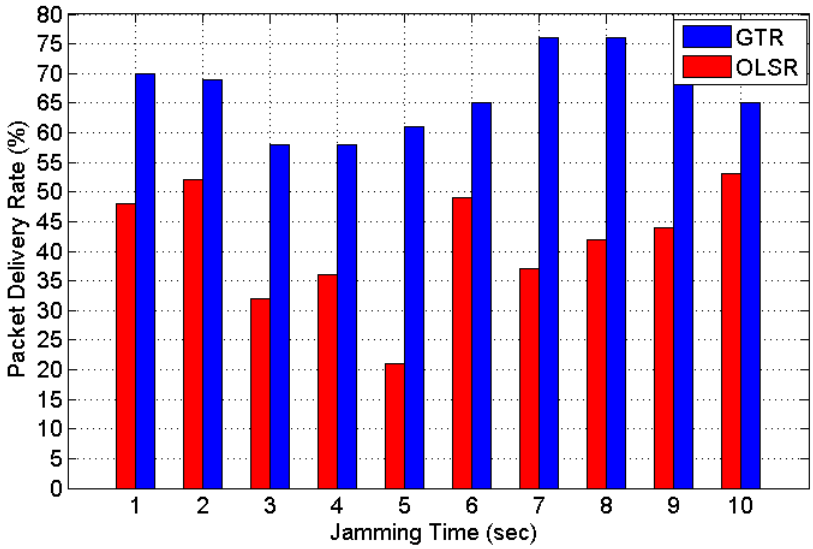

Figure 4.6: The results of periodic jammer

Table 4.4: The Packet Delivery Rate(%) for OLSR and GTR in Jamming Environment

Jam Time(sec)	**Idle Time(sec)**	**PDR(OLSR)**	**PDR(GTR)**
1	1	48	70
2	1	52	69
3	2	32	58
4	2	36	58
5	3	21	61
6	3	49	65
7	13	37	76
8	13	42	76
9	13	44	71
10	10	53	65

Table 4.5: The Average End-to-end Delay(sec) for OLSR and GTR in Jamming Environment

Jam Time(sec)	**Idle Time(sec)**	**Delay(OLSR)**	**Delay(GTR)**
1	1	0.006	0.021
2	2	0.006	0.021
3	2	0.006	0.021
4	2	0.007	0.022
5	3	0.007	0.021
7	13	0.007	0.021
8	13	0.006	0.021
9	13	0.006	0.020
10	10	0.0066	0.021

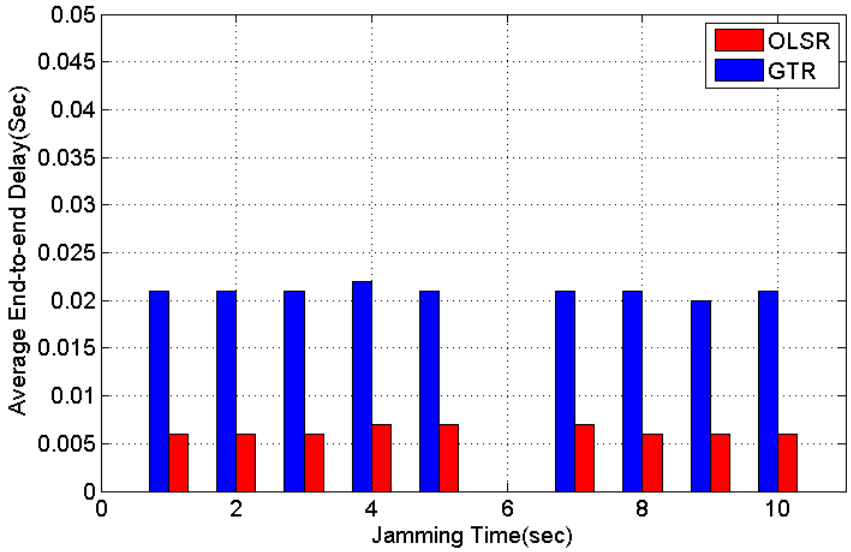

Figure 4.7: The results of periodic jammer

Chapter 5

CONCLUSION

A game theory routing protocol has been proposed in this thesis to improve the robustness of MANETs in jamming environment. Game theory routing protocol is developed based on OLSR. The problem of routing against a jammer can be view as a zero-sum game between the network and the jammer. The bast strategy can be achieved by solving the zero-sum game via linear programming. The performance of jammer is simulated in order to design a powerful jammer. With this jammer, the performance of game theory routing and OLSR is compared. The results show that game theory routing gets higher packet delivery rate and comparable average end-to-end delay.

Bibliography

[1] C. Puttamadappa S. K. Sarkar, T. G. Basavaraju. *Ad Hoc Mobile Wireless Networks: Principles, Protocols, and Applications*. Auerbach Publications, 2008.

[2] M. Ilyas. *The Handbook of Ad Hoc Wireless Networks*. CRC Press, 2003.

[3] I. Moerman C. Blondia P. Demeester B. Latre, B. Braem. A survey on wireless body area networks. *Wireless Networks*, 17:1–18, 2011.

[4] G. Mazzini A. Conti R. Verdone, D. Dardani. *Wireless Sensor and Actuator Networks*. Elsevier, 2008.

[5] M. Srivastava D. Culler, D. Estrin. Overview of sensor networks. *IEEE Comput.*, 37:4149, 2004.

[6] Bluetooth core specification version 4.0.

[7] IEEE 802.11ac: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.

[8] A. Patel R. Razali S. Mohseni, R. Hassan. Comparative review study of reactive and proactive routing protocols in manets. *Digital Ecosystems and Technologies (DEST), 2010 4th IEEE International Conference on*, pages 304 – 309, 2010.

[9] P. Jacquet, P. Mhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot. Optimized link state routing protocol for ad hoc networks. pages 62–68, 2001.

[10] Charles E. Perkins and Pravin Bhagwat. Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers. pages 234–244, 1994.

[11] David B. Johnson and David A. Maltz. Dynamic source routing in ad hoc wireless networks. In *Mobile Computing*, pages 153–181. Kluwer Academic Publishers, 1996.

[12] C. Perkins, E. Belding-Royer, and S. Das. Ad Hoc On-Demand Distance Vector (AODV) Routing, 2003.

[13] Douglas S. J. De Couto, Daniel Aguayo, John Bicket, and Robert Morris. A high-throughput path metric for multi-hop wireless routing, 2003.

[14] Richard Draves, Jitendra Padhye, and Brian Zill. Routing in multi-radio, multi-hop wireless mesh networks. In *In ACM MobiCom*, pages 114–128. ACM Press, 2004.

[15] Kannan Srinivasan and Philip Levis. RSSI is Under Appreciated. In *In Proceedings of the Third Workshop on Embedded Networked Sensors (EmNets*, 2006.

[16] Alec Woo, Terence Tong, and David Culler. Taming the underlying challenges of reliable multihop routing in sensor networks. In *In SenSys*, pages 14–27. ACM Press, 2003.

[17] Bor rong Chen, Kiran kumar Muniswamy-reddy, and Matt Welsh. Ad-hoc multicast routing on resource-limited sensor nodes. In *in Proceedings of the 2nd International Workshop on Multi-hop Ad*, pages 87–94. ACM Press, 2006.

[18] Wenyuan Xu, Ke Ma, Wade Trappe, and Yanyong Zhang. Jamming sensor networks: attack and defense strategies. *IEEE Network*, 20(3):41–47, 2006.

[19] Wenyuan Xu, Yanyong Zhang, and Timothy Wood. The feasibility of launching and detecting jamming attacks in wireless networks. In *In ACM MOBIHOC*, pages 46–57, 2005.

[20] RFC 3626: Optimized Link State Routing Protocol (OLSR), 2003.

[21] Martin Dufwenberg. Game theory, 2010.

[22] Stephan Bohacek, Joao Hespanha, Junsoo Lee, Chansook Lim, and Katia Obraczka. Game theoretic stochastic routing for fault tolerance and security in computer networks. *IEEE Transactions on Parallel and Distributed Systems*, 18(9):1227–1240, 2007.

[23] Joao Hespanha and Stephan Bohacek. Preliminary results in routing games. In *In Proceedings of the American Control Conference*, 2001.

[24] Frederick S. Hillier and Gerald J. Lieberman. *Introduction to Operations Research, 4th Ed.* Holden-Day, Inc., 1986.