Constructive Relay based Cooperative Routing in Mobile Ad hoc Networks

By

Jingwen Bai

SUBMITTED FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

Supervised by Chris Phillips and Yan Sun

Department of Electronic Engineering and Computer Science,
Queen Mary, University of London

September 21, 2016

To my family

Acknowledgements

I would like to express my sincere gratitude to my primary supervisor, Reader. Chris Phillips, who has given me all the supports, assistance and encouragement throughout my Ph.D. study. Chris spends huge amount of time discussing ideas, carefully and in time written checking, as well as offering positive support when I feel frustrated. I would also like to thank my second supervisor, Dr. Yan Sun, who supervises me every detail of this research and provides endless suggestions along four years of my study.

In addition, I will thank Chinese Scholarship Council (CSC), funding me for four years' study. Without CSC, I cannot have the opportunity for further study in this great university.

Further more, my big thanks will give to Dr. Ling Xu, who guides me a lot about my Ph.D. life. Also, I feel appreciated for my friend Shuang Liu, who gives me quick help of programming when I need. Besides, I am deeply grateful for Mingming Fan, Tao Liang and Dan Liu from Beijing University of Posts and Telecommunications, thanks for having heated discussion with me about my research.

Last but not least, my thanks will go to my mother, my father and my little brother. Thanks for the endless supports from my families, both emotionally and financially. Without you, I cannot finish this huge work.

Abstract

Mobile Ad hoc networks (MANETs) are flexible networks that transmit packets node-by-node along a route connecting a given source and destination. Frequent link breaks (due to node mobility) and quick exhaustion of energy (due to limited battery capacity) are two major problems impacting on the flexibility of MANETs. Cooperative communication is a key concept for improving the system lifetime and robustness and has attracted considerable attention. As a result, there is much published research concerning how to utilize cooperative communication in a MANET context. In the past few years, most cooperative technologies have focused on lower layer enhancements, such as with the Physical Layer and MAC Layer, and have become very mature. At the Network Layer, although some re search has been proposed, issues still remain such as the lack of a systematically designed cooperative routing scheme (including route discovery, route reply, route enhancement and cooperative data forwarding), the use of cooperative communication for mobility resilience, and route selection (jointly considering the energy consumption, energy harvesting potential and link break probability).

Driven by the above concerns, a novel Constructive Relay based CooPerative Routing (CRCPR) protocol based on a cross-layer design is proposed in this thesis. In CRCPR, we first modify the traditional hello message format to carry some additional neighbour information. Based on this information, a key aspect of this protocol is to construct one or more small rhombus topologies within the MANET structure, which are stored and maintained in a COoPerative (COP) Table and Relay Table. Next, the route request procedure is re-designed to improve resilience to node mobility with a scheme called Last hop Replacement. Finally, assuming nodes are mostly battery-powered, destination node based route-decision criteria are explored that can consider energy consumption, energy harvesting and link break probability to determine an appropriate route across the MANET.

As the hello message format is modified to carry additional information, the control overhead is increased. However, in order to improve the control message efficiency, a new generalised hello message broadcasting scheme entitled Adjust

Classified Hello Scheme is developed, which can be deployed onto every routing protocol employing a hello mechanism.

As well as designing a new routing protocol for MANETs, including route discovery, route selection, route reply, route maintenance, route enhancement and cooperative data forwarding, the proposed scheme is implemented within an Opnet-based simulation environment and evaluated under a variety of realistic conditions. The results confirm that CRCPR improves mobility resilience, saves energy via cooperative communication and reduces the control overhead associated with the hello message mechanism.

Contents

List of Figures

List of Tables

Glossary

ABR	Associativity Based long-lived Routing
ACHS	Adjust Classified Hello Scheme
AODV	Ad hoc On-demand Distance Vector routing
AODV-EHA	Energy Harvesting Aware Ad hoc On-Demand Distance Vector Routing Protocol
BER	Bit Error Rate
CC	Cooperative Communication
CCON	Cooperative CONfirm
CDMA	Code Division Multiple Access
CENF	Cooperative Error NotiFication
CH	Cluster Head
CHLO	Cooperative HeLlO
CLRP	Cooperative Local RePair
CORMAN	Cooperative Opportunistic Routing in Mobile Ad hoc Networks
CRABSLS	Cooperative Routing Algorithm Based on Symmetric Link Selection
CRCPR	Constructive Relay based CooPerative Routing
CREP	Cooperative route REPly
CREQ	Cooperative route REQuest
CSI	Channel State Information
CSMA	Carrier Sense Multiple Access
CWR	Cooperative Wireless network Routing
DAG	Directed Acyclic Graph
DCF	Distributed Coordination Function

DEHAR	Distributed Energy Harvesting Aware Routing Algorithm
DSDV	Destination Sequence Distance Vector
DSR	Dynamic Source Routing
DSTC	Distributed Space-Time Coding
EBCR	Energy Balanced Cooperative Routing
EEDCR	Energy Efficient Decentralized Cooperative Routing
EH	Energy Harvesting
ER	Energy Restricted
FB	Function Block
FDMA	Frequency Division Multiple Access
FSM	Finite State Machine
HB	Header Block
IN	Intermediate Node
IoT	Internet of Things
ITU	International Telecommunication Union
LEACH	Low Energy Adaptive Clustering Hierarchy
LORA	Least Overhead Routing Approach
M2M	Machine to Machine
MAC	Medium Access Control
MANET	Mobile Ad hoc NETwork
MIMO	Multiple Input and Multiple Output
MPR	MultiPoint Relay
NSN	Neighbour'S Neighbours
OFCOM	OFfice of COMmunications
OFDMA	Orthogonal Frequency Division Multiple Access
OLSR	Optimized Link-State Routing
ORA	Optimum Routing Approach
OSI	Open Systems Interconnection
PFA	Path Finding Algorithm
PHMS	Periodic Hello Message Scheme
PNL	Potential Next-hop Location

QoS	Quality of Service
RHMS	Reactive Hello Message Scheme
RMV	Routing Matrix Values
RR	Request to Recruit
RREP	Route REPly
RREQ	Route REQuest
RERR	Route ERRor
SISO	Single Input and Single Output
SNR	Signal-to-Noise Ratio
SSR	Signal Stability Routing
STBC	Space-Time Block Coding
STC	Space-Time Coding
SV	State Variables
TDMA	Time Division Multiple Access
TORA	Temporally Ordered Routing Algorithm
TV	Temporary Variables
WBA	Wireless Broadcast Advantage
WLAN	Wireless Local Area Network

Chapter 1

Introduction

1.1 Research Motivation

1.1.1 New Applications

Currently, the most common systems which utilize wireless communication are cellular networks and Wireless Local Area Networks (WLANs). However, both of these can only traverse the "last hop" connecting the mobile device to wired infrastructure and cannot adapt to the emerging self-organized communication applications. The first representative application is the Internet of Things (IoT), in which portable mobile devices can provide complex and intelligent services. Another application scenario is in the field of Robotics. The BigDog robot funded by the Defence Advanced Research Projects Agency (DARPA) [1] is designed for the U.S. military to accompany soldiers in terrain which is too rough for vehicles and engage in cooperative search and rescue operations. Other application scenarios include students needing to interact during a lecture in the open air; or disaster recovery teams needing to coordinate relief information after earthquakes or floods. These scenarios require devices to organize themselves into a network, and to build routes among themselves without external additional support. Therefore, the current infrastructure-based wireless networks like cellular networks and WLANs are no longer suitable.

1.1.2 Problems of MANETs

Mobile Ad hoc Networks (MANETs) [2], with infrastructure-less, spontaneous and arbitrary multi-hop features, have been recognized as a popular approach for above new scenarios. Nevertheless, MANETs also face certain constraints, e.g.,

the limited communication range of mobile nodes, link breaks due to node mobility and the restricted power supply.

Due to the limited capabilities of nodes in MANETs in terms of processor performance and battery capacity, the transmission range of each device is short. If long range communication is required, it is therefore necessary for intermediate node to forward traffic on behalf of other nodes as a router. In addition, all these nodes (routers) which form a multi-hop route to implement data transmission are free to move independently in any direction and can change their links to other devices frequently; thus, link breaks cannot be avoided due to rapid and unpredictable changes in the wireless topology. Also, as the nodes in MANETs rely on the batteries, once one node in the network suffers energy exhaustion, the transmission may be terminated. Therefore energy consumption is a key issue which needs to be noted.

Multi-hop, mobility and battery power make the design of adequate routing protocols in MANETs a major challenge to meet the requirement of the new scenarios.

1.1.3 Exiting Research Limitations

In order to address these difficulties, cooperative communication [3] has received much attention for its perceived benefits, such as lower power consumption, reduced interference and potential channel diversity gain. Alongside more mature Physical Layer [4][5][6] and MAC Layer [7][8][9][10] mechanisms to support cooperative communication, research interest has grown regarding cooperative communication at the Network Layer.

However, several important aspects of cooperative communication in the Network Layer are still overlooked: (1) A fundamental routing structure at the Network Layer for cooperative communication does not exist. Without this, contributions from Physical Layer and MAC Layer cannot lead to better overall performance. Also, research efforts on the selection of cooperative relay nodes at the Network Layer is unable to make a practical contribution, without a complete routing structure for cooperative communication. (2) Even though cooperative communication can reduce the energy consumed for data transmission with the help of cooperative diversity, how to reduce the link breaks via cooperative communication, has still not received much attention. (3) If the energy harvesting

ability is also involved when selecting a route, the lifetime of the whole network can be improved.

1.2 Research Objectives

Given above issues, a reactive fundamental cooperative routing structure, called Constructive Relay based CooPerative Routing (CRCPR) [1] protocol is proposed in this thesis, which utilizes topological information stored in a Cooperative Neighbour Table, COoPerative (COP) Table and Relay Table to implement cooperative transmission and provide mobility resilience.

The objectives of this research are:

- To design a complete networking framework which can support cooperative communication and enable the research achievements concerning cooperative communication in the Physical Layer and MAC Layer contribute to practical network-wide scenarios.
- To extend the tolerance to node mobility and reduce the risk of link breaks via cooperative communication at the Network Layer.
- To prolong the network lifetime from two aspects: utilize cooperative diversity to save transmission energy and consider node energy harvesting ability when making the route selection.
- To improve the hello message broadcasting efficiency and reduce the control overhead for routing protocols in MANETs.

1.3 Novelty and Contributions

Our research proposes a reactive fundamental cooperative routing structure, called Constructive Relay based CooPerative Routing (CRCPR) protocol, which includes a complete routing scheme with economic energy consumption and high robustness to mobility induced link breaks. The unique contributions of this work are:

[1] Please note, the protocol name is changed from the previous title "Cooperative Relay Routing Protocol (CRRP)" to "Constructive Relay based CooPerative Routing (CRCPR)". The reason is the revised title better encompasses CRCPR's key feature of using several table structures to construct relay nodes and improve mobility resilience.

1. A complete systematic design at the Network Layer to support cooperative communication, developed and evaluated using the OPNET platform. This network design can utilize research contributions from Physical Layer and MAC Layer to improve practical network performance.

2. A locally self-managed scheme for cooperative communication based on a four-node COoPerative (COP) topology including in COP Table. Based on an innovative COP Possibility Detection Algorithm employing information included in a Cooperative Neighbour Table, a COP topology can be created and maintained to implement cooperative communication, energy saving and providing mobility resilience.

3. A robust link-break handling mechanism to construct relays for data forwarding via a Relay Table. This mechanism explores a "hello unicast scheme" to enhance the neighbour relationship and increase the possibility of a valid route being identified. In order to utilize this enhanced neighbour relationship during data transmission to improve route robustness, the relay mode of forwarding data is considered.

4. A novel route request procedure to carry COP topology information to the destination that contributes to the final route selection. The more COP topologies that are utilized in the data transmission, the more stable and energy-efficient will be the final route. The novel route request procedure cannot only carry traditional route metric like the number of hops, but also COP topology information and some energy factors to improve mobility resilience and energy efficiency.

5. A novel route selection algorithm that is resilient to link breaks and provides economic energy consumption and takes into account energy harvesting. Based on information transported by the novel route request procedure, the destination node will invoke this new route selection algorithm to obtain a comprehensive value to evaluate each route and decide a final one in terms of energy harvesting ability, real-time residual energy and link break probability.

6. An improved hello message broadcasting scheme named the Adjust Classified Hello Scheme (ACHS) is proposed, which not only can be deployed into CRCPR to reduce the broadcasting hello messages, but can be adapted to any routing

protocol with hello messages to improve broadcasting efficiency. Also, ACHS is an extensible scheme which is easy to adjust to different scenarios by considering other classification methods.

1.4 Authorship

1. **Bai, J.**; Sun, Y.; Phillips, C.; "CRRP: A Cooperative Relay Routing Protocol for IoT Networks," in *IEEE PIMRC 2016 Mobile and Wireless*, September 2016.

2. **Bai, J.**; Liang, T.; Sun, Y.; Phillips, C.; "Hello Message Scheme Enhancement in CRMANET," in *Wireless Telecommunications Symposium (WTS)*, April 2016.

3. **Bai, J.**; Fan, M.; Yang, J.; Sun, Y.; Phillips, C.; "Smart Energy Harvesting Routing Protocol for WSN based E-Health Systems," in *MobileHealth '15: Proceedings of the 2015 Workshop on Pervasive Wireless Healthcare*, June 2015.

4. Sun, Y.; Phillips, C.; **Bai, J.**; Zhang, H.; Hou, J.; Wang, S.; "Cognitive Radio Mobility Based Routing Protocol for CR enabled Mobile Ad Hoc Networks," in *2013 IEEE Globecom Workshops (GC Wkshps)*, December 2013.

5. Sun, Y.; Phillips, C.; Wang, S.; **Bai, J.**; "An improved QoS awareness scheduling scheme for CR Mobile Ad hoc Networks," in *Wireless Telecommunications Symposium (WTS)*, April 2013.

1.5 Thesis Outline

The rest of the thesis is organized as follows:

Chapter 2 provides relevant background knowledge concerning MANETs and typical routing protocols as well as related work. Section 2.2 presents a basic introduction to MANETs. In Section 2.3, several typical routing protocols are described in terms of their type, features and operating principles. In Section 2.4, the state of art in cooperative routing protocols and energy-aware routing protocols are considered.

Chapter 3 gives a detailed description of the CRCPR design. Initially, the overview of CRCPR is given in Section 3.1. Next, some necessary assumptions are

presented in Section 3.2. The detailed design of CRCPR is then presented in Section 3.3, which includes Message Format, Neighbour Discovery, Route Discovery, Route Reply, DATA Forwarding, Route Enhancement, Route Break Detection, Route Reconstruction and Route Selection Criteria.

Chapter 4 introduces the OPNET simulation platform and the CRCPR implementation workflow. More precisely, Section 4.1 gives a brief introduction to OPNET features, such as Hierarchical Network Models, Event-Driven Simulation, State Machine, Editors and so on. In Section 4.2, details of how CRCPR was implemented in OPNET are provided.

Chapter 5 proposes a new hello message scheme called Adjust Classified Hello Scheme (ACHS) to improve hello broadcasting efficiency and reduce network congestion. In Section 5.2, typical hello message broadcasting schemes are introduced. Section 5.3 then presents details regarding ACHS. Finally, conclusions are presented in Section 5.4.

Chapter 6 presents results to assess the CRCPR performance. Section 6.1 mathematically analyses the CRCPR performance. From Section 6.2 to Section 6.6, simulation results are provided and discussed. A summarized conclusion of CRCPR performance is given in Section 6.7.

Chapter 7 provides the conclusion and considers future works.

Chapter 2

Background and Related Work

2.1 Mobile Ad hoc Networks

Two relevant definitions of the term "ad hoc" are listed in the Merriam-Webster dictionary [11]: "formed or used for specific or immediate problems", and "fashioned from whatever is immediately available". Furthermore in [12], more definitions about "ad hoc" are given: "can take different forms", "can be mobile", "standalone" or "networked". All these definitions show two main advantages of ad hoc wireless networks: (1) They can be designed for specific applications. (2) They can be built from whatever network nodes are available.

As shown in Figure 2.1, a mobile ad hoc network (MANET) is a collection of wireless mobile nodes (or routers) dynamically forming a temporary network without the use of any existing network infrastructure or centralized administration [13].

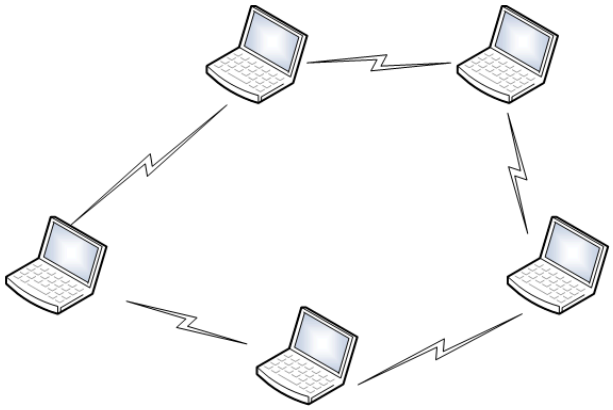


Figure 2.1: Mobile Ad hoc Networks

MANETs can reduce the cost to install and maintain a typical network infrastructure because of the features of being infrastructure-less, spontaneous and

arbitrary multi-hop. All of these features can be best illustrated by contrasting them with the most prevalent wireless networks: cellular networks [14] and Wireless Local Area Networks (WLANs) [15]. For cellular networks in Figure 2.2, the geographic area is divided into small cells by the base station located in the cell center. All the mobile devices in each cell can communicate with the base station directly. Additionally, the base station is connected to a backbone wired network which is responsible for all networking functions like authentication, call routing and handoff. As we can see, there is no peer-to-peer communication between mobile devices and all the communication between the base station and the mobile devices employs a single-hop routing. For WLANs in Figure 2.3, they also have a similar centralized, single-hop architecture: mobile devices communicate with a centralized access point (router) which is also connected to the backbone Internet. The access point performs all networking and control functions. In contrast, MANETs use peer-to peer communication networking. Control functions are distributed among all the mobile devices in the network. In addition, MANETs can be rapidly deployed, configured and may be connected to the Internet. Therefore they enhance the quality of service access and provide wireless connectivity in areas with poor or no cellular network or WLAN coverage.

All the above characteristics of MANETs are especially important for military applications; therefore a lot of researches into Ad hoc wireless networking [16][17][18][19][20][21] has been supported by the Defence Advanced Research Projects Agency (DARPA) [1] and the US Navy. As such many basic design principles for MANETs were explored and confirmed through this early research. However, although many advantages of MANETs have been discussed over the past several decades, optimal design, performance, and fundamental capabilities of MANETs remain poorly understand [13].

2.2 Challenges in Mobile Ad hoc Networks

2.2.0.1 Spectrum Allocation

Currently, radio spectrum usage is under the control of the Office of Communications (OFCOM) [22], which is the government-approved regulatory and competition authority for the broadcasting, telecommunications and postal industries of the United Kingdom. Most research in MANETs are operated within the Indus-

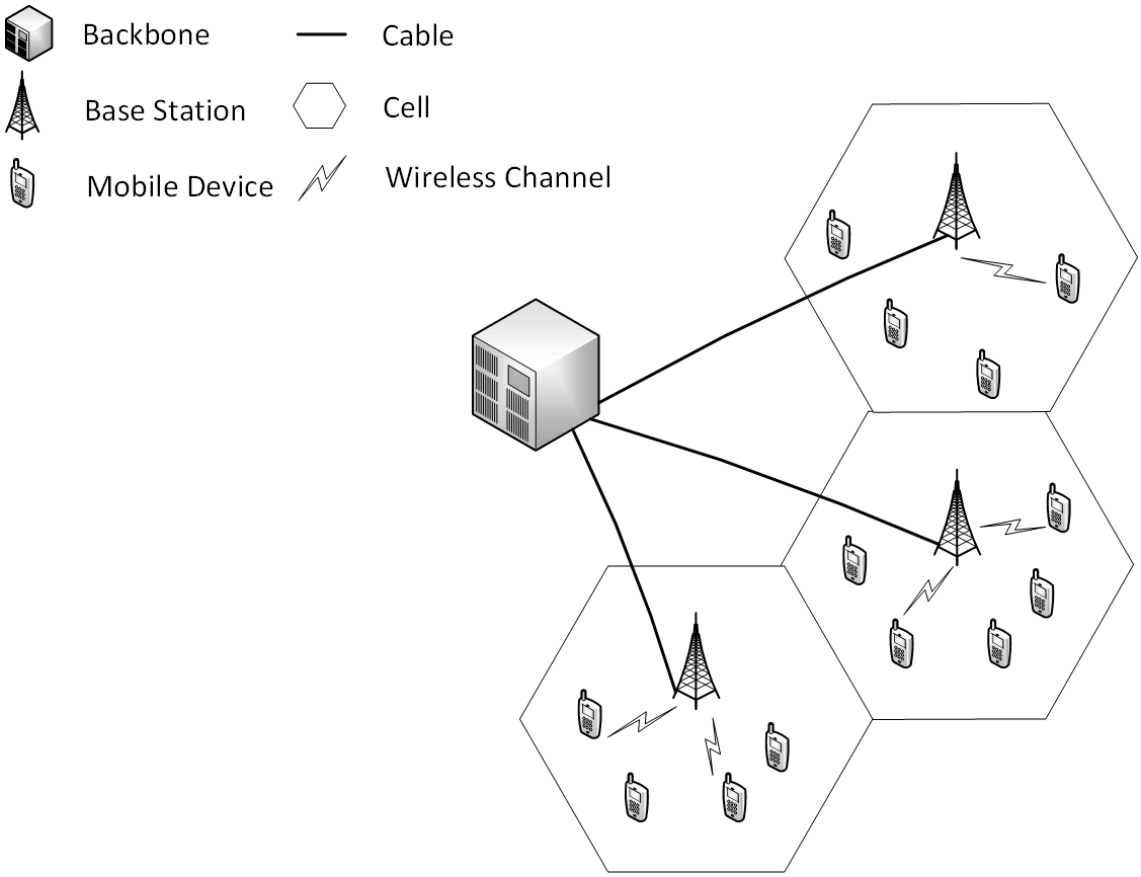


Figure 2.2: Cellular Networks

trial, Scientific and Medical (ISM) radio bands which are defined by the International Telecommunication Union (ITU) Radio Regulations [23]. Therefore, if there is no specified allocation of spectrum for MANETs, interference cannot be avoided. The most common example of spectrum interference is from the daily used microwave oven which works in the 2.4GHz band can interfere with WLANs signals.

2.2.0.2 Medium Access Control

The Medium Access Control (MAC) Layer [24] in wireless networks decides how different users share the available spectrum and ensures the reception of the packets successfully over the shared spectrum. Due to the lack of centralized control in MANETs, Time Division Multiple Access (TDMA) [25], Frequency Division Multiple Access (FDMA)[26] and Code Division Multiple Access (CDMA) [27] schemes are not suitable controlled access schemes. However, a random access scheme can adapt to the infrastructure-less feature of MANETs such as Carrier Sense Multiple

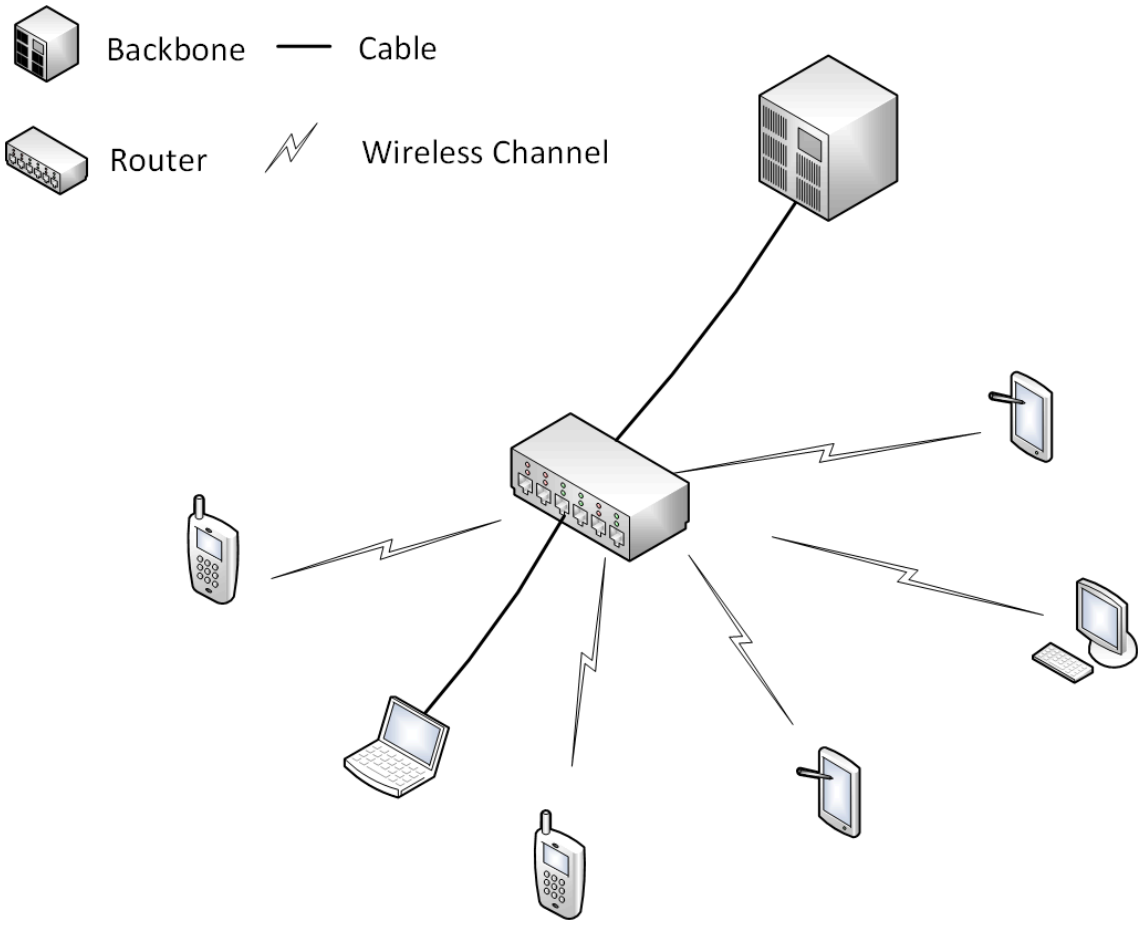


Figure 2.3: Wireless Local Area Networks

Access (CSMA) [28], thus distributed MAC mechanisms such as Multiple Access with Collision Avoidance (MACA) [29], MACA for Wireless (MACAW) [30] and 802.11 Distributed Coordination Function (DCF) [31] have obtained widespread popularity in MANETs. However, as all of the above MAC mechanisms are based on CSMA, they all suffer from the two well-known problems: the hidden terminal (node) problem and exposed terminal (node) problem [32]. These issues must be addressed when designing an efficient MANET framework.

2.2.0.3 Networking

The Network Layer is responsible for building and maintaining the end-to-end connections in the network. As the mobile nature of MANETs leads to frequent and unpredictable changes of network topology, most of the main functionalities of the networking protocols need to be redesigned. Much work has focused on routing protocol design in MANETs in the past decades and there are two main categories:

proactive routing like Destination-Sequence Distance Vector (DSDV) [33], Optimized Link-State Routing (OLSR) [34], Topology-dissemination Based on Reverse Path Forwarding (TBRPF) [35], Wireless Routing Protocol (WRP) [36], Cluster Switch Gateway Routing (CSGR) [37], Source Tree Adaptive Routing (STAR) [38][39] and reactive routing such as Ad hoc On-demand Distance Vector routing (AODV) [40], Dynamic Source Routing (DSR) [41], Temporally Ordered Routing Algorithm (TORA) [42][43], Associativity Based long-lived Routing (ABR) [44], Signal Stability Routing (SSR) [45].

Unfortunately, it is difficult to design a single routing protocol that can efficiently support all the applications in MANETs. All of the above protocols have their own advantages and disadvantages in relation to different applications. Along with the development of hardware technology, more research in the Network Layer is required to satisfy emerging applications such as home networks, vehicle networks, sensor networks, emergency response networks and so on.

2.2.0.4 Mobility

Due to the mobility of nodes in MANETs, a mobility model that can emulate the movement pattern of targeted real life application in a reasonable way is desirable when determining the routing protocol performance.

The mobility model is designed to describe the movement pattern of mobile users in terms of: velocity, location and acceleration changes over time. There are two kinds of mobility model for network simulation: traces and synthetic models [46][47]. Trace mobility models are obtained from a real life system based on long-term observations. They provide accurate information such as the number of mobile nodes, the speed, the location, the movement traces and so on. However, some new application environments in MANET are not easy to be modeled based on the long time-consuming mobility model traces [48]. For this type of situation, synthetic models are an appropriate choice. A synthetic models' purpose is to represent the behaviors of mobile nodes realistically without the need for traces. One popular example of a synthetic model for MANETs is the Random Walk Mobility Model [47][49]. The Random Walk Mobility Model is proposed to model the environment where mobile nodes move in unpredictable ways. In this model, one mobile node moves from its current location to a new location by a randomly selected velocity. The new velocity is decided from pre-defined [*min-speed*, *max-*

speed] and [*0*, *2π*] ranges. The calculation of a new location starts at the end of a constant time interval t or a constant distance traveled d.

Although synthetic models can be utilized to test the influence of mobility on network performance at a simulation level, more accurate trace models are still needed to better reflect realistic environments.

2.2.0.5 Energy Efficiency

Firstly, a node in MANETs can be both a data source/sink node and a router that forwards data for other nodes. Forwarding packets on the behalf of others will consume power. Additionally, most mobile nodes are operated by batteries with a limited lifetime; thus energy efficiency must be considered when designing a MANET. Furthermore, although rechargeable batteries can be employed, some special applications like sensors imbedded in walls or dropped into a remote region cannot be recharged easily. Therefore, according to this critical issue, energy efficiency is a very important aspect of MANETs.

Various techniques, in terms of hardware and software, have been proposed to reduce energy consumption of MANETs. The u-AMPs and Picoradio projects are aimed at developing radios for hardly-recharged applications that can operate on less than 100 microwatts and exploit energy harvesting to prolong the device lifetime [50][51][52]. Much research at the Physical Layer utilizes signal processing techniques to reduce transmission power and it is widely assumed that the energy required for signal processing is small given the improvement in hardware technology [53][54]. However, the results in [55][56] suggest that signal processing associated with packet transmission and reception, and even computation still consumes considerable power.

In fact, energy efficient design involves all layers of the protocol stack not only the Physical Layer [57][58][59][60]. Therefore, further focus on cross-layer design for MAENTs is needed further to meet the energy performance requirement.

2.2.0.6 Summary

MANETs need to exhibit self-organizing features and they must perform routing and packeting-forwarding functions. As there is no centralized access points, the routing functions need to be implemented in a distributed manner considering mobility and power constraints, which remains a big challenge for the current research

field. Furthermore, if higher layers like the Network Layer cannot adequately support mobility and energy-aware route selection, corresponding techniques in the lower layers like the MAC Layer and Physical Layer cannot be applied to realistic applications. Hence, in this work, we focus on the Network Layer and try to solve the MANETs challenges in terms of the networking protocol, mobility accommodation and energy efficient transmission based on cooperative communication.

2.3 Classic Routing Protocols in MANETs

MANETs are autonomously self-organizing networks, and all the nodes can typically move randomly; therefore, it may experience more rapid and unpredictable topology changes. Also, because of the multi-hop feature, the transmission may be completed via a data forwarding procedure. Hence, as a node in MANETs must perform both the role of an end system (where the user interacts and where users applications are executed) and that of an intermediate system (packet forwarding) without any centralized control, the routing protocols in MANETs should consider node mobility and the responsibility for routing in a distributed way.

Due to the importance of routing protocols for the efficient operation of a MANET, a lot routing protocols have been proposed. Currently, based on how routing information is obtained, the main categories of routing protocols are: proactive and reactive routing protocols.

2.3.1 Proactive Routing Protocols

Proactive routing protocols are also known as "table-driven" protocols which have inherited most of the routing procedures of wired networks, such as DSDV. All the nodes in the network periodically calculate the routes to all reachable nodes and save this routing information in a priority list. Any changes of topology will lead to routing information updates for each node in the network. For a proactive routing protocol, once a node needs to send data as a source node, it can obtain the complete routing path immediately and choose the most suitable route to transmit the data.

The advantage of a proactive routing protocol is that quick confirmation of a route for data transmission can reduce the transmission delay. The disadvantages are: (1) a large control overhead to maintain the consistent and up-to-date routing

information in each node, (2) it is difficult to adapt to scenarios with rapid topology changes and slow to respond to route failures.

2.3.1.1 DSDV

Destination-Sequence Distance Vector (DSDV) [33] is a proactive unicast routing protocol in MANETs which is an improved version of the traditional Bellman-Ford-Moore algorithm [61]. The most important point in DSDV is the route table. Each entry in route table consists of the next hop to a destination, the cost of each entry, and a destination sequence number. The routing information is broadcast periodically to the neighbours to keep the route table updated and consistent across the whole network. Once a source node has traffic to send, it can get the next hop immediately by searching the information saved in the route table. The next hop address in the entry with the lowest cost will be regarded as the target node. When the data reaches the next hop, the search procedure is repeated until the packet arrives at the destination node. The destination sequence number corresponding to a given destination entry is used to ensure obsolete information is overwritten and to avoid route loops.

The advantages of DSDV include: (1) loop paths can be avoided, (2) transmission delays are low, (3) quick confirmation of a route for data transmission. The disadvantages are: (1) many control messages need to be sent to maintain the routing information stored in each node, (2) DSDV may lead to congestion in high-density networks.

2.3.1.2 STAR

Source Tree Adaptive Routing (STAR) [38][39] is also a proactive routing protocol which was developed in the SPARROW [62] project. In most proactive routing protocols like DSDV, they tend to maintain optimum routes between the source node and the destination node which is called the Optimum Routing Approach (ORA) [63]. The problem with ORA is that a lot of control messages are present in the network. In order to minimize control overhead, STAR implements the Least Overhead Routing Approach (LORA) [64] and does not require periodic route updates. In STAR, each node maintains a source tree which is a set of links to a destination. Every node only knows the links between itself and the neighbours as well as the source trees reported by its neighbours. To reduce the number

of route updates, only changes of the source tree are propagated. Although the selective update scheme of STAR can decrease the control overhead of the network compared to DSDV, frequent topology changes due to mobility in MANETs still make the number of route updates increase dramatically.

2.3.2 Reactive Routing Protocols

Reactive routing protocols are also called "on-demand" protocols which are only established when the source node has traffic to send to a given destination. The route request procedure is initialized by the source node via flooding way: the route request packet is broadcast across the whole network until it reaches the destination. Then a route reply packet will be sent to the source node via the reverse path. After the route is established, data transmission commences.

The advantages of reactive routing protocol are: (1) less control messages are needed as there is no need to update and maintain the routing information at all times, (2) more adaptable to different scenarios because the route determination is determined only when needed. The disadvantages are: (1) as the data transmission commences only when the route discovery has concluded, so there is a high initial delay that cannot be avoided, (2) the sudden flooding broadcast scheme may be not efficient and can even lead to network congestion.

2.3.2.1 AODV

AODV (Ad hoc On-demand Distance Vector), an improvement on DSDV [33], typically creates a route on a demand basis by minimizing the number of hops instead of maintaining a complete list of routes as in DSDV. When a source node intends to send data and does not have an appropriate route, it will broadcast a Route REQuest (RREQ) packet to its neighbours until either the destination or an intermediate node with an appropriate route is confirmed. Then a Route REPly (RREP) packet will be unicast along the reverse path established during the route discovery phase by the RREQ back to the source node. After the shortest route is found, data will be forwarded along the route to the destination. Due to node mobility, the route may be broken. A failure notification carried by a Route ERRor (RERR) packet will be sent to the source node by the upstream neighbour of the broken link and the source node will initiate a new route discovery phase. In addition, AODV utilizes a periodic local broadcasting scheme to broadcast hello

messages to maintain the neighbour relationship which is used to check the route validity.

2.3.2.2 DSR

DSR is also an on-demand routing protocol like AODV. However, the most important difference is that DSR requires each mobile node to maintain a route cach which is updated as new routes are obtained.

When a source node intends to send data, it will consult the route cache instead of broadcasting route request packets. If true, the source node will use this route to send data. If the appropriate route cache entry does not exist, a route request packet will be broadcast. Each intermediate node receiving the route request packet checks if it has a appropriate route cache entry for the destination. If not, its own IP address will be appended into this route request packet and the route request packet is rebroadcast. A route reply packet is generated when either the route request reaches the destination, or when it reaches an intermediate node that contains in its route cache an suitable route the the destination. As all the IP addresses of the route have been saved in the route request packet, the route reply packets can utilize these records to get back to the source node. Another difference from AODV is that DSR maintains the route through the use of route error packets or acknowledgements instead of hello messages. When an ACK of the data is not received after a specified time, the route will be regarded as invalid.

The route cache scheme can help DSR recover from route break states quickly rather than having to initiate the route discovery phase. However, maintaining of route cache also results in a high cost in terms of battery-limited devices in MANETs.

2.3.2.3 TORA

Temporally Ordered Routing Algorithm (TORA) [42][43] is a loop-free and distributed routing protocol based on the concept of link reversal [65]. It is designed to adapt to highly dynamic mobile networking environments. The key feature of TORA is that if a link break happens in the network, the control message can be restrictedly sent to locations near the occurrence of the link break. In order to achieve this feature, all the nodes are required to maintain routing information about their adjacent nodes. When a source node has traffic to send, it broadcasts

a request packet to the destination. After the destination receives the request packet, it will establish a Directed Acyclic Graph (DAG) with a "height" metric value "0". The neighbours of the destination continue to build a DAG with a higher metric value until the DAG reaches to the source node. Then the data will be transmitted along the DAG from the node with a higher cost value to the node with a lower cost value like water rolling down a hill. Due to mobility, if the DAG route is broken and a node loses all its downstream links, this node will set a new "height" metric value as a reference level and re-establish its DAG in the reverse direction. More details are provided in [42]. All the "height" metrics are dependent on the logical time of a link failure, therefore TORA can only work well in a network which has synchronized clocks typically provided via an external time source such as GPS.

2.3.2.4 ABR

Associativity Based long-lived Routing (ABR) was invented by the author in [44]. The concept of associativity related to the spatial, temporal and connection stability of a node are the key aspects to improve the link stability and require fewer route reconstructions. A periodic beacon message is used to evaluate the link associativity. If a node hears a beacon message from its neighbours, this message will be saved in an associativity-table as an associativity tick. ABR thinks the more associativity ticks exist between two adjacent neighbours, the more stable is the link. An associativity tick can be carried by a route request packet to the destination node. After an appropriate waiting time (during this period, all the reasonable route request packets are assumed to have arrived at the destination node), the destination node will choose the route with highest associativity ticks as the final route. If the overall degree of the association of two or more routes is the same, then the route with the least number of hops will be selected. If several routes have the same hop count, then one of the routes will be chosen randomly. Finally, the destination sends a route reply packet back to the source node via the selected route and data forwarding commences. If a link break happens, ABR also proposes a local-repair scheme to recover the route locally so that it can reduce the control overhead and minimize the affected area.

2.4 Related Work

In addition to the classic routing protocols described in Section 2.3, some more protocols in MANETs, such as [66][67][68], are refined according to the protocols in wired networks, where a packet can only be received by the nodes attached to the same cable. However, once a data packet is sent out over a physical wireless channel, it can be heard by all the neighbour nodes. This overhearing feature (referred to Wireless Broadcast Advantage (WBA) in [69]) has been considered as totally negative for most research because the overhearing signal will influence the reception of targeted data as interference. Later, with the development of improved antenna techniques, cooperative communication [70][71], which can utilize the overhearing feature in wireless networks, has received much attention due to its perceived benefits, such as lower power consumption, reduced interference and potential channel diversity gain. In order to solve the problems of MANETs such as mobility and energy-limitations, cooperative communication is regarded as a very promissing direction.

Initially, research on cooperative communication started to attract interest at the Physical Layer and MAC Layer. Two types of cooperative communication are defined in [72]: *repetition-based* and *spacetime-coded*. The former consists of the sender broadcasting data to its receiver and relays and relays repeating the sender's packet to the receiver as a backup scheme. The latter requires the relays to transmit the data simultaneously using a suitable coding scheme such as Space-Time Coding (STC) [73], Space-Time Block Coding (STBC) [74] or orthogonal Distributed Space-Time Coding (DSTC) [75] and then obtaining the cooperative diversity to save transmission energy. The authors in [4] and [5] propose a Physical Layer MANET solution for scenarios where two nodes cooperatively transmit the same data towards a common destination while the work in [76][6] considers a relay network scenario where transmission from a single source is assisted by one or more cooperative nodes, also from the perspective of the Physical Layer. Research concerning cooperative communication in the Medium Access Control (MAC) Layer has attracted attention as well. Proactive cooperative MAC [7][8] and reactive MAC [9][10] are well discussed. In these papers, proactive cooperative MAC schemes employ a relay selection process before direct transmission is attempted, whilst reactive cooperative MAC schemes employ relay node selection only when direct transmission fails.

Alongside more mature Physical and MAC Layer enhancements, the importance and usability of cooperative communication has also been considered at upper layers of the network protocol stack.

2.4.1 ExOR

Cooperative diversity schemes proposed by the information theory community [71] [70] suggest that typical multi-hop routing techniques in wireless network which are similar to those in wired network [77][78][68][33][40] are not the best approach. In contrast, the cooperative diversity can take advantage of the broadcasting feature to send data via multiple relay nodes. The destination can combine all the received information from different relay nodes via synchronized techniques [79] or additional radio channels for each relay [80]. Therefore, the authors in [81] describe the Opportunistic Multi-hop Routing (ExOR) for Wireless Networks protocol aimed at improving throughput in wireless networks. This scheme is a milestone work that utilizes the overhearing feature of wireless links. As a data packet can be received by all the neighbour nodes, ExOR designs a scheduling transmission scheme for each intermediate node; if the transmission with a higher-priority fails, the transmission with a lower-priority will be triggered by another intermediate node.

This approach regards the overhearing signal as a backup scheme to make sure at least one intermediate node can make the transmission successfully. However, the problems are: (1) the source node completely takes charge of selecting all the intermediate nodes for each packet. Once the intermediate node is selected as a forwarder, it has to follow the pre-defined transmission schedule. This design may not response effectively to network topology changes, (2) although the overhearing feature has been considered in ExOR, some additional benefits of cooperative diversity like energy-saving are overlooked.

2.4.2 CORMAN

Based on ExOR [81], some interesting extensions have been inspired. [82] enhances ExOR to increase spatial reuse based on intra-flow network coding [83]. The works in [84] [85] utilize location information to provide a mobility handling. Intra-flow network coding and location-based schemes are complicated for MANET, hence lightweight routing algorithms are preferred. For example, the Path Finding Algorithm (PFA) [86] and Link Vector (LV) algorithm [87] are proposed to address

the routing scalability issues in MANETs. However, both of above algorithms are event-driven which incurs a significant amount of overhead. Therefore, without relying on node position information, the approach in [88] proposes a proactive source routing scheme called Cooperative Opportunistic Routing in Mobile Ad hoc Networks (CORMAN) which broadens the applicability of ExOR.

The authors do not focus on the fundamental cooperative routing structure like route discovery, route reconstruction and route maintenance but explore deeply the aspect of how to update a new route in the intermediate forwarding node live (large scale live update) and how to re-transmit missing packets between two consecutive forwarding nodes (small scale retransmission). Finally, compared to AODV [40], which is one of the classic routing protocols in MANETs, certain improvements are implemented in regard to the packet delivery ratio and the end-to-end delay. Some problems of CORMAN are: (1) As a basic proactive source routing scheme, the source node in CORMAN has a full knowledge of how to route the data to any destination at any time like OLSR [34] or DSDV [33]. This scheme consumes a large amount of control overhead to maintain the routing information of every other node in the network all the time, (2) in terms of large scale live updates in CORMAN, if an intermediate node updates the route information in the forwarding data, it has to cache these data locally and propagate them towards the source node later for updating the route information. The cached scheme increases the energy cost of the nodes in the network, (3) in terms of the small scale retransmission scheme, if a downstream node has not received a data packet successfully, CORMAN allows the nodes that are not on the forwarding list, saved in the data packet, to retransmit the data to increase link reliability. This cooperative scheme is only a backup scheme and cannot utilize cooperative diversity to save transmission energy.

2.4.3 AOCMR

As the concept of IoT is becoming more popular, Machine to Machine (M2M) networks are attracting a lot of attention. However, the multi-hop feature can reduce the network performance because of error propagation. Hence, it is advisable to utilize cooperative communication to increase the transmission range and reduce the number of hops. Some analysis has been attempted in [89][90][91][92][93]. Previous works in [94][95] have described the simplified workings of cooperative

MIMO (Multiple-Input and Multiple-Output) in MANETs. In [96], the authors present a novel joint clustering and routing mechanism, called Ad hoc On-demand Cooperative MIMO Routing (AOCMR) which makes the use of cooperative MIMO links to reduce the number of hops in multi-hop networks and further increases the throughput.

With typical clustering schemes is that the nodes in a network gather information from their surroundings and then form groups with a Cluster Head (CH). All these clusters form the "gateways" and connections between clusters can be used by AODV (Ad hoc On-demand Distance Vector) [40] or LEACH (Low Energy Adaptive Clustering Hierarchy) [97] routing algorithm, for example. The problem is all the selected hops are restricted between CH nodes based on single links even if a cooperative MIMO link in each cluster can expand the transmission range of the CH node. Therefore, firstly, the authors analyze the importance and benefits to reduce the number of hops in a large scale multi-hop network in the aspects of received SNR (Signal-to-Noise Ratio) and ergodic capacity of the channel. Secondly, focusing on the symbol error rate, an estimation scheme is proposed to evaluate the quality of cooperative MIMO links over SISO (Single-Input and Single-Output) links. After the estimation phase, each CH in the cluster will form a table of all the other clusters it can communicate with using cooperative MIMO links. Thirdly, AOCMR which is similar to AODV, routes the packets through cooperative MIMO links between clusters. The problem of AOCMR is that it inherits the basic routing scheme of AODV, thus the information of cooperative MIMO links can only be saved locally in each CH node but cannot be carried to the destination node to contribute to the final route selection criteria.

2.4.4 CRABSLS

A lot of research has considered how to utilize cooperative communication to improve network performance. The work in [98] focusing on maximum throughput cooperative routing with QoS (Quality of Service) constraints by presenting a polynomial time algorithm. The authors in [99][100] proposes an algorithm to select the best relay nodes based on link quality for cooperative communication. However, all these research are based on a premise that the links between nodes are symmetric, which is not necessarily true in MANETs. Therefore, the Cooperative Routing Algorithm Based on Symmetric Link Selection (CRABSLS) [101], inspired

by MultiPoint Relay (MPR) [102] computation, is proposed.

Due to the mobility in MANETs, links between nodes can be asymmetric. Once one node communicates with another node which is equipped with an asymmetric link between them, it can seriously impact on network performance. In order to solve this problem, firstly, a cooperative routing model is designed such that if one node is not on the selected route but it is the neighbour of two adjacent nodes on the route, this node will be selected as the cooperative node to help transmit data packets. In order to make the cooperative routing model simple, only one cooperative node can be selected between two adjacent nodes on the route. Secondly, a symmetric link detection mechanism can ensure the link between the node on the route and its cooperative node is symmetric. Thirdly, a repeated packet processing scheme is designed to deal with the repeated packet from the node on the route and the cooperative node. Although CRABSLS can solve the problem caused by asymmetric links when selecting a cooperative node, this scheme is also a backup method via the cooperative communication and does not utilize cooperative diversity to save transmission energy.

2.4.5 EEDCR

The work in [103][80] provides a paradigm where the cooperative communication, making nodes cooperate with each other in transmitting each other's information, contributes to the network throughput and energy efficiency. Although simple two-hop or three-hop relay topologies have received a lot of attention in [104][105][106][107][108][109], MANETs with hundreds of nodes need a cooperative communication model based on more complicated/realistic relay topologies. Therefore, the authors in [110] propose an Energy-Efficient Decentralized Cooperative Routing (EEDCR) scheme. EEDCR is a model for cooperative communication using a decode-and-forward approach, where a node which plays the role of a relay tries to decode an entire message and forwards it to the next hop [80]. Additionally, multiple relays can cooperate at the symbol-level and forward data together, under the assumption that frame level synchronization can be achieved among nodes and bandwidth is high enough to mitigate interference.

The whole process of cooperative communication is divided into two steps according to Figure 2.4:

- Broadcast: A message transmitted from a single node (like the source in

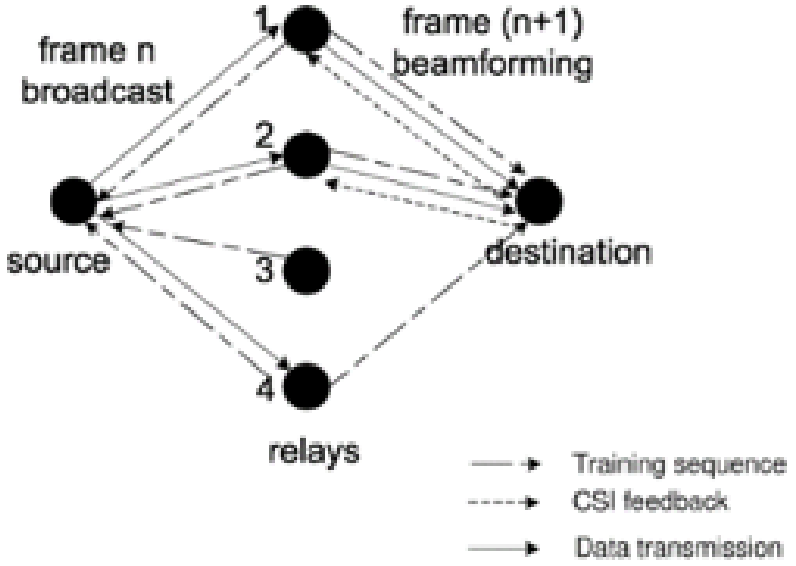


Figure 2.4: Model for Cooperative Communication [110]

Figure 2.4) can be received by more than one of its neighbours (like relays 1,2,3,4 in Figure 2.4).

- Cooperative Beamforming: If relay nodes meet the conditions for successfully decoding the message and know the Channel State Information (CSI) between themselves and the destination node, they can phase-align and scale their transmit signals so that all the messages sent by these different neighbours can be received by destination coherently. The amplitude of each message which is received by destination is P_i, therefore the total amplitude of this message will be $\sum P_i$. If $\sum P_i > \gamma$ (γ is a minimum threshold to decode a message at the destination), the message can be decoded successfully.

Collaboration among cooperative nodes depending on the CSI is further explored in [110], including slow fading and fast fading channels. For a slow fading channel, we can see the analysis results in Table 2.1 that Broadcast Cooperative (BC) policies can save more than 40% energy when compared to a None Cooperative (NC) policy.

Although EEDCR does not consider the cooperative routing protocol from the aspect of a fundamental routing structure, a general framework for decentralized cooperative communication is explored for a slow fading channel and a fast fading channel. This framework can be utilized by other researchers to further enhance

Table 2.1: Average Normalized Energy per Message under Slow Fading Channel

Class	Average energy for optimal policies
NC	5.13
BC	3.05

cooperative communication.

2.4.6 CQRP

[111] analyzes the development of cooperative communication in the Physical Layer [112][113][114][103] and MAC Layer [115][116][117][118] [119][120][121]. It then focuses on the Network Layer. For example, the work in [122] designs a cooperative communication aware routing scheme to save energy in a static wireless network. Considering the required Bit Error Rate (BER) at the destination node, [123] proposes a multi-hop cooperative routing approach for power savings. Other related works include, but are not limited to [124][125]. However, few works consider the QoS issues especially for meeting the users' bandwidth requirements. Therefore, the authors in [111] propose a Cooperative Qos Routing Path (CQRP) scheme for a multi-hop wireless network and finally implement this design in the real test bed. Firstly, a cooperative system model with "source-helper-destination" is presented to implement cooperative communication. Secondly, in order to find a cooperative path with maximum end-to-end bandwidth and reduced interference, an optimization problem based on the cooperative system model is formulated. Thirdly, in order to solve the optimization problem, a distributed Widest Cooperative Routing Path (WCRP) algorithm is proposed and finally implemented in the real test bed with the underlying routing protocol AODV.

The problem with CQRP is that the communication model is based on a three-node cooperative scheme: the transmitter, the receiver and the relay node instead of the EEDCR cooperative model proposed in [110]. The disadvantage of this three-node cooperative model is that it is difficult to utilize cooperative diversity to save energy during transmission. In addition, as the underlying AODV routing protocol is not cooperative in nature, so the quality of the cooperative rely node such as its stability as a cooperative link cannot contribute to the final route selection.

2.4.7 EBCR

As cooperative communication (CC) [3] is designed to allow a group of single antennas to form a MIMO (Multiple-Input and Multiple-Output) system, a lot of work has explored the performance of CC to minimize the energy consumption. [122] has analyzed the problem of selecting a cooperative route with minimum energy cost and proposes a dynamic-programming-based algorithm and two polynomial-time heuristic algorithms. [126] tries to find cooperative routes with minimum energy cost by assuming that the last L predecessor nodes on a route can cooperatively transmit the data to the next hop. The authors in [127] design a cooperation-based routing algorithm to select a minimum-power route by utilizing any number of cooperation-based blocks which requires the least possible transmission power. In [128][110], a complicated fading model is employed to verify that the proposed cooperative multi-hop route can find a route with the least energy cost. However, all the above works overlook the fact that the use of a minimum cost route may result in uneven energy distribution among nodes and finally reduce the network lifetime due to node death.

Therefore, in [129], the authors focus on the impact of cooperative routing on balancing the energy distribution among nodes rather than how to minimize the total energy consumption from the source to destination. A novel routing scheme called Energy Balanced Cooperative Routing (EBCR) is proposed to select cooperative relay nodes from one-hop neighbours and decide their transmission power for each hop. Based on the EEDCR cooperative communication model in [110], it can take advantage of the Physical Layer design that combines partial signals containing the same information to obtain the complete information. Taking into account the the residual energy and transmission energy, an energy-balancing algorithm is designed to choose an appropriate cooperative relay node. Finally, energy balancing performance along single and multiple routes are considered.

The problem of EBCR is that the authors assume an underlying routing protocol has been deployed that can support cooperative communication in the network; or certain non-cooperative routing strategies can be utilized to realize the proposed energy-balanced cooperative route algorithm. However, non-cooperative routing strategies cannot fully support cooperative communication. Without a complete cooperative routing protocol, the existing research is not suitable for practical scenarios.

2.4.8 CWR

A key advantage of cooperative communication is to utilize multiple relay nodes to cooperatively transmit data so that it can reduce the probability of bit errors and energy consumption. However, if only one relay node is involved at each hop during the transmission, this advantage cannot be achieved [130][114]. Therefore, the authors in [131] propose a Cooperative Wireless network Routing (CWR) scheme which considers the energy efficiency and can support synchronized transmission of multiple relay nodes and finally increase the energy efficiency and reliability of packet delivery.

CWR is based on AODV. Therefore, it inherits most of the features of AODV like route discovery, route link break detection, route reply, the hello message broadcast scheme etc. The key point of CWR is that it employs a "Recruit-and-Transmit" scheme for the transmitting and receiving nodes on the selected route to recruit neighbour nodes to assist in communication. Before the current hop forwards a data packet, it sends the valid next hop a Request-to-Recruit (RR) packet to cause the next hop to start the formation of a receiving cluster. Then the next hop will broadcast a RECruit (REC) packet to its neighbours. Each neighbour that receives a REC packet, called a potential recruit, will reply with a GRant (GR) packet to indicate their availability. After waiting a time T and collecting a number of GR responses, the next hop broadcasts a CLear (CL) packet to the potential recruits and the current hop to confirm which neighbours are selected as the cooperative nodes. Upon receiving the CL packet, the current node broadcasts a ConFirm (CF) packet to its cooperative nodes which have been selected in the last "Recruit-and-Transmit" phase to synchronize their transmission and confirm their willingness to send data cooperatively. According to the above procedure, CWR implements cooperative communication in the Network Layer. However, based on AODV scheme, CWR does not consider the factor of "recruit" ability, energy saving ability or link break probability when selecting a final route. In addition, five control packets are used for negotiating in the "Request-to-Recruit" scheme before forwarding data which leads to significant control overhead and transmission inefficiency.

Besides CWR, several other schemes have been proposed to utilize cooperative communication to save transmission energy[126][122][132][133]. However, there is a lack of a systematic strategy for evaluating these cooperative routing schemes.

Therefore, authors [134] attempt to address this issue. By comparing the current cooperative routing pros and cons, [134] lays out the future directions in this area: as existing schemes only concentrate on a single aspect but ignore others, hence an algorithm which can provide an integrated solution is needed to consider cooperative links, node's residual energy, link break probability and so forth. In order to consider integrating energy awareness into cooperative communication, we now examine several existing energy-aware protocols.

2.4.9 DEHAR

There are mainly three kinds of energy-aware protocols: energy efficient, residual energy aware and energy harvesting aware protocols. The energy efficient protocols aim at improving the network lifetime by minimizing the energy consumption. For example, [131] utilizes cooperative diversity to save transmission energy or [135][136][137] avoid data going through the nodes with low energy. The residual energy aware protocols measure the residual battery energy of the nodes and take into account the actual available energy when selecting the route such as [138][139][140][141][142][143][144]. The energy harvesting aware protocols [145][146][147][148][149][150] estimate the potential to harvest energy from external energy sources (solar energy or wind energy) to improve network lifetime.

Currently, most energy harvesting aware protocols do not consider the energy impact of selecting the route except [149][150]. The works in [149] propose a mathematical framework for an energy harvesting aware routing protocol in multi-hop wireless networks and an algorithm based on this framework is presented which can instantaneously analyze the energy harvesting ability of the nodes. However, the assumption that energy changes at the node level are broadcast immediately to the neighbours is not realistic because of considerable control overhead this would incur and limited transmission range. The authors in [150] combine geographical information and energy harvesting information to find an energy efficient route to a destination. However, the need for a GPS chipset si not realistic for many multi-hop wireless networks.

Therefore, [151] proposes an approach called a Distributed Energy Harvesting Aware Routing Algorithm (DEHAR) to solve the above issues. DEHAR does not need geographical information of the nodes but only calculates the consumption and production of energy inside a node, which presents the concept of "energy

distance". The "energy distance" is encoded from the spatial distance and it relates the real distance to the energy status (how much energy can be consumed inside a node and harvested from the surroundings) of the sending node. Finally, the route with the shortest energy distance is selected as the final route.

DEHAR only considers the energy harvesting ability of nodes when selecting a route, but does not proposes a route selection algorithm with integrated factors such as cooperative links, link break probability and the forth.

2.4.10 AODV-EHA

In order to consider integrated factors when selecting a route, [152] proposes a solution called Energy Harvesting Aware Ad hoc On-Demand Distance Vector Routing Protocol (AODV-EHA). As analyzed in [152], two aspects in MANETs are considered: topology changes and energy status. AODV-EHA inherits the advantages of AODV to deal with the first aspect. As AODV only requires knowledge of the network topology when it needs to send data, it has an effective response to the topology changes. For the second aspect, AODV-EHA takes into account external energy sources which can be harvested from the surrounding environment [153][154][155][156]. AODV-EHA considers the energy harvesting ability of all nodes and tries to find a route with the least transmission cost by replacing the "hop count" in AODV with an "energy count". Here, the energy count can be obtained by predicting the average transmission cost if forwarding a data packet successfully from the sending node to the receiving node. The authors later compare the AODV-EHA with DEHAR [151], which involves the concept of "energy distance" when measuring the energy status. The route with the shortest energy distance is selected as the final route. Finally, [152] concludes that although AODV-EHA usually finds a longest route compared with AODV and DEHAR, it has the minimum energy consumption along the determined route.

Nevertheless, the features of AODV inherited by AODV-EHA are not the appropriate in many cases. For example there is no special mobility handle scheme in AODV, once the link break happens, the source node will initiate the route discovery again which lead to considerable control overhead.

Other energy-aware routing protocols have been well studied like [157][158][159] [160]. Most of them do not consider the classic route metric of the minimum hop count but only consider energy-related metrics such as the energy requirement

to communicate over a link [157][158], the remaining energy [159] or both [160]. Therefore, the research direction explored in [134] which suggests that more integrated factors need to be involved when selecting the final route is still to be achieved.

2.5 Summary

Though various routing schemes for MANETs are actively discussed in terms of cooperative communication and energy awareness, a fundamental routing structure for cooperative communication is overlooked. What is more, how to improve the robustness against mobility and consideration for energy efficiency, especially with cooperative communication is still a neglected subject. CRCPR provides a complete systematic cooperative routing scheme including cooperative route discovery, route reply, route enhancement, route selection, cooperative data forwarding. In addition, a new route selection criterion which considers integrated factors: energy consumption, energy harvesting and link break probability, is proposed to determine the selection of final route with economic energy consumption and high robustness resulting from mobility induced link breaks.

Chapter 3

CRCPR Design

This chapter provides the details of CRCPR Design. At the beginning, a protocol overview and certain necessary assumptions are given. Then, the main aspects of the CRCPR design are described in the following sections: The establishment of COP Table and Relay Table in the COP topology, which plays fundamental roles in the protocol, are introduced in the Neighbour Discovery section. The route discovery procedure, concerning how the COP topology information is carried via route discovery messages, is explained in the Route Discovery section. The route set up procedure, explaining how route request messages are replied to, is then addressed in the Route Reply section. Considering the COP topology, how data is forwarded in CRCPR is introduced in Data Forwarding section. The Route Enhancement section describes the methods to improve the robustness against mobility. Finally, the route selection algorithm which can exploit COP topology to find a more stable route is described in the Route Selection Criteria section.

3.1 Protocol Overview

In general, CRCPR (Constructive Relay based CooPerative Routing) is a reactive routing protocol with proactive local enhancements in MANETs. It is reactive in the sense that a route is built only when data needs to be sent and proactive in the sense that the COoPerative (COP) topology is set up in advance for all source-destination pairs within the MANET and is locally self-managed.

3.2 Protocol Assumptions

CRCPR is designed to work as a cross-layer scheme that can be built atop of off-the-shelf wireless networking equipment and mainly focuses on ad hoc networks with low rates of mobility. Some necessary assumptions are provided below:

- All the links between transmitter and receiver are the symmetric.
- The transmission of cooperative nodes are synchronized and the power level of the receiving signal at the receiving node is the sum of all the incoming signal powers.
- Some mechanism for error detection is deployed in the message formats.
- No interference or collision happens between different wireless channels.

3.3 Protocol Design

3.3.1 Message Format

All the message formats consist of two parts: the common fields and the type-specific fields. Every message type shown in Figure 3.2 has the same common fields in Figure 3.1 but different type-specific fields. The common fields carry unified formats common to all the message types and the type-specific fields, if any, carry specific format that is relevant to particular functions.

	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	*Version*				*Type*				*Reserved*								*Length*															
4	32	*Src*																															
8	64	*Dest*																															

Figure 3.1: Common Fields in CRCPR

Version: 4-bit length. Identifies the version of the current routing protocol.

Type: 4-bit length. Indicates the different message types referred to in Figure 3.2

Type	Message	Full Name
0	CREQ	Cooperative route REQuest
1	CREP	Cooperative route REPly
2	CLRP	Cooperative Local RePair
3	CENF	Cooperative Error NotiFication
4	CHLO	Cooperative HeLlO
5	CCON	Cooperative CONfirm
6	DATA	DATA

Figure 3.2: Message Types in CRCPR

Src: 32-bit length. This represents the IPv4 address of the source node.

Dest: 32-bit length. This represents the IPv4 address of the destination node. If a message needs to be broadcast, a specific broadcast IP address will be set in this field instead of the IP address of the destination.

Length: 16-bit length. This indicates the length of the whole messages.

Reserved: The length and content of the reserved field may vary according to the type of messages.

3.3.2 Neighbour Discovery

3.3.2.1 Cooperative Hello Generation

Every HELLO_INTERVAL in milliseconds, nodes need broadcast a Cooperative HeLlO (CHLO) message to inform their neighbours about their existence. Figure 3.3 shows the CHLO message format:

	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	*Version*				*Type*				*Reserved*								*Length*															
4	32	*Src*																															
8	64	*Dest*																															
12	96	*Neighbor address [1]*																															
16	128	...																															
20	160	*Neighbor address [n]*																															

Figure 3.3: CRCPR CHLO Message format

This CHLO message format follows the classic hello message structure used in most MANET routing protocols but with an added field called *Neighbour ad-*

dress [n]. Before a node broadcasts a CHLO message, all its neighbour addresses that have already been recorded through previous CHLOs need to be added in the *Neighbour address [n]* field, which means that the CHLO does not only announce its own existence but its neighbours' presents. When its neighbours receive the CHLO message, each neighbour node will create or update the Cooperative Neighbour Table.

3.3.2.2 Cooperative Neighbour Table

- Cooperative Neighbour Table Creation

Once a node receives a CHLO message from its neighbours, the Cooperative Neighbour Table can be built based on the collected information as shown in Figure 3.4. Two new items are added in the Cooperative Neighbour Table compared with the traditional Neighbour Table: the *NSN Addr List* field and the *B/U* field. Each Neighbour'S Neighbours (NSN) is filled in the corresponding *NSN Addr List* field, which facilitates building the Cooperative (COP) Table and maintaining the COP topology. *B/U* marks whether an incoming CHLO, which updates a given entry, is received via a broadcast or unicast message. As with most classic routing protocols in MANET, broadcasting is the common transmission method for its hello messages, whilst unicast is only employed by cooperative nodes and relay nodes when the COP and Relay Tables are being created in CRCPR. Further details are provided in Section 3.3.6.

Neighbour Addr	NSN Addr List	B/U

Figure 3.4: Cooperative Neighbour Table

The example of Cooperative Neighbour Table creation takes place as follows:

1. Assume that Intermediate Node 1 (IN1) and IN2 are not aware of each other at the beginning, so IN1 has two neighbours b and IN3 while IN2 has two neighbours e and IN4.

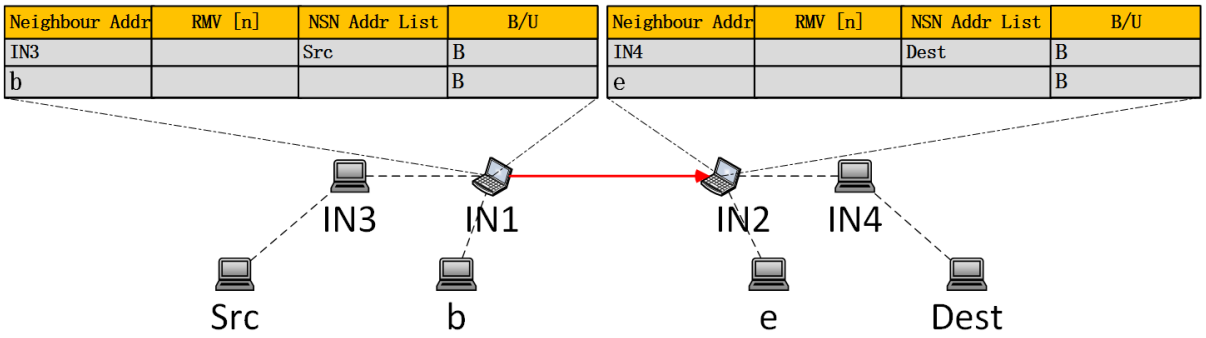


Figure 3.5: Current State of Cooperative Neighbour Table

2. IN1 sends a CHLO. (For simplicity, we ignore the CHLO reception of node IN3 and node b but only focus on the CHLO reception of IN2)

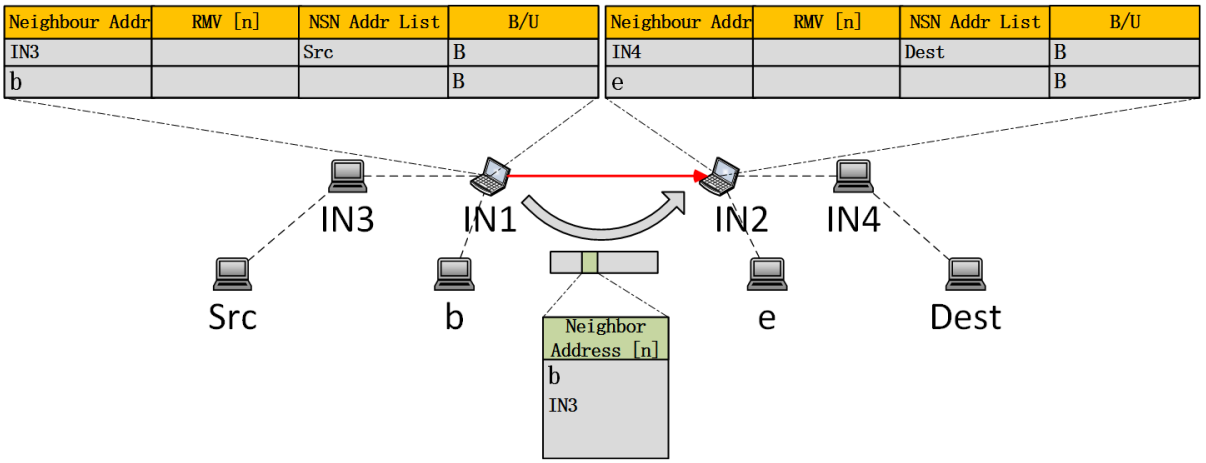


Figure 3.6: IN1 Sends CHLO

3. When IN2 receives the CHLO, it updates its Cooperative Neighbour Table, putting IN1 in the Neighbour Addr field together with b and IN3 in the NSN Addr List. Because the CHLO message is broadcast by IN1, the tag "B" which means "broadcast" is added in the entry. The unicast CHLO with tag "U" in the entry will be introduced in Section 3.3.6.1.

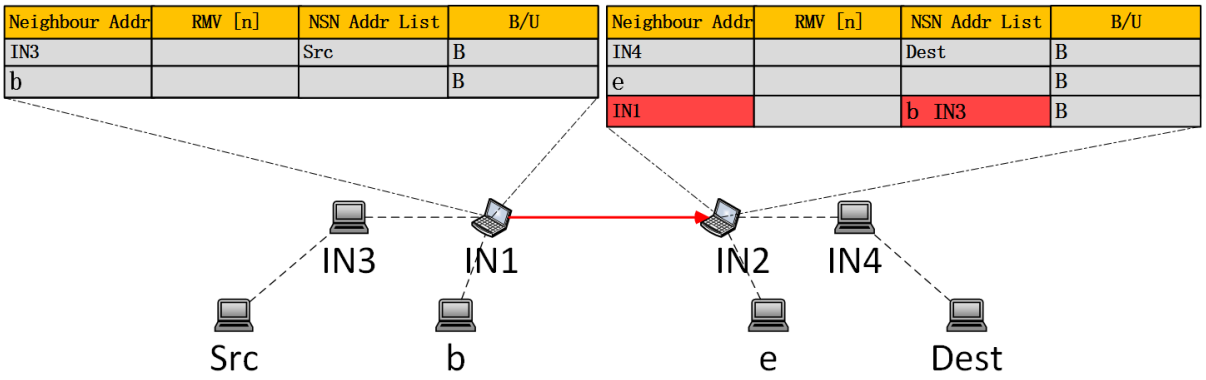


Figure 3.7: IN2 Updates Cooperative Neighbour Table

4. IN2 then sends a CHLO. (For simplicity, we ignore the CHLO reception of node IN4 and node e but only focus on the CHLO reception of IN1)

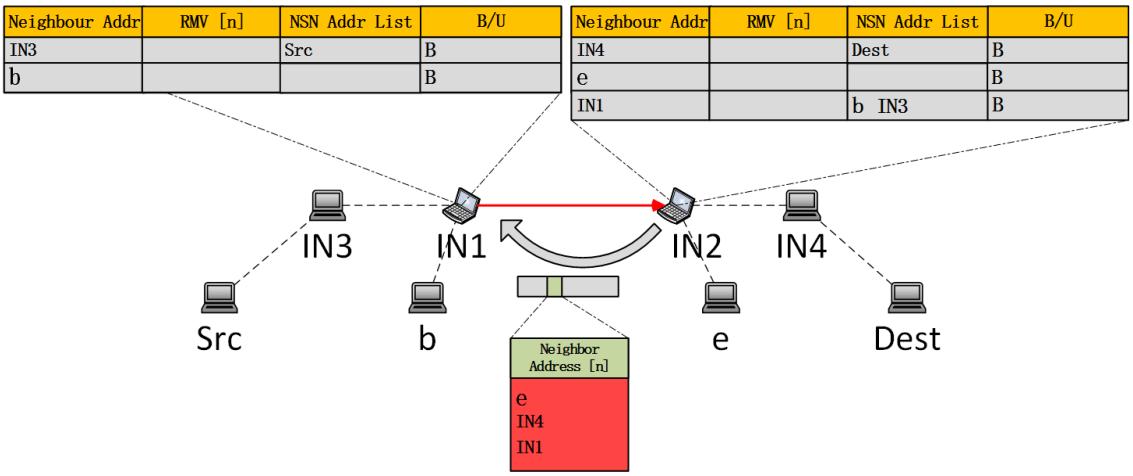


Figure 3.8: IN2 Sends CHLO

5. IN1 receives IN2's CHLO and deletes its own IP address in the Neighbour Address [n] field. IN1 then updates its Cooperative Neighbour Table, placing IN2 in the Neighbour Addr field together with e and IN4 in NSN Addr List. After that, the Cooperative Neighbour Tables of IN1 and IN2 are established.

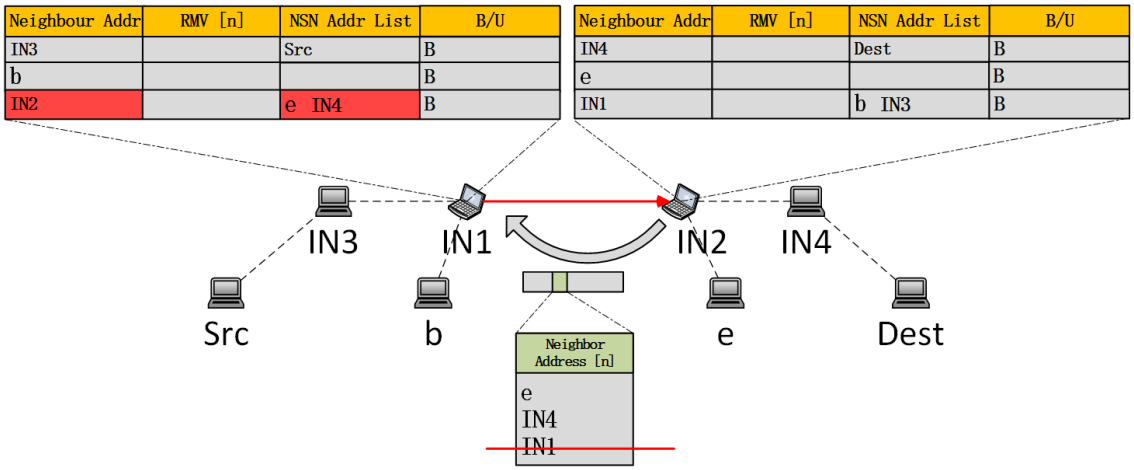


Figure 3.9: IN1 Updates Cooperative Neighbour Table

- Cooperative Neighbour Table Deletion

CRCPR uses ALLOWED_HELLO_LOSS * HELLO_INTERVAL in milliseconds to wait for notification from neighbours. After an entry is added into the Cooperative Neighbour Table, a timer will be set to indicate its validation. If the node receives another CHLO message from the neighbour which is already in the Cooperative Neighbour Table before its timer runs out, it will reset the timer for this entry and the *NSN Addr List* field will also be updated. Otherwise, the entry will be deleted once the timer expires.

3.3.2.3 COP Table

- COP Table Creation

As long as a node learns through its Cooperative Neighbour Table (with the information in the *Neighbour Addr* and *NSN Addr List* fields) that there exist two neighbour nodes that are also a common neighbour to another node via the COP Possibility Detection Algorithm, shown in Table 3.3.2.3, a four-nodes COP topology is formed as illustrated in Figure 3.10. The reason why we choose four-node to form the COP topology is because the lower layer mechanism for this form of cooperative transmission is well understood, whilst the technical challenges regarding frame synchronization for cooperative communication are lessened. Furthermore, this approach is not restrictive, as many four-node COP topologies can coexist within a single MANET. This provides ample opportunity to save energy and improve robustness.

CRCPR is proactive and self-managed in terms of the COP topology, which means the COP Table in Figure 3.11, used to maintain COP topology, can be constructed based only on the Cooperative Neighbour Table instead of route establishment. The four Intermediate Nodes (INs) have specific roles in the COP Table (We use IN to name the nodes between the Src and the Dest along the route which is referred to in [44]). The COP Source (Src), as the instigator of the COP topology decides the role assignment and initiates the local COP Table. The specific proactive principle is as follows: along the route, the first node within the COP topology receiving valid data will be regarded as the COP Src and the other INs will be assigned roles according to the COP Table of the COP Src via a Cooperative CONfirm (CCON) message shown in Figure 3.12. More precisely, before the COP Src forwards data, it chooses a suitable entry from its COP Table list and places this entry in a CCON message which will be used to notify the proper Cooperative (C) nodes to prepare to transmit data cooperatively and the appropriate COP Destination (Dest) to combine the cooperative data signals. After the COP Table is confirmed across the four INs in the COP topology, the data forwarding procedure commences. The details about data forwarding via COP topology will be introduced in Section 3.3.5.

Sometimes, there are several COP topologies between two hops along a route. Only the COP topology activated via a CCON message will participate in cooperative communication and the others remain "silent". The CCON message will not be sent again to trigger activation of a "silent" COP topology until the previously activated one ceases.

COP Possibility Detection Algorithm

```
//Let NA_i be neighbour addresses in each IN.
//Let NSN_j be each list of neighbours' neighbour addresses.
//Let L_(i,j) be each neighbours' neighbour addresses in one NSN_j.
1. begin
2.     For each i in NA_i
3.         begin
4.             For each j in NSN_j
5.             *Find neighbours' neighbour address*
6.             Obtain L_(i,j)
7.             begin
8.                 For each i + 1 in NA_i
9.                 begin
10.                    For each k in NSN_j
11.                    *Find neighbours' neighbour address*
12.                    Obtain L_(i+1,k)
13.                    *Compare neighbours' neighbour address*
14.                    if (L_(i,j) == L_(i+1,k))
15.                        *Insert a new entry in COP Table*
16.                        Add own IP address in COP Src
17.                        Add L_(i,j) or L_(i+1,k) in COP Dest
18.                        Add two NA_i in C Node
19.                    else (L_(i,j) == L_(i+1,k))
20.                        *Start the next loop*
21.                        continue
22.                end
23.            end
24.        end
25. end
```

The COP Possibility Detection Algorithm starts when the Cooperative Neighbour Table updates (Line 1). Then, according to the "neighour address" item saved in the entry of the Cooperative Neighbour Table, the algorithm will find all the corresponding "Neighbour'S Neighbours (NSN) addresses " (From Line 2 to line 6). By comparing the NSN value in different entries, once a same NSN value exists in two different entries, which means the current node has two neighbour nodes that are also a common neighbour to another node (From Line 10 to Line 18), the COP Table will be constructed in the current node. Otherwise, a next detection loop commences (from Line 19 to Line 21). The COP Possibility Detection Algorithm does not finish until the "neighbour address" in all the entries of the Cooperative Neighbour Table have been detected.

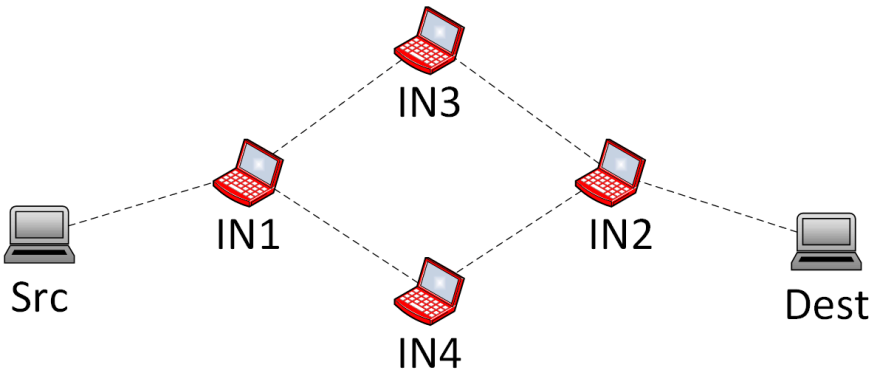


Figure 3.10: COP Topology

COP Src	COP Dest	C Node 1	C Node 2

Figure 3.11: COP Table

	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	*Version*				*Type*				*Distance*				*Total hop*				*Length*															
4	32	*Src*																															
8	64	*Dest*																															
12	96	*COP Src*																															
16	128	*COP Dest*																															
20	160	*C Node 1*																															
24	192	*C Node 2*																															

Figure 3.12: CRCPR CCON Message Format

The example of COP Table creation mechanism operates as follows:

1. Assume all the Cooperative Neighbour Tables of the INs have been established.

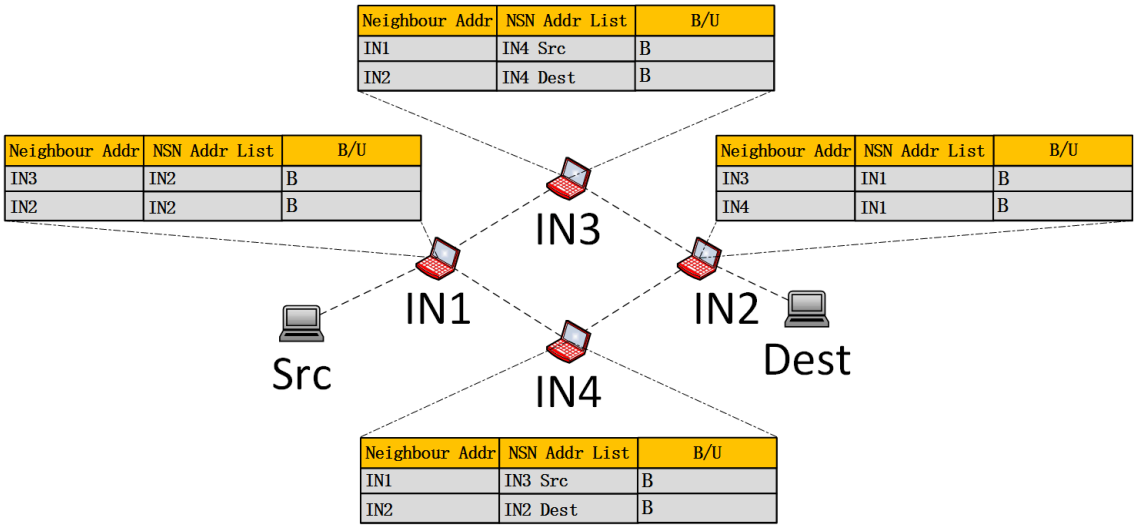


Figure 3.13: Current State of Cooperative Neighbour Table

2. Only the entries in the Cooperative Neighbour Tables with the value "B" of *B/U* fields are saved to build a temporary COP Driven Table shown in Figure 3.14. COP Drive Table is a temporary intermediate table between the Cooperative Neighbour Table and the COP Table and its purpose is to turn the Cooperative Neighbour Table to the COP Table via the COP Possibility Detection Algorithm.

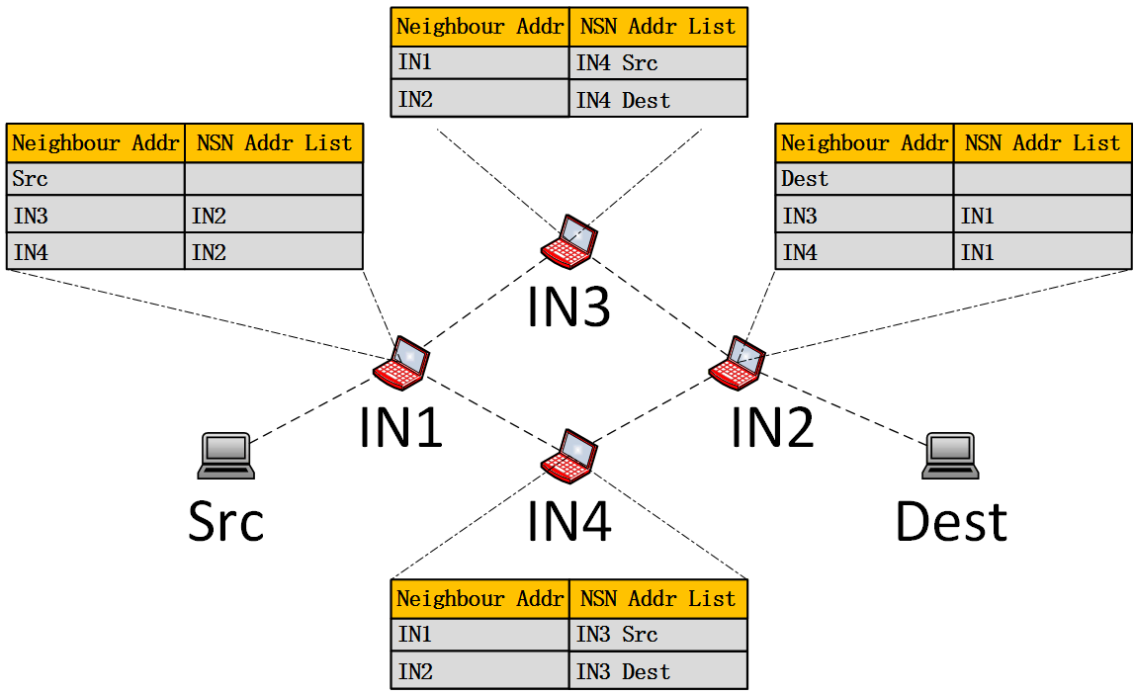


Figure 3.14: COP Driven Table Creation

3. Run the COP Possibility Detection Algorithm based on the COP Driven Table.

4. After every Intermediate Node (IN) completes the COP Possibility Detection Algorithm, they establish the COP Table.

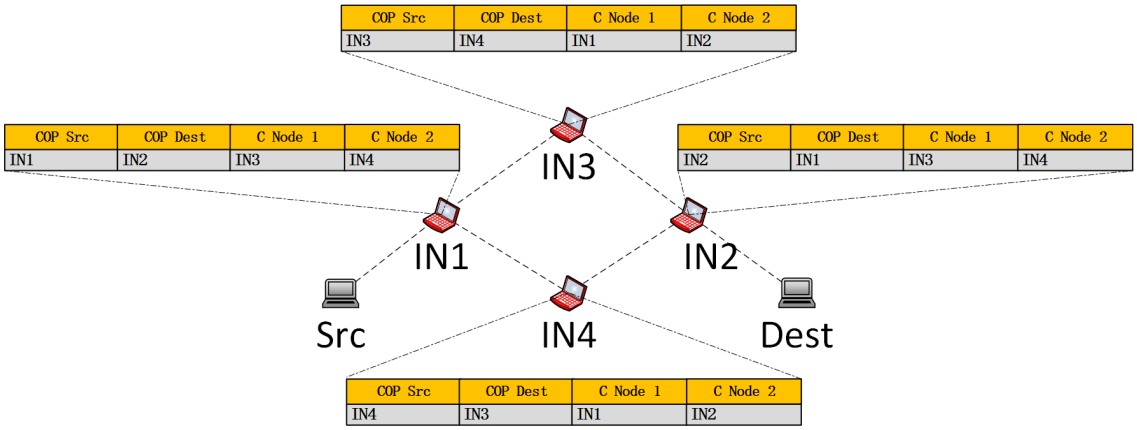


Figure 3.15: COP Table Creation

- COP Table Deletion

Once a node receives a CHLO message, it runs the COP Possibility Detection Algorithm to check the existence of the COP topology. In addition, the COP Possibility Detection Algorithm is also responsible for the deletion of invalid entries in the COP Table. There are five cases which can result in the deletion of an entry from the COP Table.

1. One Link Break:

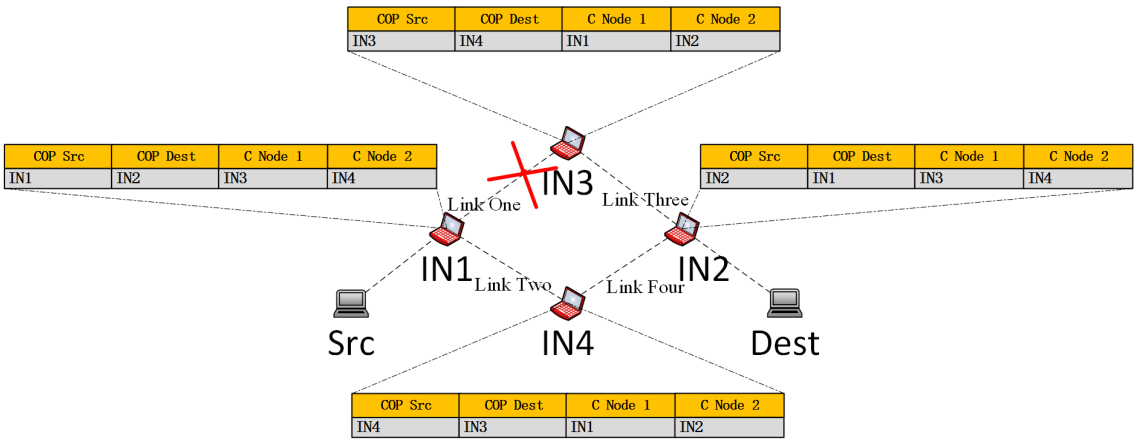


Figure 3.16: One Link Break

An example is shown in Figure 3.16. After ALLOWED_HELLO_LOSS * HELLO_INTERVAL milliseconds, because IN1 does not receive CHLO message from IN3 and IN3 does not receive CHLO message from IN3, Link One in Figure 3.16 breaks. Based on the deletion principle for the Cooperative Neighbour Table in Section 3.3.2.2, IN1 deletes IN3 and IN3 deletes IN1 from their Cooperative Neighbour Tables, respectively. Also, IN2 and IN4 will find Link One is broken by learning the *Neighbour address [n]* field in the CHLO message from IN3 and IN1, respectively. So the Cooperative Neighbour Tables of all four INs of this COP topology will be updated. This leads to the deletion of the invalid entry in their COP Tables. The deletion principle is as follows: Every IN checks if the broken link is between its COP Src and C node 1 or C node 2. If true, the IN deletes the entry directly such as IN1 and IN3 in this example. If the broken link is between its COP Dest and C node 1 or C node 2, the IN will create a Relay Table based on this entry and then delete it, as in the case of IN2 and IN4. The creation of the Relay Table will be introduced in Section 3.3.2.4.

2. Two Diagonal Links Breaks

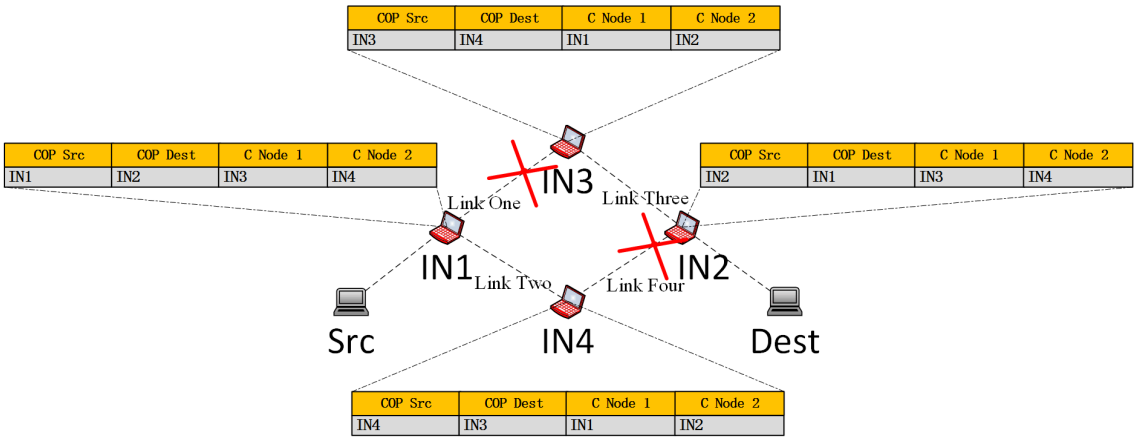


Figure 3.17: Two Diagonal Links Breaks

An example is shown in Figure 3.17. After ALLOWED_HELLO_LOSS * HELLO_INTERVAL milliseconds, because IN1 does not receive CHLO message from IN3 and IN3 does not receive CHLO message from IN3, Link One in Figure 3.17 breaks. Similarly, IN4 and IN2 cannot receive the CHLO message from each other. In addition, IN1 and IN3 will find that Link Four

is broken and IN2 and IN4 will find that Link One is broken with the information in the *Neighbour address [n]* field in the CHLO message. So the Cooperative Neighbour Tables for all four INs of this COP topology will be updated, which leads to the deletion of the invalid entry in their COP Table. The deletion principle is: Every IN checks if the broken link is between its COP Src and C node 1 or C node 2. If true, the IN will delete the entry directly. In this particular case, all four INs just delete the entry in their COP Tables directly.

3. Two Neighbour Links Breaks

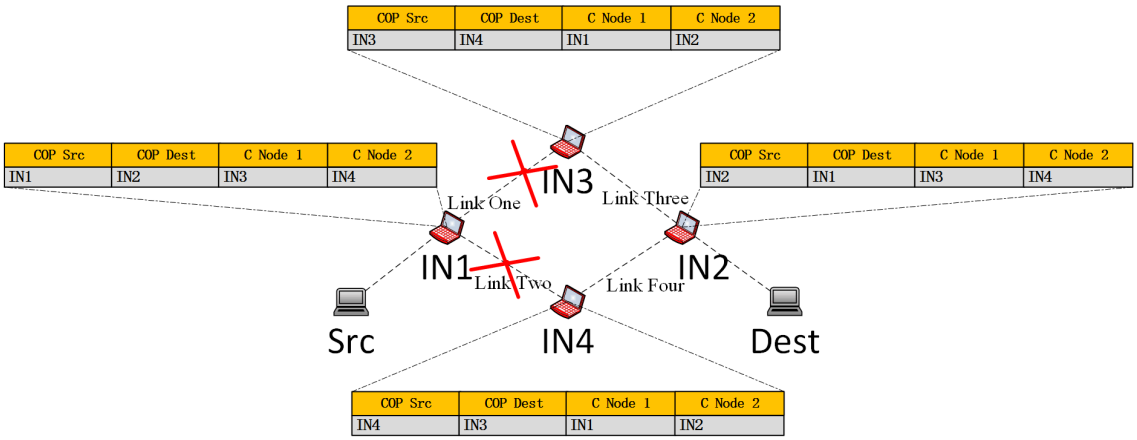


Figure 3.18: Two Neighbour Links Breaks

An example is shown in Figure 3.18. After ALLOWED_HELLO_LOSS * HELLO_INTERVAL milliseconds, IN1 will find the CHLO message from IN3 and IN4 cannot be received. Furthermore, IN3 and IN4 cannot receive the CHLO message from IN1. Also, IN2 will find that Link One and Link Two are broken by interrogating the *Neighbour address [n]* field in the CHLO message from IN3 and IN4, respectively. So the Cooperative Neighbour Tables for all four INs of this COP topology will be updated, which leads to the deletion of the invalid entry from their COP Table. The deletion principle is as follows: Every IN checks if the broken link is between its COP Src and C node 1 or C node 2. If true, the IN will delete the entry directly such as IN1, IN3 and IN4 in this example. If the broken link is between its COP Dest and C node 1 or C node 2, the IN will create a Relay Table (details are described in Section 3.3.2.4) such as IN2 in this example.

4. Three Links Breaks

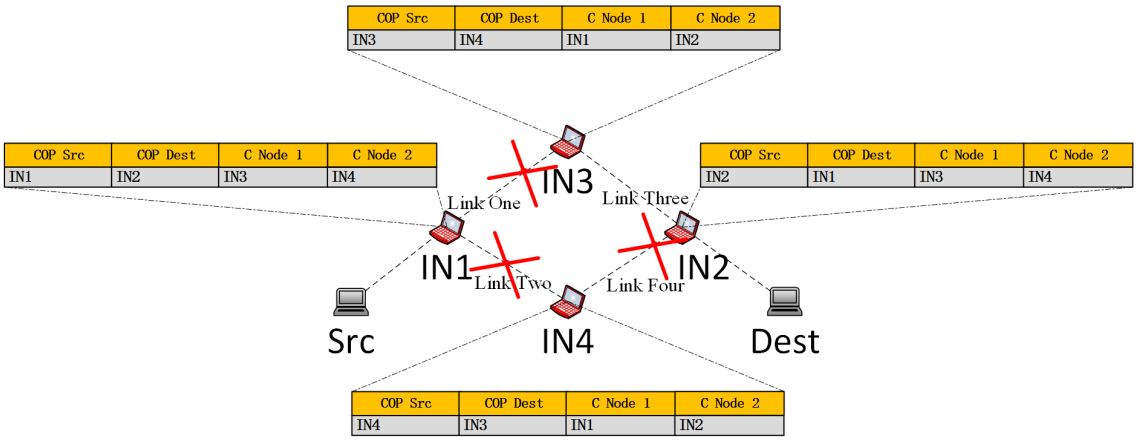


Figure 3.19: Three Links Breaks

If this case arises, as in Figure 3.19, according to the above analysis about Two Neighbour Links Breaks, every IN should delete the entry directly.

5. Four Links Breaks

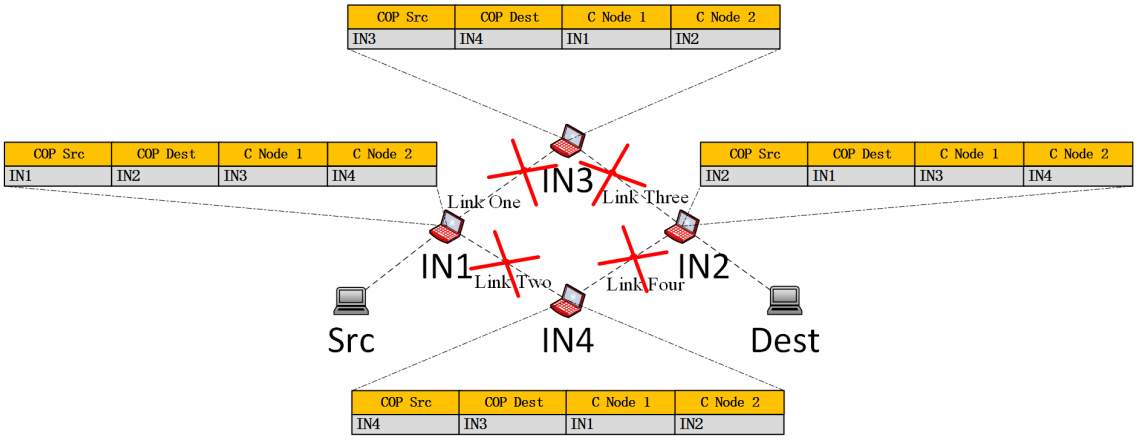


Figure 3.20: Four Links Breaks

The case in Figure 3.20 is the same with regard to Three Links Breaks.

3.3.2.4 Relay Table

Relay Neighbour 1	Relay Neighbour 2

Figure 3.21: Relay Table

When the Cooperative Neighbour Table is updated by a CHLO message, every IN runs the COP Possibility Detection Algorithm to update the COP Table or delete invalid entries if the COP topology no longer exists. If an entry is deleted from the COP Table but the C nodes in this entry remain neighbours, a Relay Table will be built.

- Relay Table Creation

As illustrated in Figure 3.21, the elements in the Relay Table: Relay Neighbour 1 and 2 are the IP addresses IN1 and IN2, respectively. An example considering the Relay Table creation of IN2 and IN4 is given in Figure 3.22: if the link between IN1 and IN3 is broken, IN1 and IN3 will delete each other from their Cooperative Neighbour Tables. IN2 and IN4 are notified that IN1 and IN3 are no longer neighbours via the *Neighbour address [n]* field in CHLO message. Then, according to the COP Table, IN1 and IN3 understand that the broken link is between their COP Src and C node. Therefore, they delete the corresponding entry in the COP Table directly. IN2 and IN4 realize that the broken link is between their COP Dest and C node. Therefore, they delete the entry in the COP Table and create an entry in the Relay Table.

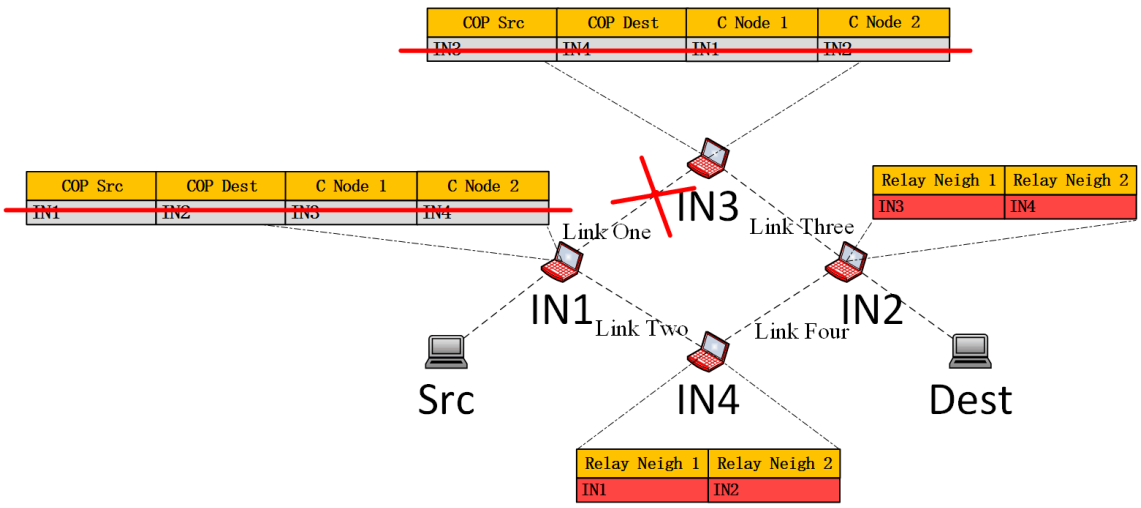


Figure 3.22: Relay Table Creation

- Relay Table Deletion

We have two methods to delete the entry in the Relay Table: soft deletion and hard deletion. The soft deletion is to use the timer of the entry. Once an entry in

the Relay Table has been used to relay data, the timer will be reset. However, if the entry has not been used for a long time and the timer runs out, this entry will be deleted automatically. The hard deletion is to use the Cooperative Neighbour Table. If a CHLO message from any one of two Relay Neighbours (Relay Neighbour 1 and Relay Neighbour 2) cannot be received by the relay node for a long time (ALLOWED_HELLO_LOSS * HELLO_INTERVAL), the entry in the Relay Table will be deleted.

3.3.3 Route Discovery

3.3.3.1 CREQ Generation

Initially, all nodes except the neighbours of DEST do not have the route to DEST. When a node requires a route to a destination it must broadcast a Cooperative route REQuest (CREQ) message to seek a route across the MANET.

- CREQ Format

	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	*Version*				*Type*				*Hop_No*				*Reserved*				*Length*															
4	32	*Src*																															
8	64	*Dest*																															
12	96	*SEQ_No*																															
16	128	*IN Address [0]*																															
20	160	*IN Address [1]*																															
24	192	*RMV [0]*																															
28	224	*IN Address [2]*																															
32	256	*RMV [1]*																															
36	288	...																															
40	320	*IN Address [n]*																															
44	352	*RMV [n-1]*																															

Figure 3.23: CRCPR CREQ Message Format

The CREQ message is of variable length, as follows:

Hop_No: 8-bit length. This represents the number of the hops from the source node to the node handling the CREQ.

SEQ_No: 32-bit length. This field will be set with a standard Unix time called "timestamp" and used to uniquely identify the particular CREQ with the help of *Src* and *Dest* fields, which is called "source-dest triplet"

$< SRC, SEQ\_No, DEST >$. (The standard Unixtime is asigned integerdata type, traditionally of 32bits)

IN Address [n]: 32-bit length for each IN IP Address. The IN which receives the CREQ will add its IP address to this field.

Routing Matrix Values (RMV): 32-bit length. RMV contains the node performance data passed from the Physical Layer and MAC Layer, such as the Battery Capacity Degree, Energy Accumulation Rate, Real-time Residual Energy Degree, Link Break Probability and so on. All this information contributes to the final route selection which will be introduced in Section 3.3.9.

- CREQ Configuration

  Initially, the *Hop Count* field is set zero by the source node, then the node receiving CREQ should add HOP_INCREMENT to the Hop Count field. The *SRC* field is set to the IP address of the source node, and the *DEST* field is set to the IP address of the destination node. The *SEQ_No* field will be set to a timestamp to uniquely identify each CREQ message via "source-dest triplet" $< SRC, SEQ\_No, DEST >$. *IN Address [n]* field will be set to the IP address by the IN nodes which will rebroadcast the CREQ message. Also, the IN nodes should add the RMV to the *Routing Matrix Values [n]* field.

3.3.3.2 CREQ Broadcasting

- Route Request Table

  Before the CREQ message is broadcast by the source node, a Route Request Table should be established as shown in Figure 3.24. The items of the *Src*, *Dest* and *Timestamp* fields are filled with the value of *Src*, *Dest* and *Seq_No* in the CREQ message. If the first Routing Discovery attempt does not receive any valid reply by comparing the ROUTE_REQUEST_TABLE_TIMER with the *Timestamp* value in the Route Request Table, the source node will rebroadcast a new CREQ message and update the *Timestamp* field. Due to different application requirements, if the source node intends to establish another route to a different destination, it will add a new entry into the

Route Request Table. The Route Request Table in the source node is used to make sure that only the valid route reply information can be processed and indicates the success of the Route Discovery. The Section 3.3.4 considers how to handle the route reply information.

Figure 3.24: Route Request Table

Every IN receiving the CREQ message will also build a Route Request Table. If the value of *Src* or *Dest* in a CREQ message is different from the entry in Route Request Table, a new entry will be added and the CREQ message will be processed. When the value of *Src* and *Dest* in the CREQ message are the same with entry in the Route Request Table, if *Seq_No* in the CREQ is newer than the *Timestamp* field in the corresponding entry, the CREQ message will be processed. More precisely,

1. A CREQ carrying a different SRC from the entry saved in Route Request Table in the INs will be rebroadcast.
2. A CREQ carrying a different DEST from the entry saved in Route Request Table in the INs will be rebroadcast.
3. A CREQ carrying the same SRC and DEST, but a newer SEQ_No will be rebroadcast.

Otherwise, the CREQ message will be discarded.

- CREQ Handle

  The CREQ message handling procedure, including broadcasting across both non-COP and COP topologies is shown in Table 3.3.3.2. It is designed to allow the RMV, if any, to be carried in a CREQ message to the destination therefore contributing to the final route selection.

  In a non-COP topology, the flow chart of the CREQ handle procedure is referred to in Figure 3.25 [1]. When an IN receives a CREQ message, it firstly checks if its own IP address exists in the *IN Address [n]* field. If true, discard the CREQ message; otherwise it checks if this CREQ matches with the entry

[1]The shaded boxes represent CREQ packet discard.

saved in its Routing Request Table. If not, it discards the CREQ message; otherwise, it checks if the address of *the immediate upstream node of the last hop* [2] is one of its neighbour nodes. If yes, it discards the CREQ message. If not, it appends its own IP address in the *IN Address [n]* field. Also, the field of Hop_No should be added with HOP_INCREMENT. Finally, the CREQ will be rebroadcast by the IN to its neighbours (if it has any).

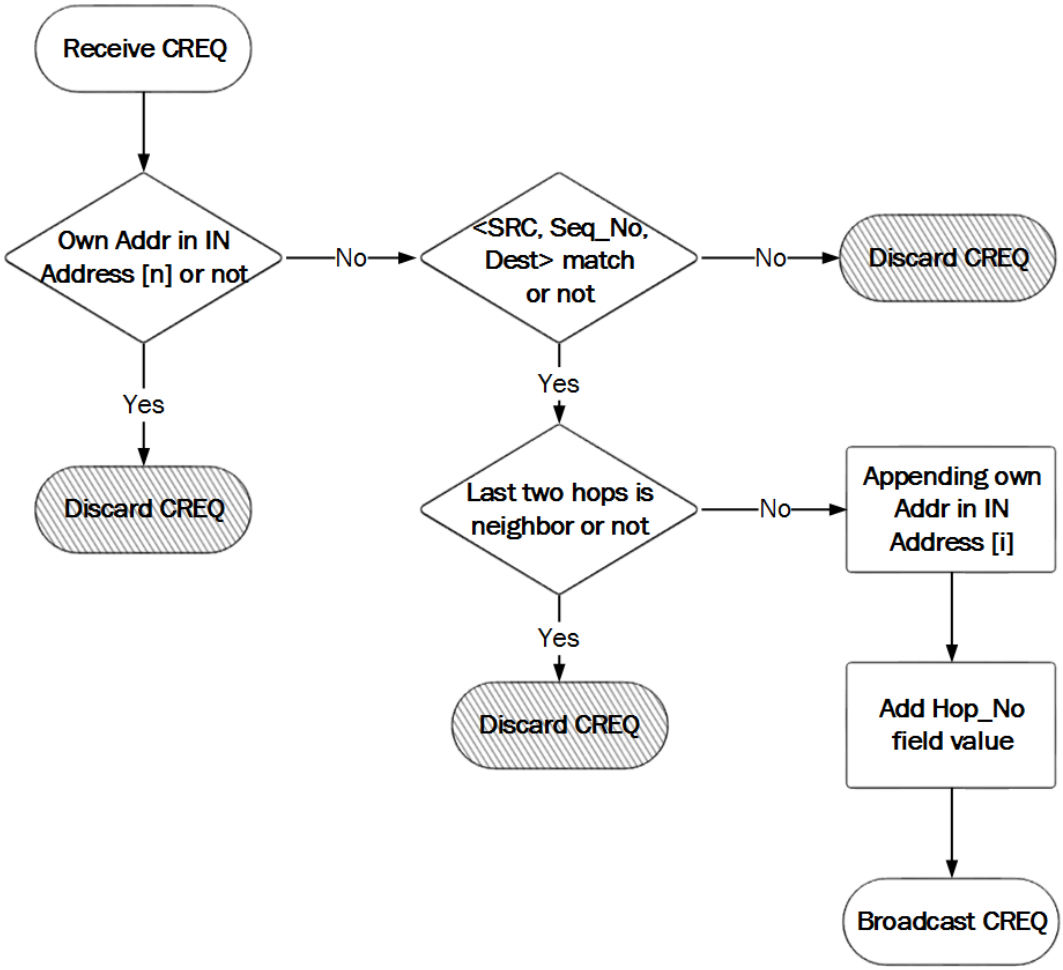


Figure 3.25: The Flow Chart of CREQ Handle in the Non-COP Topology

In the COP topology, once an IN receives a CREQ message and finds that *the last two hops* is the COP Dest in its COP Table, "last hop replacement" scheme will be triggerred; that is the IN replaces the last hop IP address in the CREQ message IP list with its own IP address before re-broadcasting it to its neighbours which can make the location of COP Dest closer to the destination and reduce the total hops in the final route. Furthermore,

[2] *"The immediate upstream node of the last hop"* will be refer to as *"the last two hops"* in the following contents.

the "last hop replacement" leads to the C nodes invisibility if this COP topology is selected for the final route. The invisibility of C nodes actually results in a virtual point-to-point connection diagonally within the COP topology even though COP Src and COP Dest may not be within each other's direct transmission range. This virtual point-to-point connection not only contributes to saving energy via cooperative diversity, but also improves robustness against mobility. This is because if any one of the C nodes moves away from the COP topology, a Relay Table will be built to maintain the connection between the COP Src and COP dest.

All the CREQ messages are forwarded based on the above procedure until they reach the destination node.

CREQ Handle

```
//Let N_i be the number of entries in the COP Table of one node.
1. if(N_i == 0)
2.    *Broadcast CREQ based on non-cop topology broadcasting principle*
3. else
4.   begin
5.      For each i in N_i
6.        if(COP Dest is last two hop)
7.             if(COP topology set in RMV by last two hop)
8.                 if(Two C nodes of COP Table in the current receiving node
                      are the same with two C nodes set in RMV)
9.                   *Broadcast CREQ*
10.                else
11.                  *Replace old RMV*
12.                  *Replace last hop IP with its own IP*
13.                  *Broadcast CREQ*
14.            else
15.                *Replace old RMV*
16.                *Replace last hop IP with its own IP*
17.                *Broadcast CREQ*
18.       else if(COP Dest is last hop)
19.            if(COP topology set in RMV by last hop)
20.                if(Two nodes in new COP topology are the same with
                      two nodes in old COP topology set by last hop)
21.                  *Broadcast CREQ*
22.                else
23.                  *Set new RMV*
24.                  *Broadcast CREQ*
25.            else
26.                *Set new RMV*
27.                *Broadcast CREQ*
28.       else
29.           *Broadcast CREQ*
30.   end
```

CREQ Handle scheme starts when a CREQ message is received by a node. If there is no COP Table for the current receiving node, the CREQ message

will be processed according to Figure 3.25 (Line 2). If there is a COP table for the current receiving node, it means the current receiving node is an IN in a COP topology and CREQ message will be processed as the following principle: if the COP dest receives a CREQ message, it will perform "last hop replacement" scheme (From Line 6 to Line 17). If any one of two Cooperative (C) nodes receives a CREQ message and there exists a new COP topology between itself and the last hop, which is different from the one saved the in *RMV* field of the received CREQ packet, it will add this COP topology into *RMV* field and re-broadcast the CREQ message (from Line 18 to Line 27). If the COP Src receives a CREQ message, it just broadcasts the CREQ (Line 29).

3.3.3.3 CREQ Loop Avoidance

In CRCPR, we consider the loop avoidance in terms of two types of topology: non-COP topology and COP topology.

In the non-COP topology, the basic principle is to check if the CREQ message has been processed already. More precisely, if its own IP address has been added to the *IN Address [n]* field, it will discard the CREQ message.

In the COP topology, as the IP address of the C node has been removed from the *IN Address [n]* field via the "last hop replacement" procedure, we should prevent the CREQ from being processed again by the C nodes in the COP topology. More precisely, each C node in the COP topology should check if the CREQ has been processed by the COP Src and COP Dest already. If true, this CREQ message should be discarded.

Based on the above procedure, we can avoid CREQ loops in CRCPR and ensure the CREQ messages can reach their destination successfully.

3.3.3.4 RMV Appending

As CRCPR is a cross-layer scheme that can exploit the RMV information passed from the Physical Layer and MAC Layer up to the Network Layer as important factors contributing to the final route decision, during the CREQ message handling procedure, the RMV information needs to be carried in the CREQ messages to destination.

In ABR [44] protocol, the piggybacking of associativity "ticks" in the route

request packet can be used to indicate the stability between links. This is different from the RMV Appending scheme in CRCPR. In ABR, the last hop node appends all the "ticks" information between itself and its neighbours onto the route request message. After the next succeeding neighbour receives the route request packet, it will only retain the "ticks" which are concerned with itself and the last hop node and removes the other irrelevant "ticks" information. However, because the sending node does not know which neighbour will be selected in the final route, all the "ticks" between the current sending node and its neighbours are blindly appended in the route request message, which increases the overhead of the whole network. This process, which will be refer to as "pre-appending" scheme in the following contents, is complicated and does not fit well with CRCPR. Therefore, a new method of piggybacking the RMV information onto CREQ messages is proposed.

In CRCPR, the piggybacking of RMV data onto CREQ messages in the non-COP topology, which will be refer to as "post-appending" scheme in the following contents, is illustrated in Figure 3.26. The source node does not include any RMV data in the CREQ message. It only appends its own IP address in the *IN Address [1]* field and then broadcasts the CREQ to its neighbours (if it has any). Any next succeeding IN will check the IP address of the last hop in the *IN Address [n]* field. Then, the IN only appends the RMV information pertaining to itself and the last hop into the CREQ message. This procedure is also extended to the COP topology. When the CREQ message arrives at an IN that contains entries in the COP Table, if the IN is a COP Src or Cooperative (C) node, it will process the CREQ message using the same procedure as for the non-COP topology. However, if the IN is a COP Dest, it will perform the "last hop replacement" scheme described in Section 3.3.3.2. More precisely, when the IP address of one C node is replaced, its RMV information in the CREQ message will also be replaced. The piggybacking of the RMV data in a COP topology is illustrated in Figure 3.27.

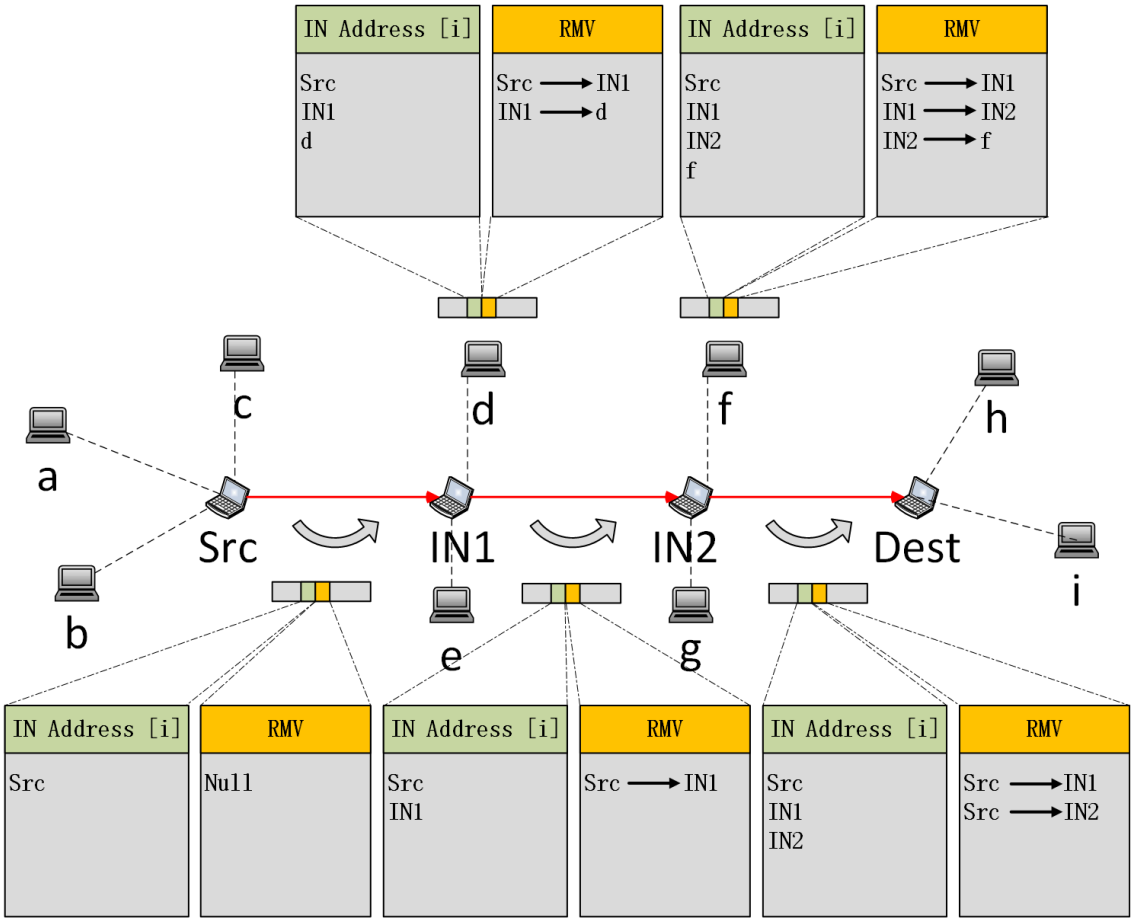


Figure 3.26: RMV Appending in a non-COP Topology

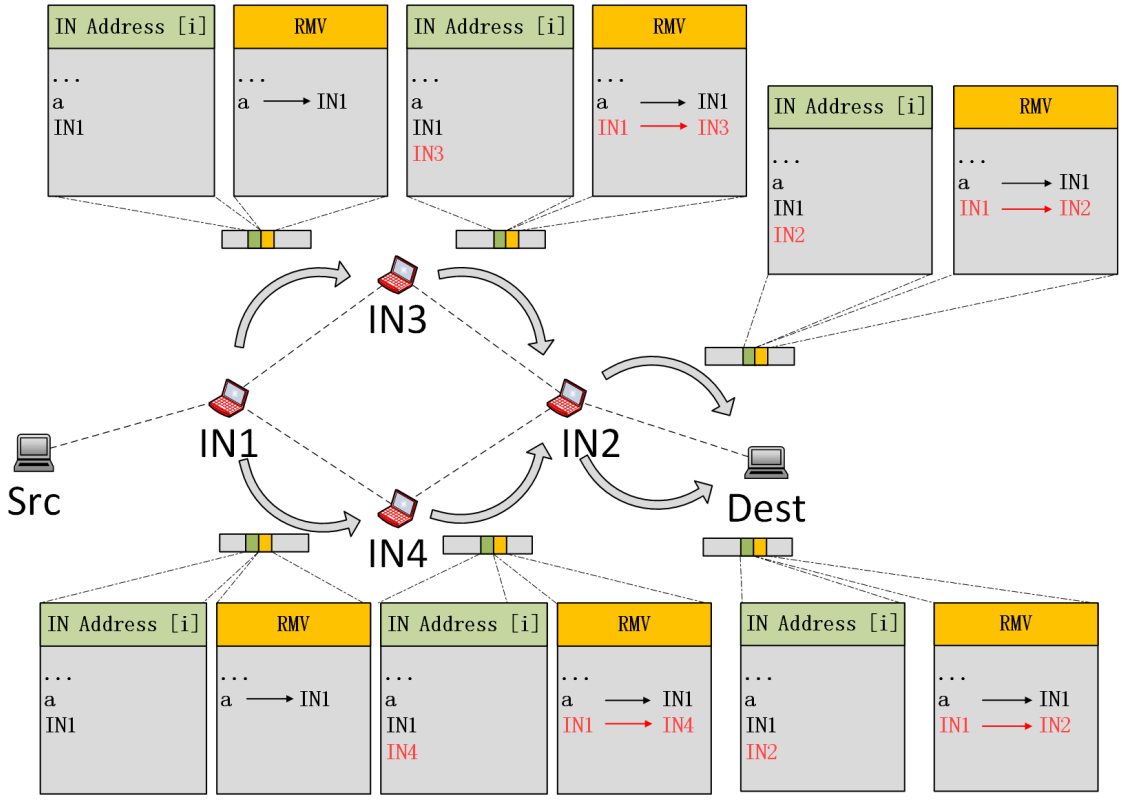


Figure 3.27: RMV Appending in a COP Topology

In summary, the RMV Appending scheme in CRCPR has three features: First, it replaces the "pre-appending" scheme in ABR by a new "post-appending" scheme, which leads to less information being carried in the CREQ message. Second, the data transmission direction is from the source node to the destination node (i.e. from the last hop node to the current receiving node) and the RMV between the current receiving node and the last hop node is measured by the current receiving node to obtain the Channel State Information (CSI) from the last hop node, which matches the data transmission direction. Therefore, as the RMV data is included by the current receiving node, instead of the last hop node, it provides the appropriate information for the final route selection. Third, this RMV Appending scheme can easily be extended to other routing protocols. This may need different RMV information to be collected, though this is for further work.

3.3.4 Route Reply

3.3.4.1 CREP Generation

After receiving the first CREQ message, the destination node waits for a period of time to allow collecting all possible CREQ messages originating from the same source but via different routes. A Cooperative route REPly (CREP) message is then generated with the output of route selection criteria introduced in Section 3.3.9.

- CREP Format

Octet	Bit	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	*Version*				*Type*				*Distance*				*Total hop*				*Length*															
4	32	*Src*																															
8	64	*Dest*																															
12	96	*Last hop*																															
16	128	*Next hop*																															
20	160	*SEQ_No*																															
24	192	*Selected IN Address [1]*																															
28	224	...																															
32	256	*Selected IN Address [n]*																															

Figure 3.28: CRCPR Cooperative Reply Message Format

The CREP message is of variable length, as follows:

Distance: 4-bit length. This represents the hop number from the current node to the destination node.

Total hop: 4-bit length. This represents the total hop number of the selected route.

SEQ_No: 32-bit length. This field is set with the same value of SEQ_No as the CREQ message whose route information is selected for the final route.

Selected IN Address [n]: 32-bit length for each *Selected IN Address [n]*. All the IP addresses of each IN in the final route are set in this field.

- CREP Configuration

When a CREP is generated, the *Distance* field is set to 0 and the *Total hop* field is set to the total hop number of the selected route. *Src* is set to the IP

address of the destination node. *Dest* is set to the IP address of the source node. *Last hop* is the IP address of the destination node and *Next hop* is the IP address of the immediate upstream node of the destination node in the selected route. *Seq_No* has the same value as that of the CREQ message which composed the "source-dest triplet" $< SRC, SEQ\_No, DEST >$. *Selected IN Address [n]* is set to the selected IP addresses of each IN.

3.3.4.2 CREP Forwarding

- Route Table Structure

  Before a destination node sends CREP to its immediate upstream node, a route entry is established in the Route Table as shown in Figure 3.29. *Src* represents the IP address of the source node and *Dest* is the IP address of the destination node. *Last hop* and *Next hop* mean the data ingress node and egress node of the route, respectively. *Distance* means the hop number from the current handling node to the destination node. *Total Hop* is the total hop number of the route. *Timestamp* identifies a timer when the route is built and is updated once the entry is used to forward a data message. *Silent* is a boolean value which can indicate if the route is being repaired in the Local Repair scheme which is introduced in Section 3.3.8. The details are introduced in Section 3.3.8. Finally, the *Next Hop* of the destination node and the *Last Hop* of the source node will be set to "NULL" in their Route Tables.

  If the CREP message reaches an IN and the $< SRC, SEQ\_No, DEST >$ in the CREP message are the same as the values in an entry in the IN's Route Request Table, this IN will add a new entry in the Route Table and unicast the CREP to the next hop. Otherwise, the IN will regard this CREP as invalid and discard it.

Src	Dest	Last Hop	Next Hop	Distance	Total Hop	Timestamp	Silent

Figure 3.29: Route Table

- CREP Handle

A CREP message which contains the IP list of the reverse selected route is unicast from the destination node back to the source node.

In a non-COP topology, when any IN receives a CREP message, the first thing it needs to do is to check whether the *Next Hop* field is its own IP address. If not, it discards the CREP message. If true, it continues to check whether the $< SRC, SEQ\_No, DEST >$ pair matches an entry saved in its Route Request Table. If not, it discards the CREP message. If yes, Route Table will be updated and the CREP is unicast to the next hop. Most fields of CREP message remain unchanged such as *Version*, *Type*, *Total hop*, *Length*,*Src*, *Dest Seq_No* during unicast transmission. The *Last hop* field will be filled with its own IP address. IP address of next hop saved in the newest *Selected IN Address[n]* is removed but placed in the *Next hop* field, which reduces the length of *Selected IN Address[n]* field and the control overhead. The *Distance* indicating the hop number from the current IN to the destination node is incremented by 1. After the CREP message is sent out, the $< SRC, SEQ\_No, DEST >$ triplet saved in the current IN is removed from its Routing Request Table to release the memory. The flow chart of the CREP handle procedure in a non-COP topology is referred to in Figure 3.30 [3].

[3]The shaded boxes represent CREP packet discard in the non-COP topology.

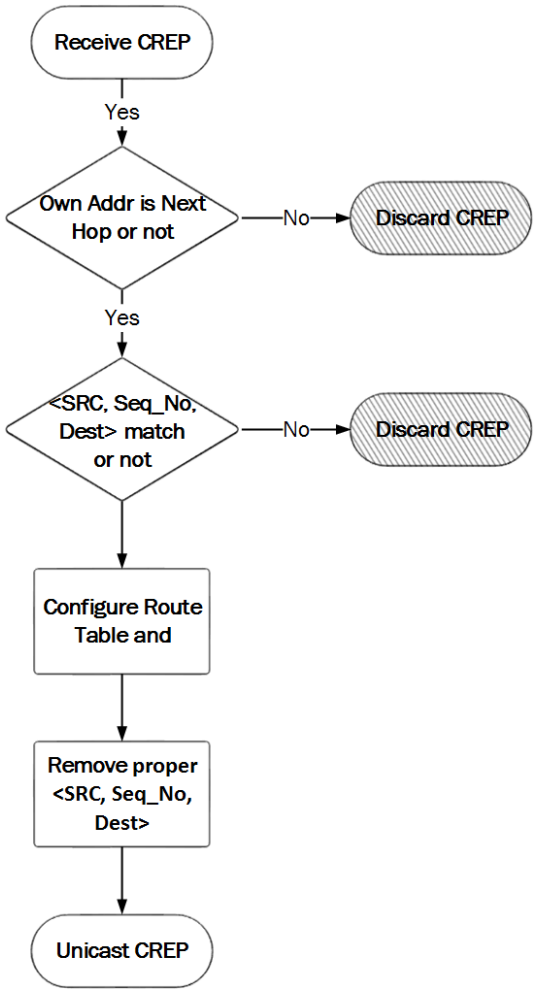


Figure 3.30: The Flow Chart of CREP Handle in the Non-COP Topology

In the COP topology, different IN performs different procedures: (1) for the COP Dest, if the Next hop field in the CREP message is its own IP address, it will unicast CREP to its neighbours. Otherwise, it destroys it. (2) for C nodes, if the connectivity does not exist between the COP Dest and COP Src (the connectivity can be obtained with the knowledge of the *Neighbour Addr in CHLO* and *NSN Addr List* fields), it unicasts the CREP message to the COP Src. However, if connectivity exists, it destroys the CREP message. The reason for this is that due to connectivity between the COP Dest and COP Src, the CREP message from the COP Dest can be received directly by the COP Src and sending it again via C nodes would be redundant. The same procedure arises when the CREP message comes to a node with a Relay Table. The only difference is that the node needs to check whether connectivity exists between its preceding and succeeding Relay Neighbours. (3) for a COP Src, valid CREP messages are unicast to

the next hop. However, if any repeated CREP messages arrive from different C nodes, they are discarded. Once the CREP message successfully arrives at the source node, data forwarding can commence. The flow chart of the CREP handle procedure in a COP topology is referred to in Figure 3.31 [4].

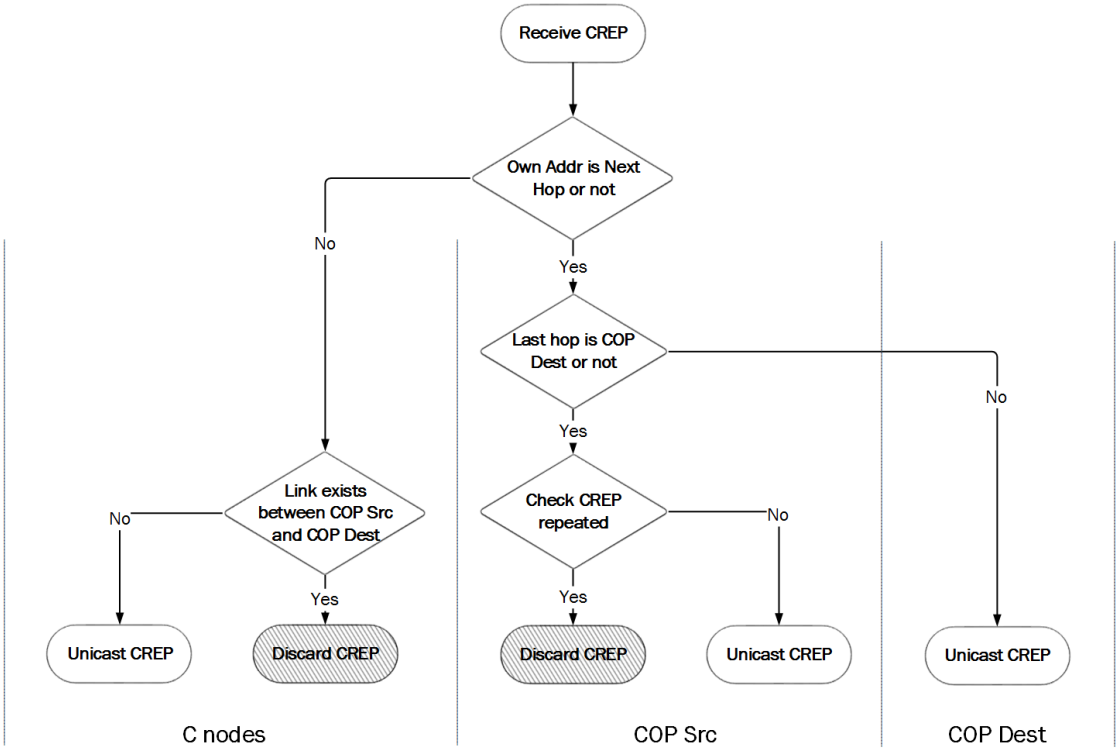


Figure 3.31: The Flow Chart of CREP Handle in the COP Topology

3.3.5 DATA Forwarding

After a CREP is unicast to the source node successfully via the above Route Reply procedure, data forwarding commences.

- Data Format

	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	*Version*				*Type*				*Reserved*								*Length*															
4	32	*Src*																															
8	64	*Dest*																															
12	96	*Last hop*																															
16	128	*Next hop*																															

Figure 3.32: CRCPR Data Message Format

[4]The shaded boxes represent CREP packet discard in the COP topology.

Last hop: 32-bit length. Before each IN forwards a DATA message, it sets its own IP address in this field. This field can only be accessed by the desired next hop and identifies which node the DATA message comes from.

Next hop: 32-bit length. This field is set with the IP address saved in the *Next hop* item of the appropriate entry in the Route Table.

- Data Handle

  For CRCPR, there are three different means of forwarding data: non-cop mode, cooperative mode and relay mode.

  In a non-COP topology, the non-cop mode for data forwarding is as follows: when a node on the selected route receives an application data message, it checks if the *Next hop* of this data is its own IP address. If not, it discards this message. If true, it passes the data to higher layer or continues to forward this data depending on whether the current node is the destination or the intermediate node on route. If the current node is an intermediate node and needs to forward data, the IP address saved in *Next hop* of the corresponding entry in the Route Table is set to the *Next hop* field of the data message. The *Last hop* field is filled with its own IP address. Finally, the data will be forwarded and the *Timestamp* of the appropriate entry in the Route Table as described in the Route Table Structure section. Every entry in the Route Table will be removed once the entry's timer expires via the *Timestamp* field. The *Timestamp* field is updated by forwarding data allowing the entry to be refreshed and so prevent deletion.

  The cooperative mode is used when data arrived at an IN in the COP topology. As mentioned in the COP Table Creation section, after the COP Table and activated C nodes are confirmed across the four INs in a COP topology, cooperative data forwarding commences, which is similar to the "Model for Cooperative Communication" in [110]. Firstly, the COP Src sends the data in a non-cop mode. Secondly, due to the feature of overhearing transmissions in wireless communication, two activated C nodes can receive the data message from the COP Src and then both C nodes will beam-form this data to the appropriate COP Dest in a cooperative mode. This provides lower power consumption via cooperative diversity. Thirdly, after the COP Dest combines the cooperative data and recovers it successfully, it continues to

forward this data. If the COP Dest plays the role of a COP Src in the next COP topology along the route, the above procedure is repeated. This cooperative data forwarding procedure differs from the "request-to-recruit" phase in CWR [131] because activated C nodes in the proactive COP Table can perform cooperative transmission without the need to recruit neighbours as the relay nodes after each data message is sent. This greatly reduces the control overhead by avoiding the complex "request-to-recruit" mechanism in CWR.

The relay mode of forwarding data is employed in a node which has a Relay Table. When data arrives at a relay node with a Relay Table, it is relayed from one Relay Neighbour to another in a relay mode if no direct connectivity exists between these two Relay Neighbours. If there is direct connectivity, the relay node simply discards this data to avoid repeated transmissions between Relay Neighbours. Both the cooperative and relay modes of forwarding data via the COP Table and Relay Table, respectively, contribute to improving robustness against mobility in CRCPR.

The data forwarding flow chart for different roles of the node in CRCPR is shown in Figure 3.33.

3.3.6 Route Enhancement

3.3.6.1 CHLO Unicast Scheme

If a valid *Next hop* field of an entry in the Route Table does not exist in the Neighbour Table, the Route Table entry will be removed. It means the more stable is the neighbour relationship of nodes on route, the more robust is the route.

In CRCPR design, the CHLO unicast scheme can be used to enhance the neighbour relationship between C nodes in the COP Table and Relay Neighbours in the Relay Table and finally increase the possibility of a valid route being identified. The reason is that both broadcast neighbours (with a value "B" in the "B/U" item in the Cooperative Neighbours Table) and unicast neighbours (with a value "U" in the "B/U" item in the Cooperative Neighbours Table) can be regarded as valid when they are used to verify the next hop validity in the Route Table.

The CHLO unicast scheme operates as follows: when a node receives a CHLO

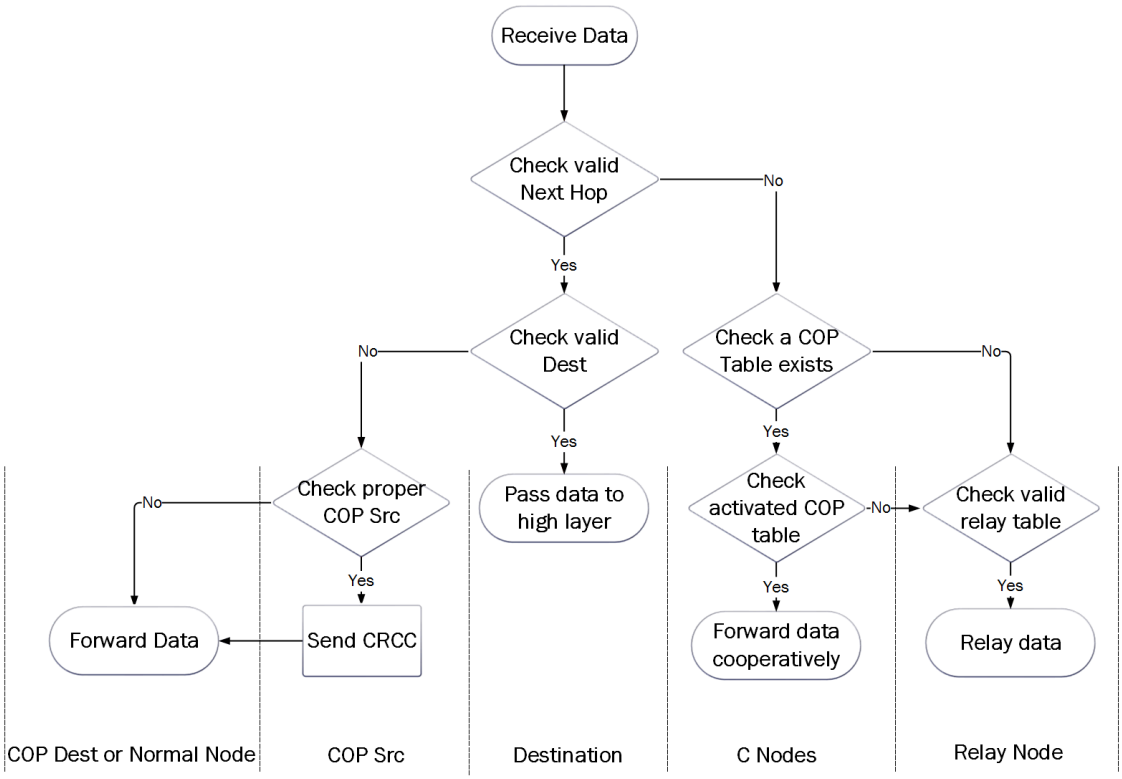


Figure 3.33: Data forwarding for different node roles in CRCPR

message from one of its C nodes in the COP Table or Relay Neighbour in the Relay Table, it unicasts this CHLO to the other C node or Relay Neighbour. More specifically, when the node is about to unicast the CHLO, the IP address of the *Dest* field in CHLO message is changed from the broadcast IP address to IP address of the other C node or Relay Neighbour. Also, *Neighbour address [n]* field needs to be removed because the unicast CHLO does not contribute to COP topology detection when performing the COP Possibility Detection Algorithm in Table 3.3.2.3. When the other C node or Relay Neighbour receives this unicast CHLO message, it sets the *Neighbour Addr* in its Cooperative Neighbour Table with the *Src* address of this CHLO message and sets "B/U" to the value "U" which means this neighbour was discovered by a unicast CHLO message (The neighbour is discovered by a broadcasting CHLO message will leads to set "B/U" to the value "B", which has been explained in Section 3.3.2.2. Both broadcast neighbours and unicast neighbours can be regarded as valid when they are used to verify the next hop validity in the Route Table). In addition, if the receiving C node or Relay Neighbour intends to broadcast its own CHLO message, it only appends the neighbour addresses, which are saved in the Cooperative Neighbour

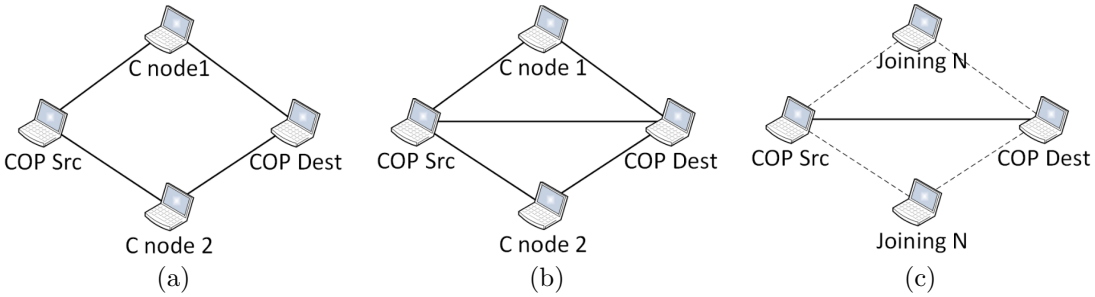


Figure 3.34: Route Enhancement Scenarios

Table with "B", into the CHLO message.

3.3.6.2 Route Enhancement Scenario

In order to utilize this enhanced neighbour relationship during data transmission to improve route robustness, the relay mode of forwarding data introduced in the Data Forwarding section is involved. Figure 3.34 provides more detail concerning route enhancement.

Scenario (a) assumes that there is no connectivity between the COP Src and COP Dest. Only when both cooperative nodes (C node 1 and 2) move out of range, will the route be broken. This is because if only one cooperative node leaves, the other cooperative node will establish a Relay Table, which allows data to be relayed from the COP Src to the COP Dest in a relay mode, maintaining the route. Assume there is connectivity between the COP Src and COP Dest as indicated as scenario (b) in Figure 3.34. Due to mobility, if the connectivity is lost, the link between the COP Src and COP Dest is also stable due to cooperative communication via the two cooperative nodes which is the same situation as scenario (a). Finally if only the COP Src and COP Dest are involved initially, and subsequently two C nodes (Joining N nodes) move into range, a COP topology is formed as shown in scenario (c). At this moment, the enhanced performance will be the same as for scenario (b). The above three cases assume there is only one COP topology between the COP Src and COP Dest. If more than one COP topology exists as mentioned in the COP Table Creation Section and two activated C nodes move out of the current COP topology range, the link between the COP Src and COP Dest is still stable. The reason is that the COP Src can manage locally to trigger another COP topology to implement cooperative data transmission to the COP Dest.

To summarize, only when all the links between the COP Src and COP Dest are

lost is the route indeed broken. Therefore, CRCPR constructs a robust, energy-efficient route by employing a COP Table, Relay Table and the CHLO unicast scheme.

3.3.7 Route Break Detection

Once established, a route in a MANET typically consists of several links between adjacent nodes and any link break on the route leads to a route break. The link connectivity between two adjacent nodes are maintained by broadcasting CHLO messages every HELLO_INTERVAL period. If one node receives a CHLO message from the other node, the link will be refreshed. However, if a CHLO message has not been received for more than ALLOWED_HELLO_LOSS * HELLO_INTERVAL milliseconds, the neighbour relationship is no longer existed and the route breaks as well. When this happens, the entry in the Route Table is deleted and the Route Reconstruction procedure commences.

3.3.8 Route Reconstruction

Once a route breaks as described in Section 3.3.7, two schemes to reconstruct the route are available: New Route Discovery and Local Repair. For these two reconstruction methods, both the Cooperative Local RePair (CLRP) message and Cooperative Error NotiFication (CENF) message are involved.

- CLRP Format:

	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	*Version*				*Type*				*Hop_No*				*TTL*				*Length*															
4	32	*Src*																															
8	64	*Dest*																															
12	96	*SEQ_No*																															
16	128	*IN Address [0]*																															
20	160	*IN Address [1]*																															
24	192	*RMV [0]*																															
28	224	*IN Address [2]*																															
32	256	*RMV [1]*																															
36	288	...																															
40	320	*IN Address [n]*																															
44	352	*RMV [n-1]*																															

Figure 3.35: CRCPR CLRP Message Format

The CLRP message is of variable length, as follows:

Src: 32-bit length. The IP address of the source node of the repairing route.

Dest: 32-bit length. The IP address of the destination node of the repairing route.

SEQ_No: 32-bit length. The time when the route broke.

TTL: 4-bit length. This field indicates the maximum hop numbers of this CLRP message can be broadcast which is calculated by the difference between *Total hop* and *Distance* of the repairing route entry in the Route Table.

- CENF Format

Octet	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	*Version*				*Type*				*ST*		*Reserved*						*Length*															
4	32	*Src*																															
8	64	*Dest*																															
12	96	*Last hop*																															
16	128	*Next hop*																															

Figure 3.36: CRCPR CENF Message Format

The CENF message is of fixed length, as follows:

Src: 32-bit length. The IP address of the source node of the repairing route.

Dest: 32-bit length. The IP address of the destination node of the repairing route.

ST: 2-bit length. ST is short for Subtype and indicates four different functions of CENF with four different ST values: 00, 01, 10, 11.

Last hop: 32-bit length. This value is set the IP address of the originating node.

Next hop: 32-bit length. This value is set based on the location of the pivoting node (the upstream node of the link break position is named as the "pivoting node" in [44]) which will be introduced later.

- New Route Discovery

In CRCPR, broadcasting CHLO messages are utilized to maintain the link connectivity between two adjacent nodes. If the link break position is located in the first half of the whole route, the immediate upstream node (pivoting node) of the broken link will unicast a CENF message with the *ST* field value "01" back to the source node to invoke a new route discovery procedure. A CENF message with the *ST* field value "10" is then unicast to the destination node by the immediate downstream node of the broken link. All the INs receiving CENF with the *ST* field value "10" and "01" will compare the *Src* field and *Dest* field in the CENF message with the *Src* field and *Dest* field of the entry saved in Route Table and finally delete the indicating entry in their Route Tables. If the link break happens between the source node and its next hop, only a CENF message with a *ST* field value "10" is unicast to the destination node and the source node invokes the new route discovery procedure directly.

- Local Repair

  If the link break position is located in the second half of the whole route, the pivoting node invokes the local repair process by broadcasting a CLRP message. The local repair follows this procedure: The pivoting node sets the *Next Validity* item of the appropriate entry in the Route Table as "invalid". As the immediate downstream node cannot receive any data from the pivoting node (the last hop node) due to the link break, it deletes this entry in its Route Table directly. Next, the pivoting node unicasts a CENF message with the *ST* field value "11" back to the source node to inform all upstream nodes to remain silent, which means they will suspend forwarding data messages. Also, the immediate downstream node unicasts a CENF message with the *ST* field value "10" to the destination node. All the downstream INs delete the indicating entry from their Route Tables. Then, the pivoting node sets a CLRP_WAIT timer and broadcasts a CLRP message to invoke the local repair process. The *Src* and *Dest* fields of the CLRP message are set the IP addresses of the*Src* and *Dest* items in the repairing route respectively. The *TTL* field is set as the difference between *Total hop* and *Distance* of the repairing route entry. Each time the CLRP message is processed, the value of *TTL* field is decremented by 1. When the *TTL* reaches 0, this CLRP message will be discarded in order to ensure that the newly repaired route (if

repaired successfully) with the number of total hops is less than the broken route. If the CLRP message arrives at the destination node successfully, a CREP message will be generated by the destination node and unicast back to the source node. All the downstream INs receiving this CREP establish a new route entry in the Route Table and the pivoting node changes the *Next Validity* item to "valid" and sets the *Next hop* item with the newly repaired IP address. All the upstream INs including the source node cancel their silent state and update the information in the appropriate route entry. Finally, the route is repaired locally and ready to forward data. If no CREP message is unicast to the pivoting node within the CLRP_WAIT time, the pivoting node will delete the repairing route entry and send a CENF message with a ST field value "00" to backtrack one hop (a new pivoting node) to trigger another local repair procedure. This "backtrack" case repeats until the position of the new pivoting node is located in the first half of the repairing route. At this moment, a CENF message with a ST field value "01" is unicast back to the source node by the pivoting node to invoke the new route discovery procedure.

3.3.9 Route Selection Criteria

In CRCPR, in order to obtain the final route with highest Route Performance Coefficient, Q, we employ the parameters given in Table 3.1:

Table 3.1: Parameter Notation in Routing Selection Criteria

H_i	Energy Harvest Degree
C_e	Energy Conversion Efficiency
K_i	Energy Harvest Contribution
C_a	Battery Capacity Degree
R_i	Energy Accumulation Rate
E_i	Real-time Residual Energy Degree
a_i	Energy Drain Rate Coefficient
L_i	Link Break Degree
p_i	Link Break Probability
Q	Route Performance Coefficient

3.3.9.1 Energy Harvest Degree

In CRCPR, we assume that some nodes have an energy harvesting ability. In order to consider these energy-harvesting nodes, the Energy Harvest Degree is proposed to contribute to the selection of the final route to improve network lifetime.

$$H_i = C_e(-e^{\frac{-K_i}{n}} + 1) \tag{3.1}$$

From Equation (3.1), the Energy Harvest Degree of node i with constant value n is relates to Energy Conversion Efficiency C_e and Energy Harvest Contribution K_i, where K_i is determined by the Battery Capacity Degree C_{a_i} and Energy Accumulation Rate R_i in Equation (3.2).

$$K_i = \frac{R_i}{C_{a_i}} \tag{3.2}$$

Generally, the Battery Capacity Degree C_{a_i} is fixed for each device but the Energy Accumulation Rate R_i is different according to the energy source like radio waves [161] or solar [162]. In this paper, we choose the more mature energy harvesting technology, that of solar energy; more details are given in [162].

3.3.9.2 Real-time Residual Energy Degree

The Real-time Residual Energy Degree is proposed to represent the residual energy of a node in real-time. This is used to calculate the Energy Drain Rate Coefficient in Section 3.3.9.3.

$$E_i = (-e^{-E_i^r(1+H_i)/n_1} + 1)^{n_2} \tag{3.3}$$

E_i^r is the real-time residual energy of node i and $E_i^r \geq 0$. H_i is the Energy Harvest Degree. n_1 and n_2 are constant values. Equation (3.3) guarantees E_i is within the range [0, 1].

3.3.9.3 Energy Drain Rate Coefficient

Energy Drain Rate can be used to reflect the energy consumption rate of one node. Even one node may have high residual energy, its lifetime may not be long if it

also has a high energy consumption rate. Therefore, Real-time Residual Energy Degree can avoid a node with high energy drain rate being selected in the route.

$$a_i = \begin{cases} 1 & \frac{R_i}{E_i} < R_{thr} \\ 0.1 & \frac{R_i}{E_i} \geq R_{thr} \end{cases} \tag{3.4}$$

$$R_i = \frac{1}{N-1} \sum_{k=i-N+1}^{l} R_k(t) \tag{3.5}$$

Energy Drain Rate Coefficient to describe this property in the Equation (3.4), where E_i is the Real-time Residual Energy Degree and E_i is the energy drain rate defined in [163] as given in Equation (3.5). R_{thr} is a scenario-selectable parameter. The value of the Energy Drain Rate Coefficient a_i can be used in the final route selection scheme to exclude nodes with high energy drain rates.

3.3.9.4 Link Break Degree

The Link Break Degree L_i is used to indicate the stability of each link on the route which can be calculated based on Equation (3.6)

$$L_i = \frac{1}{(1 + e^{-10(p_i - p_0)})} \tag{3.6}$$

where p_i is the Link Break Probability in CRCPR which will be introduced in Section 3.3.9.5 and p_0 is a scenario-selectable parameter [5] and makes our final selected route more stable. More specifically, one pivotal target of CRCPR is to enhance resilience in the network as described in Section 3.3.6. After deploying this scheme into the network, if the link break probabilities (p_i) of some specific nodes are still higher than a scenario-selectable threshold p_0, we regard these links as "extremely unstable" links. In contrast, links with lower link break probabilities (p_i) than p_0 are regarded as "extremely stable" links. Compared with a calculated p_i according to Section 3.3.9.5, for "extremely unstable" links, it could be reset a larger link break probability value to ensure they are less like to be involved in

[5] Currently, p_0 is a value based on experience which can effectively reduce the link breaks. In the future, it could be set automatically according to different mobility models.

the final route while for "extremely stable" route, it could be reset a lower link break probability value to ensure they have a greater chance of being selected in the final route. If only p_i is utilized to indicate the link break stability, it cannot implement the above concern in the route selection criteria. The reason is that once p_i is calculated, it cannot be changed to a larger or lower link break probability value to indicate a "extremely unstable" links or "extremely stable" link, respectively. Therefore, Link Break Degree L_i obtained by Equation (3.6) is designed to solve this issue.

Figure 3.37 shows the the relationship between Link Break Degree L_i and Link Break Probability p_i of Equation (3.6): when p_i is larger than p_0, every corresponding value of L_i is larger than p_i; when p_i is smaller than p_0, every corresponding value of L_i is smaller than p_i. Therefore, the link with a p_i which is higher than p_0 will be regarded as the "extremely unstable" link and it will be reset a higher link break probability value L_i. In contrast, the link with a p_i which is lower than p_0 will be regarded as the "extremely stable" link and it will be reset a lower link break probability value L_i. Link Break Degree apparently ensure "extremely unstable" links are less likely to be involved in the final route and "extremely stable" links have a greater chance of being selected in the final route, which finally enhances the resilience of the network.

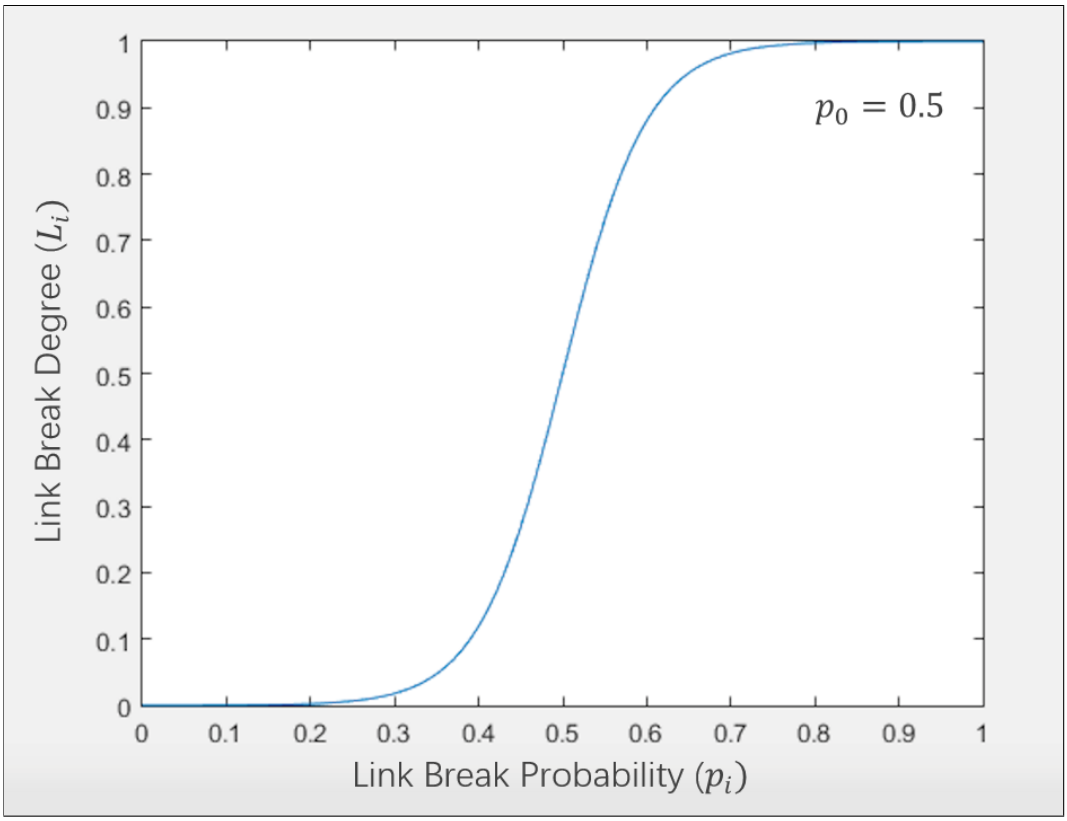
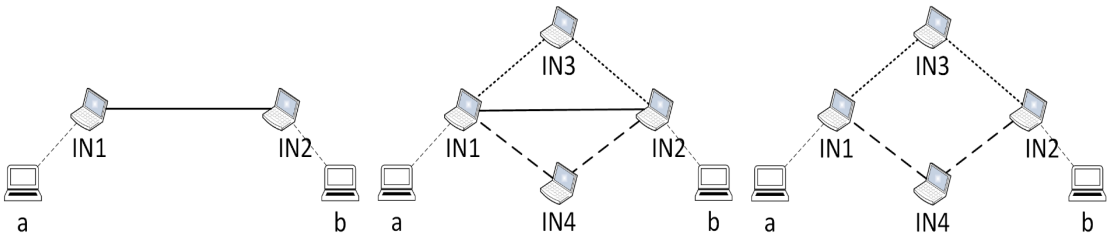


Figure 3.37: Link Break Degree

3.3.9.5 Link Break Probability p_i in CRCPR

In an Ad hoc network, any link between two nodes can be regarded as an independent event ε with outcome $\{B, \overline{B}\}$, where B denotes that the link breaks within a short time interval Δt and $\overline{B}$ denotes the link keep stable within a short time interval Δt. Assume that the Link Break Probability is $P(B) = p$. As the link only has two states within any Δt, that is, break or non-break, so the link non-break probability is $P(B) = 1 - p$. Once one link along the route breaks, the route breaks. Therefore, we can calculate the route break probability according to Link Break Probability. For CRCPR, p_i consists of three different link break probabilities according to three types of links: a non-cop link p_n (two nodes can communicate with each other directly), a connected COP link p_c (i.e. a link exists between IN1 and IN2) and an unconnected COP link p_{nc} (i.e. a link does not exist between IN1 and IN2) as shown in Figure 3.38 from (a) to (c).

As details for calculating the Link Break Probability of a non-cop link like (a)



(a) (b) (c)

Figure 3.38: COP Topology Links

in Figure 3.38 have been presented in [164], we only refer to the conclusion

$$p_n = p \tag{3.7}$$

For a connected COP link like (b), we have the dotted link L_{dotted}, solid link L_{solid} and dashed link L_{dashed}. For an unconnected COP link like (c), we have the dotted link L_{dotted} and dashed link L_{dashed}. The Link Break Probability of the dotted links, solid links and dashed links are obtained by the property of mutually exclusive events.

$$P(L_{dotted}) = 1 - (1 - p_n)^2 = 2p - p^2 \tag{3.8}$$

$$P(L_{solid}) = p_n = p \tag{3.9}$$

$$P(L_{dashed}) = 1 - (1 - p_n)^2 = 2p - p^2 \tag{3.10}$$

We can then obtain the link break probabilities for the connected and unconnected COP link cases from Equation (3.11) and (3.12), respectively.

$$p_c = P(L_{dotted}) \times P(L_{solid}) \times P(L_{dashed}) = p^5 - 4p^4 + 4p^3 \tag{3.11}$$

$$p_{nc} = P(L_{dotted}) \times P(L_{dashed}) = p^4 - 4p^3 + 4p^2 \tag{3.12}$$

After we obtain these three link break probabilities in CRCPR, the Link Break Degree L_i can be calculated based on Equation (3.6).

3.3.9.6 Route Selection Strategy

As CRCPR is a cross-layer scheme, it can utilize RMV information passed to the Network Layer from the Physical Layer and MAC Layer, such as the Battery Capacity Degree, the Energy Accumulation Rate, the Real-time Residual Energy Degree, the Link Break Probability and so on. All these data can be carried by CREQ messages to the destination and contribute to the route selection. Therefore, the final route selection strategy can be modelled by Equation 3.13:

$$Q = \min_{1 \le i \le n} [\frac{E_i}{e^{\frac{routeEntry}{2}}}] \times \prod_{i=1}^{n} (e^{a_i + 1 - L_i}) \tag{3.13}$$

where i indicates the hop numbers in the total route, *routeEntry* indicates the amount of Route Table entries at node i, E_i is Real-time Residual Energy Degree, L_i is Link Break Degree Energy and a_i is Drain Rate Coefficient. $E_i \Big/ e^{\frac{routeEntry}{2}}$ is designed to avoid choosing the node in the final route with much heavy payload which is heavily utilized with fast energy consumption or has a considerable amount of data awaiting transmission.

Assuming ε CREQ messages are received during CREQ_WAIT period, the sequence of CREQ messages can be denoted by $\{C_{pkt_1}, C_{pkt_2}, ..., C_{pkt_\varepsilon}\}$. $Q(C_{pkt_\varepsilon})$ represents each Route Selection Strategy value for each C_{pkt_ε}. Finally, a route with the maxmum value of $Q(C_{pkt_\varepsilon})$ is chosen and the corresponding $Q(C_{pkt_\varepsilon})$ will be inserted into CREP before unicasting back to source node.

$$Q_{final} = \max[Q(C_{pkt_1}), Q(C_{pkt_1}), ..., Q(C_{pkt_\varepsilon})] \tag{3.14}$$

3.4 Summary

This chapter introduced a novel routing protocol called "Constructive Relay based CooPerative Routing" based on cooperative communication to support emerging environments like IoT. By exploiting cooperative diversity with the help of a COP Table data structure, energy consumption during transmissions can be significantly reduced. Additionally, by employing a relay principle based on a Relay Table, CRCPR provides greater robustness against node mobility induced link breaks. A new route selection scheme that can utilize the RMV from the Phys-

ical/MAC Layer such as Battery Capacity Degree, Energy Accumulation Rate, Real-time Residual Energy Degree, Link Break Probability and so on, determines the final route. The overall network performance is improved significantly. Our Network Layer framework explicitly covers cooperative route discovery, route confirmation and route enhancement. This framework can be readily integrated with existing lower layer mechanisms to improve the performance of MANETs. In Chapter 4, details about how to implement CRCPR design in OPNET platform are provided.

Chapter 4

Implementation and Validation

4.1 Simulation Platform

4.1.1 OPNET Overview

Simulation plays an important role in network research. OPNET, as a system level event based network simulation tool, is generally used by researches, protocol designers, universities in the field of electronic engineering and computer science [165].

4.2 System Model

As shown in Figure 4.1, CRCPR was implemented as a sublayer within the Network Layer. The reason is that the sublayer design will not effect the original architecture and functions of the Open Systems Interconnection (OSI) model [166]. If the function of CRCPR is required, it can be activated to support multi-hop ad hoc mobile communication. Otherwise, it can still support regular IP traffic over other wired or one-hop wireless networks.

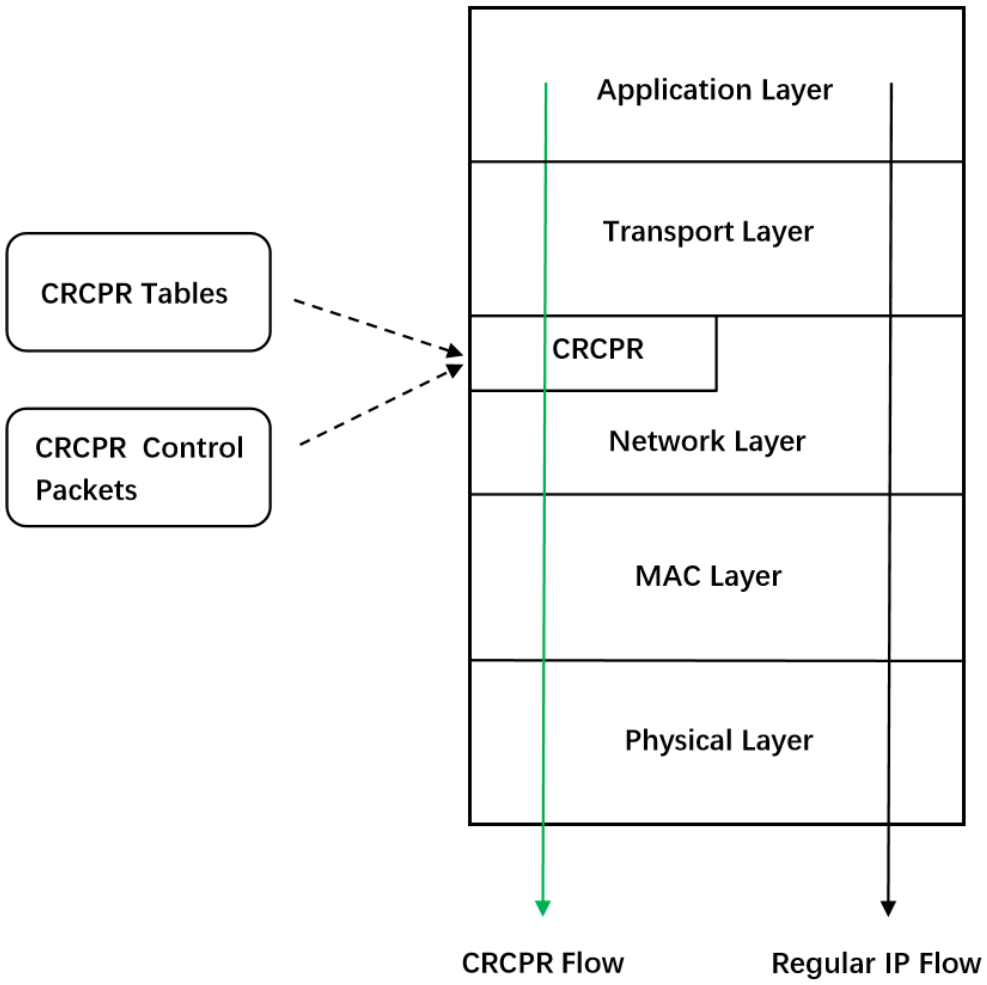


Figure 4.1: CRCPR Implementation Architecture

Figure 4.2 illustrates the CRCPR packets forwarding process in MANETs with two hops. Data packets are generated by the source node in its Application Layer and sent out. At intermediate nodes (INs), the packet is brought to the CRCPR sublayer to be processed and then is re-sent. All control packets like CREQ, CREP, CENF and so on are generated and processed at the CRCPR sublayer. If the data packet arrives at the IN, it will be processed in CRCPR sublayer and then re-sent [1]; if it reaches the destination node, the valid data packet will be passed to the higher layers.

[1]The data in CRCPR needs to be transmitted in cooperative mode via Cooperative Nodes in the COP Topology or in the relay mode via the relay nodes with the Relay Table. The MAC Layer normally drops the frame unless the frame is addressed with the Next Hop's MAC address or broadcast address. Therefore, promiscuous Mode [167] is employed in CRCPR, which allows the MAC Layer to pass all frames through to the Network Layer for processing.

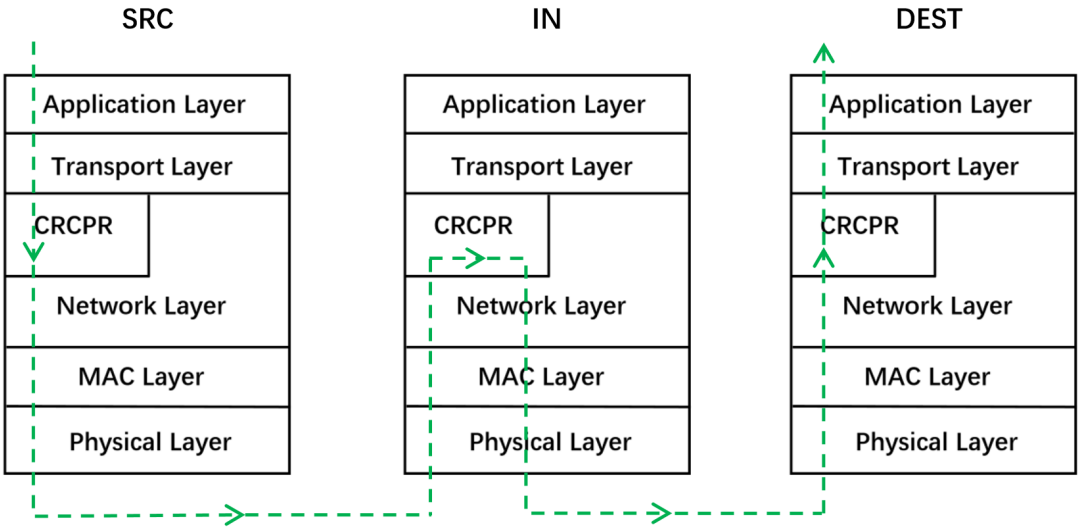


Figure 4.2: Packet Forwarding in the CRCPR Protocol

In order to implement the above architecture, we utilize the hierarchical structure of OPNET, where the modeling method of a network system is divided into three parts: Network Model, Node Model and Process Model. Figure 4.3 demonstrates the structure among these models.

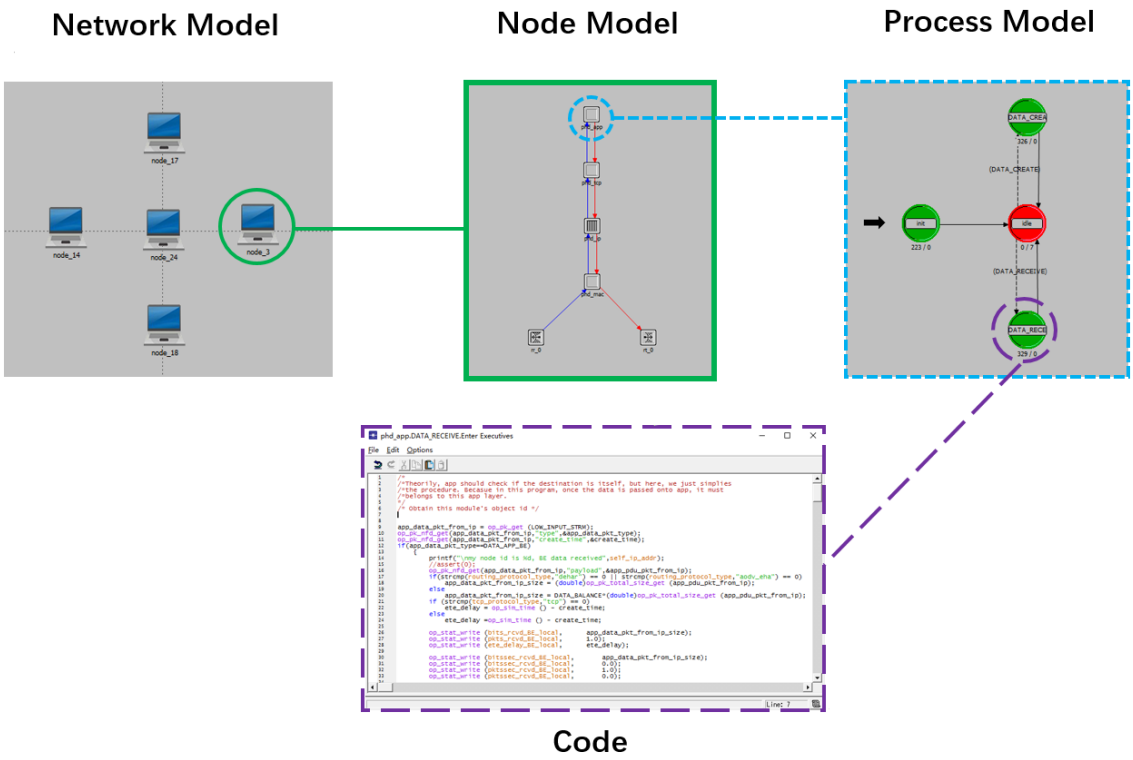


Figure 4.3: Three-tiered Hierarchy in OPNET

4.2.1 Network Model

The Network Model is used to specifies the subsets, nodes or links in the simulation scenario. Subsets can be used to divide the whole network into many sub networks for different functions. Nodes represent network devices such as servers, workstations, routers and so on. Links consist of wired connections like cables and wireless connections like cellular channels, WLAN and satellite pathways.

4.2.1.1 Protocol Configuration

All the property values of each protocol in the Application Layer, Transport Layer (TCP Layer) and Network Layer should be updated or controlled in the Process Model. However, for the CRCPR implementation in OPNET, all the protocol properties have been promoted up to the Network Model as shown in Figure 4.4. There are two reasons for this promotion: 1) It is convenient to set all the property values in the Network Model. 2) All the nodes can be configured individually without influencing other nodes. For example, any node can be set as a source node by setting the *send_data* property as "enabled" as shown in Figure 4.4. 3) If one protocol has been confirmed in one layer, its specific properties can be configured by the Collapse Row such as AODV in Figure 4.5.

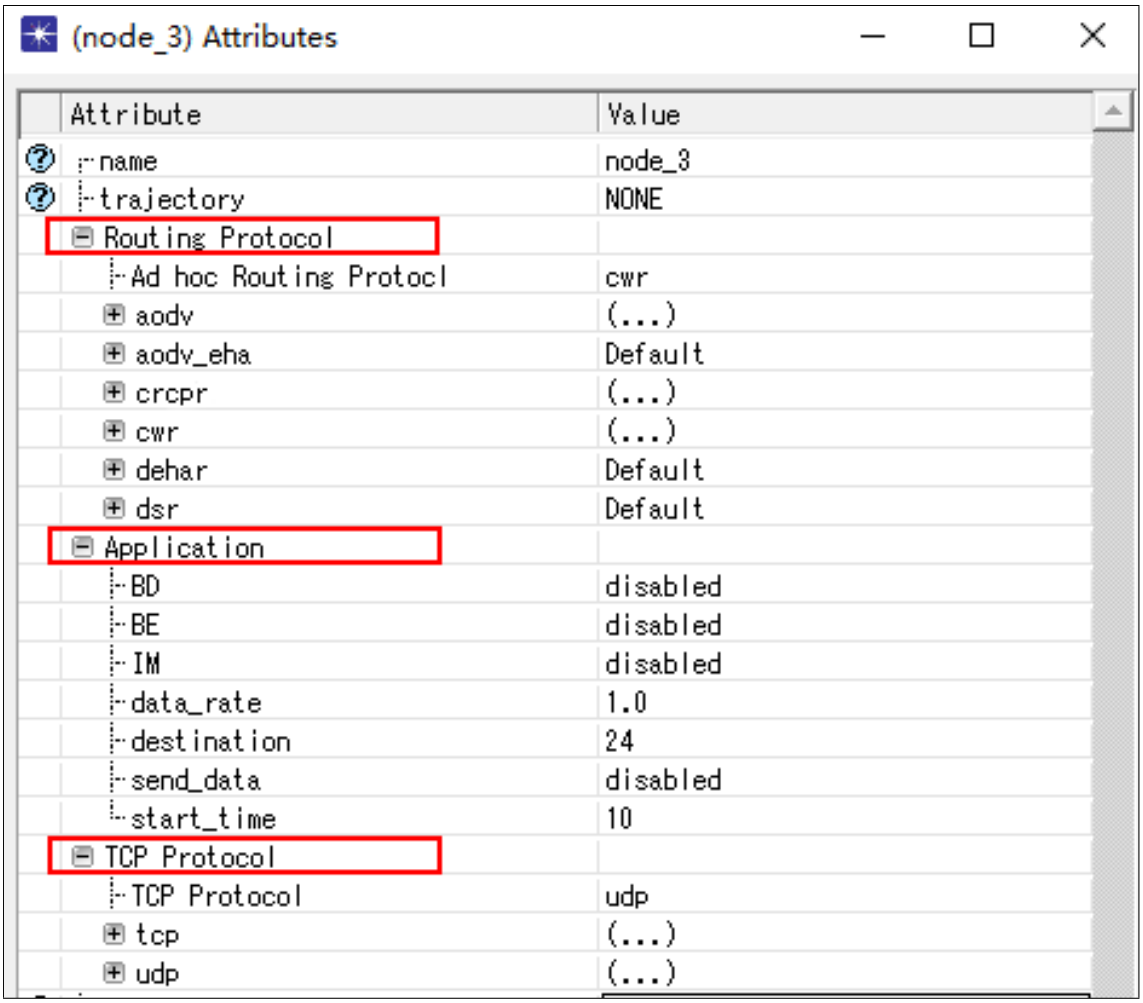


Figure 4.4: Protocol Configuration

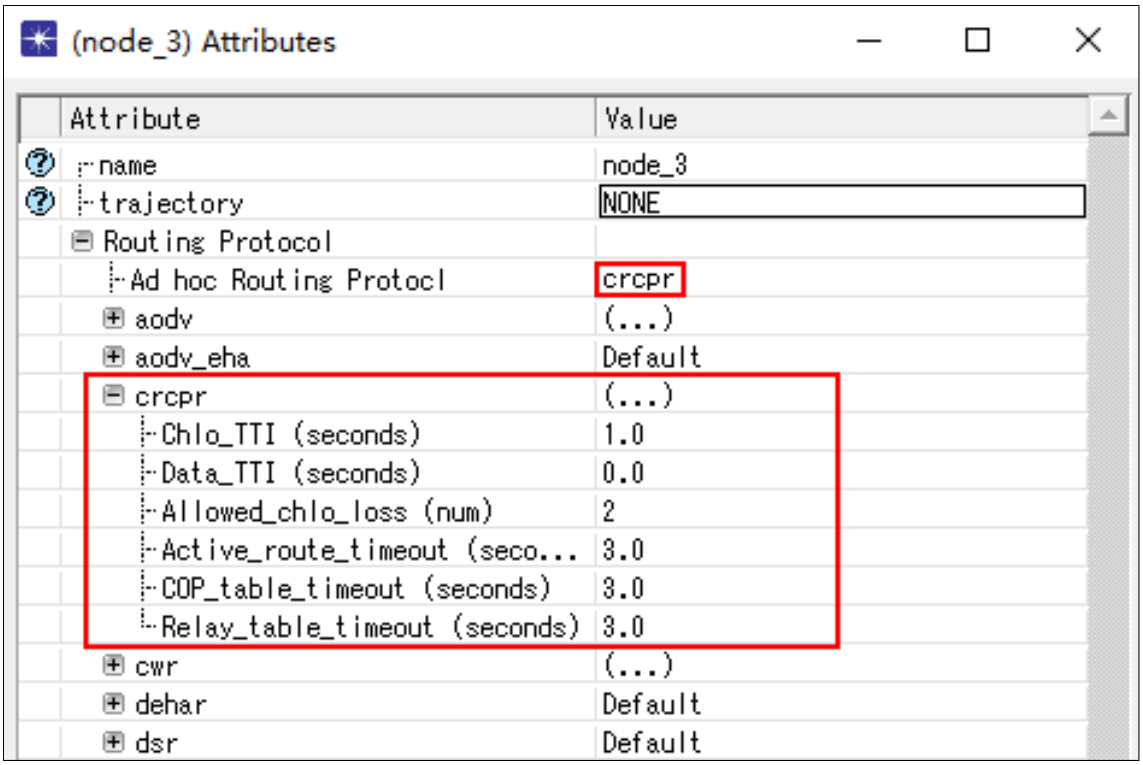


Figure 4.5: CRCPR Configuration

4.2.1.2 Mobility Configuration

As all the nodes in MANETs can move randomly, the node devices in the Network Layer are set as mobile type with two kinds of mobility manners: the *Trajectory* and *Mobility Profile*. Both mobility manners have been shown in Figure 4.6.

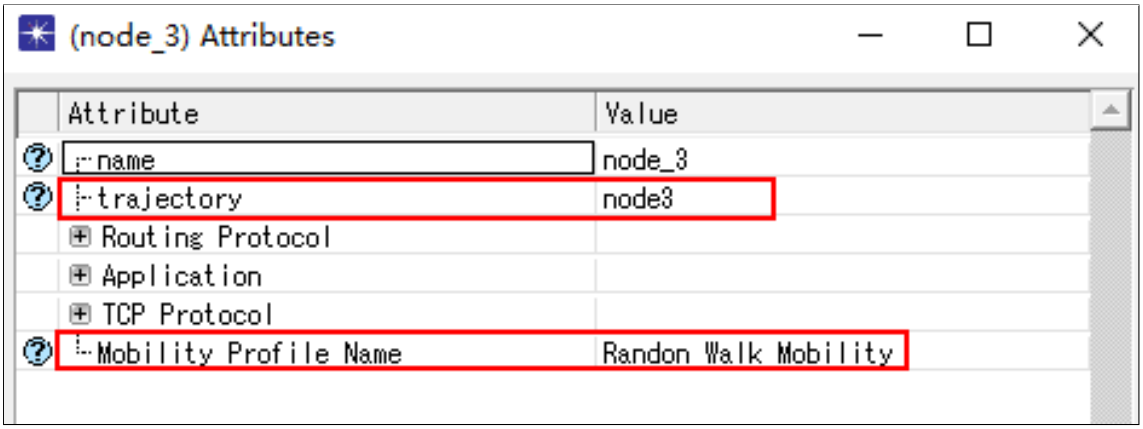


Figure 4.6: Mobility Configuration

For the *Trajectory* manner, all the movement features have to be set manually before the simulation in the Trajectory Panel in Figure 4.7. For example, if we

have set the trajectory for *node_2*, *node_2* can only move from the starting point to the ending point with a pre-defined constant velocity. After the waiting time, it will move along the next trajectory (if appropriate). The advantage of a trajectory is that it is easy to configure quickly. But the disadvantage is that it is difficult to configure complex mobility patterns like random movements because the pre-defined manual configuration is time-consuming and cannot employs the random seeds [2] for the random movements.

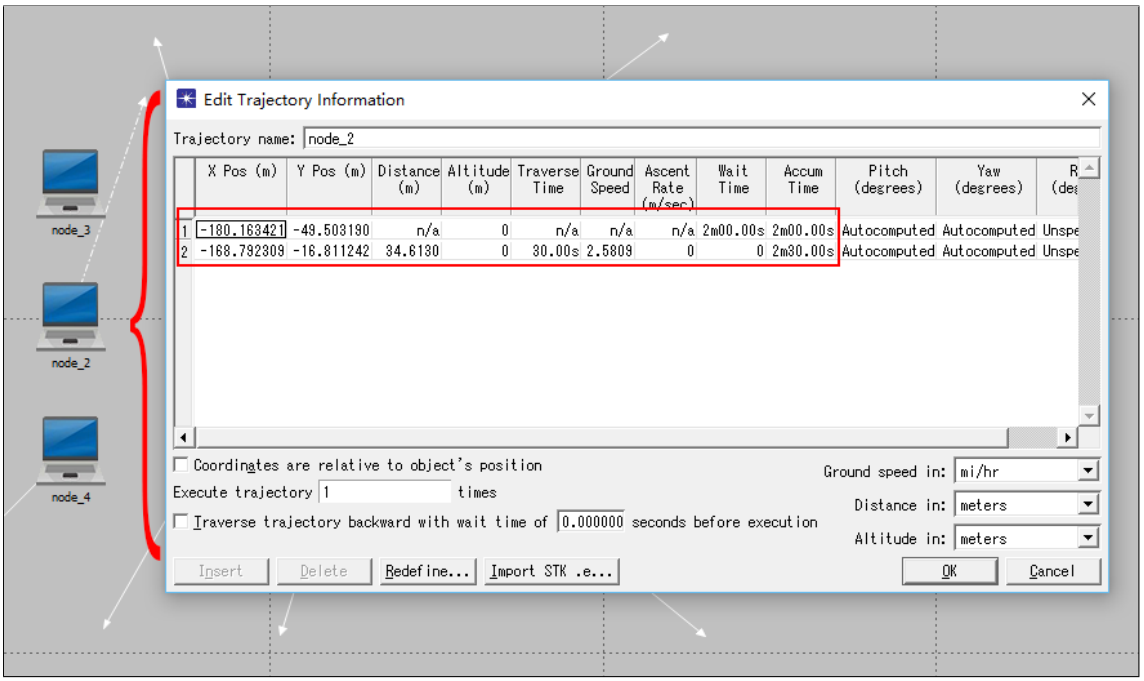


Figure 4.7: Trajectory Configuration

For the *Mobility Profile* manner, it is used to solve the complex mobility configuration issue. The detailed procedure is as follows: Firstly, a Mobility Domain needs to be selected in the Object Palette Tree as shown in Figure 4.8. The Mobility Domain is a restricted area that the configured node moves inside. Secondly, we use the Mobility Domain to select a rectangular area around a node like *Node_3* in Figure 4.9. Thirdly, the Mobility Config Module in OPNET configures the corresponding Mobility Domain properties according to the domain name as shown in Figure 4.10.

During a simulation, we use Mobility Profile to set the Random Walk Mobility for the nodes in the scenario, which is a typical synthetic mobility models in the

[2]Random seeds are set based on the in-built Random Number Generation (GNU) in the OpNET

simulation environment as introduced in Section 2.2.0.4.

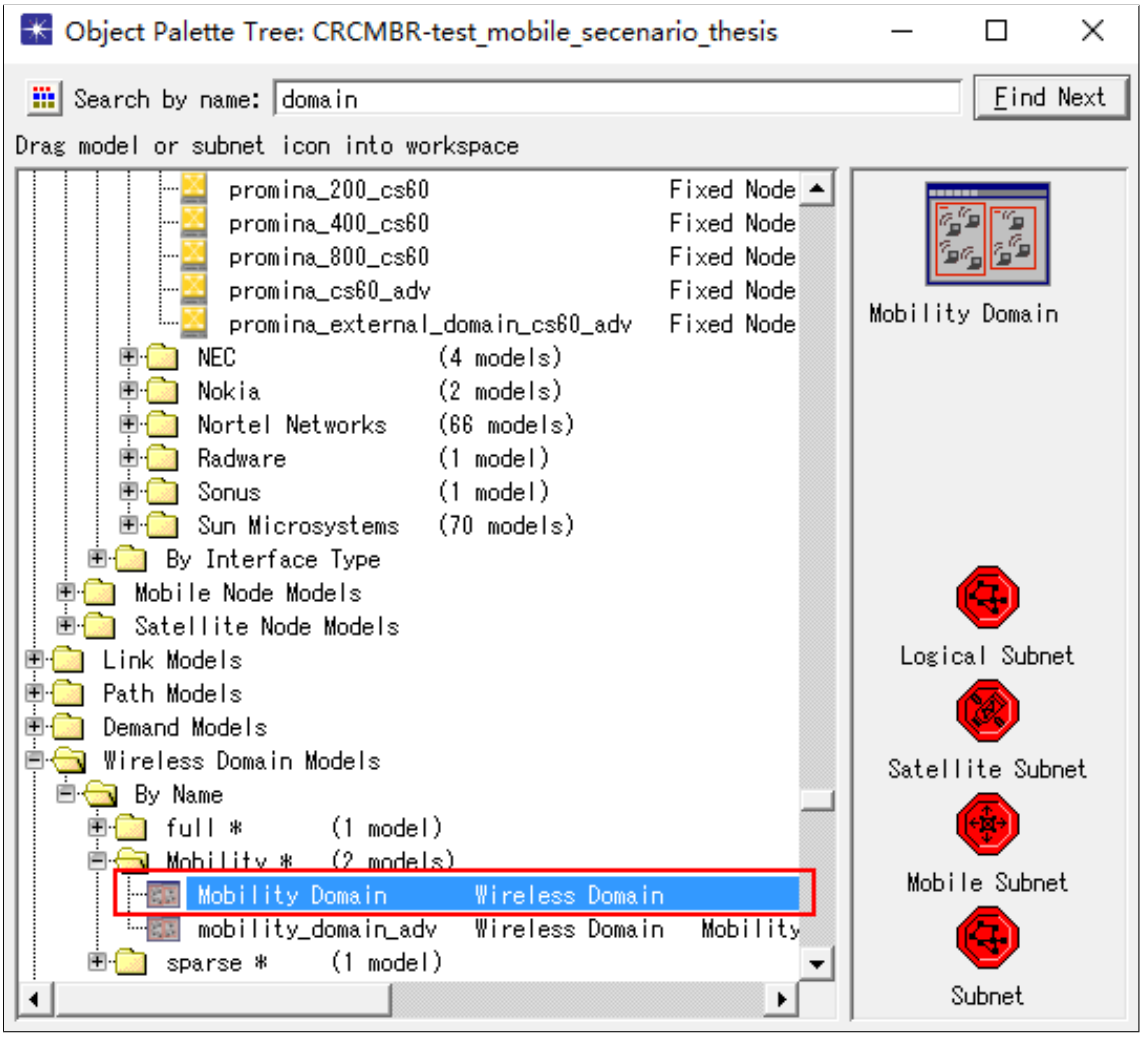


Figure 4.8: Mobility Domain Selection

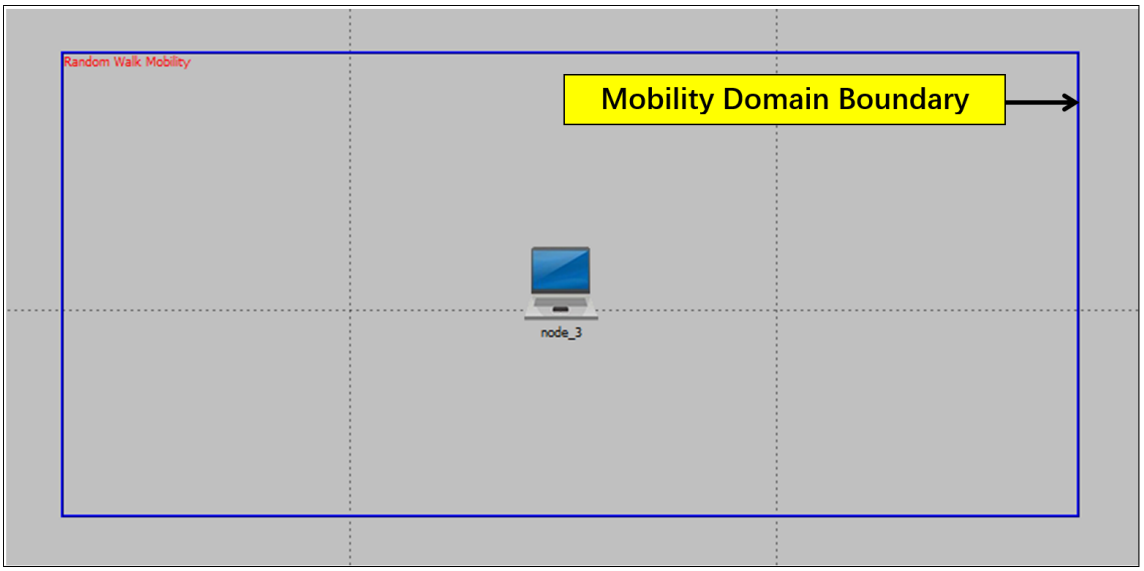


Figure 4.9: Mobility Domain Scenario Setting

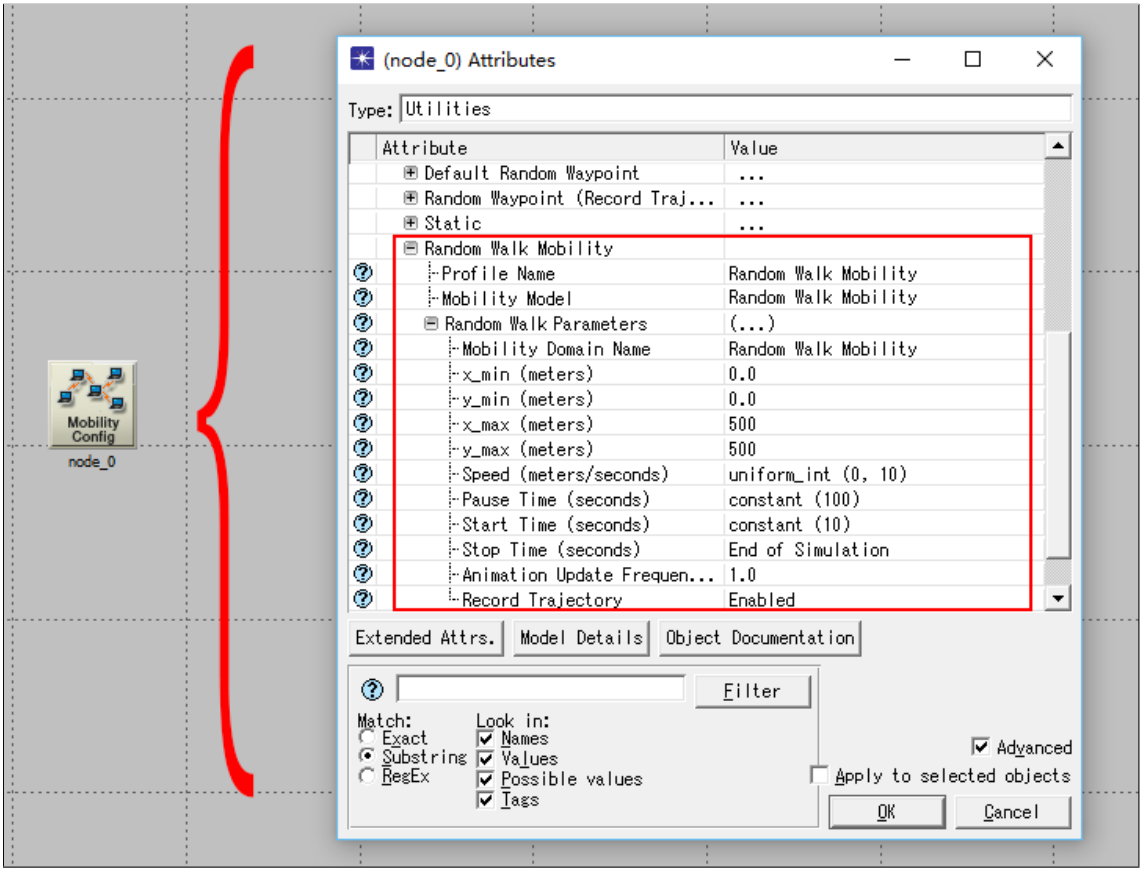


Figure 4.10: Mobility Domain Configuration

4.2.2 Node Model

The Node Model specifies the internal structure of a node in the Network Model. The Node Model consists of several blocks such as the Processor, Queue and Transceivers. The interfaces among these blocks are called the Packet Streams. All these entities are illustrated in Figure 4.11.

4.2.2.1 Node Editor

The Node Editor is used to edit the Node Model as shown in Figure 4.11. In the CRCPR Node Model, the complete OSI Models are implemented by the Processors and Queue: Application Layer, Transport Layer, Network Layer, MAC Layer and Physical Layer.

The Processor is fully programmable via the Process Model, which is introduced in Section 4.2.3. The Queue is almost the same as the Process. The only difference is that the Queue can automatically buffer and manage data packets. The Packet Stream takes charge of the connections among Processors, Queues and Transceivers. In the CRCPR implementation, if a node in the Network Model receives a packet, this incoming packet is firstly captured by the Receiver then it is passed upto the higher layers along the blue arrow if necessary as shown in Figure 4.11. The red arrow illustrates the direction of the outgoing packet. The transceivers are wireless interfaces for the node in the Network Model.

As CRCPR is a Network Layer scheme, we assume that the MAC Layer and Physical Layer with OFDMA (Orthogonal Frequency Division Multiple Access) system [168] deployment can fully support symmetric wireless channels without any interference during the transmission.

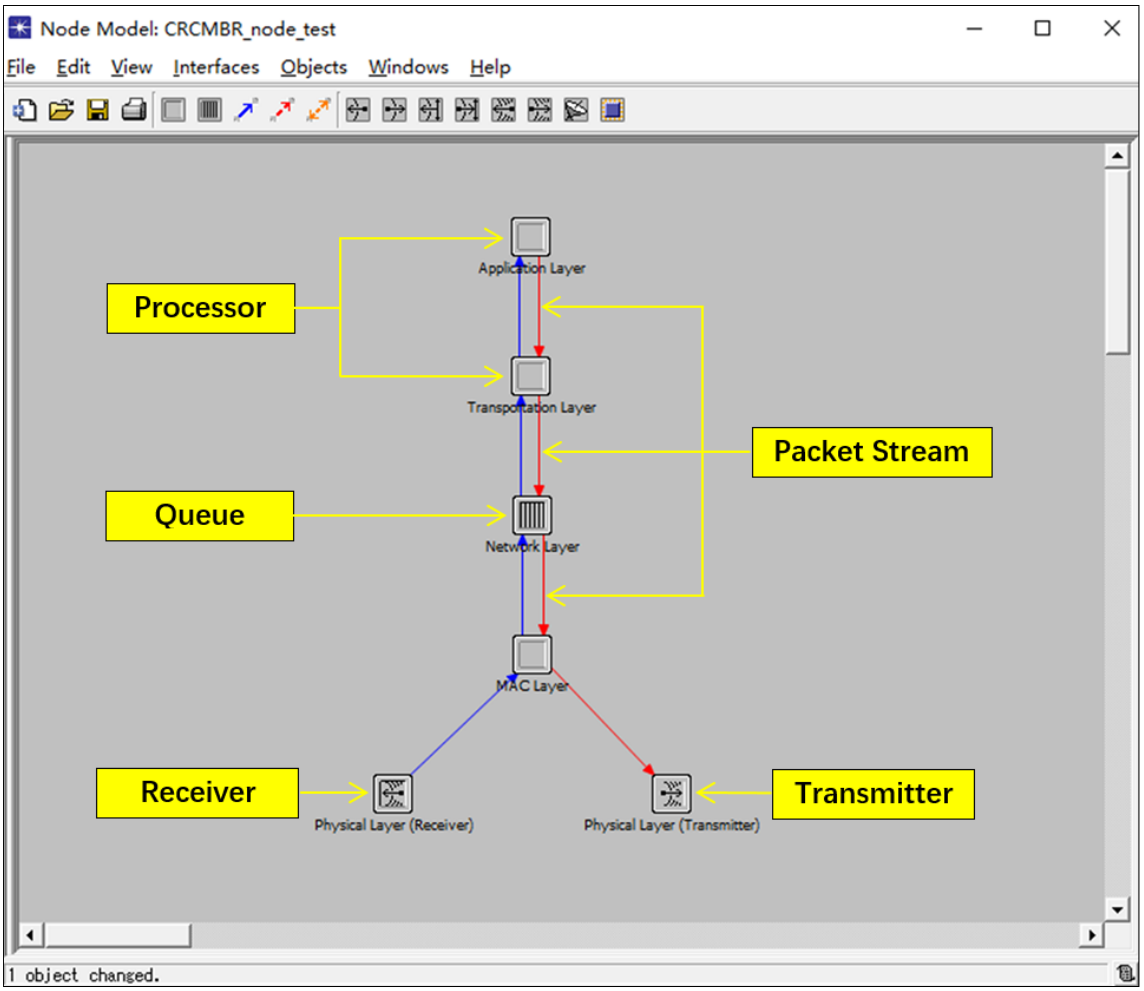


Figure 4.11: Node Model in CRCPR

4.2.2.2 Packet

The Packet can be edited in the Packet Editor, which is a information-carrying entity passed along the Packet Stream inside the Node Model as shown in Figure 4.12. Outside the Node Model in CRCPR, the sending packets are received by the other appropriate nodes via a wireless channel that provides data transmission among different nodes.

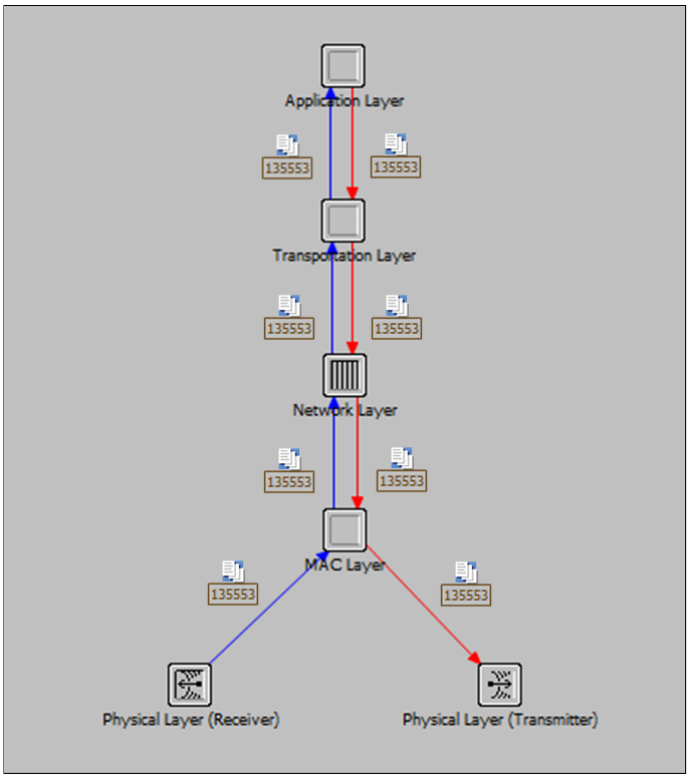


Figure 4.12: Packet Transmission inside the Node Model

Packet formats define the structure of packets with a set of fields as shown in Figure 4.13. CRCPR relies on multiple types of packets with different packet formats such as CREQ, CREP, CENF and so on. All these packets can be dynamically created and destroyed by the Process Model during the simulation process and details are provided in Section 4.2.3.

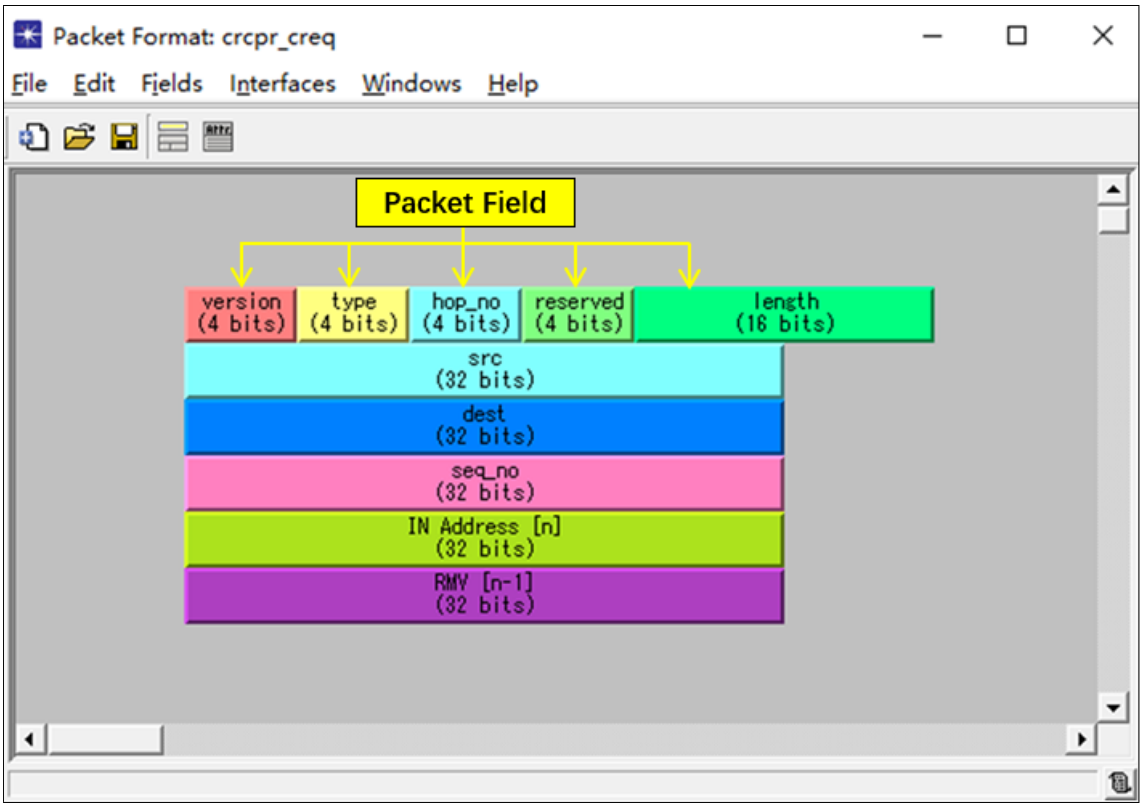


Figure 4.13: CREQ Packet Format in the Packt Editor

Figure 4.13 shows the CREQ packet format of CRCPR in the Packet Editor. Other types of packet formats are set in the Packet Editor according to the packet design introduced in Section 3.

4.2.3 Process Model

The Process Model is a programmable block used to define the functionality of each Processor or Queue in the Node Model. During the data transmission, the Process Model in the Network Layer Processor has two functions: (1) For the source node and immediate nodes on the route, it decides the valid data to be passed down to the lower MAC Layer Processor and then the Transmitter Processor. After that, the data is sent out to the Receiver Processor of the next hop node. (2) For the destination node, it decides the valid data to be passed up to the Transport Layer Processor and then the Application Layer Processor.

4.2.3.1 Process Editor

According to Figure 4.14, the Process Editor is used to create the Process Model, which consists of state transition diagram/Finite State Machine (FSM) and some variables and blocks such as State Variables (SV), Temporary Variables (TV), Header Block (HB) and Function Block (FB). In the FSM, there are two kinds of states: forced states (green state) and unforced states (red state). In a forced state, after the executive code in this state is finished, the current state will be transferred to the next state along the valid transition condition without any blocking or waiting. In an unforced state, after the executive code in this state is processed, there is no state transfer unless an event triggers the state transition such as the interruption. There are two common interruptions: stream interruption and self interruption. A stream interruption can be created by packet reception and a self interruption can be created by the OPNET kernel to implement some function periodically such as hello message broadcasting, data generation at a source node.

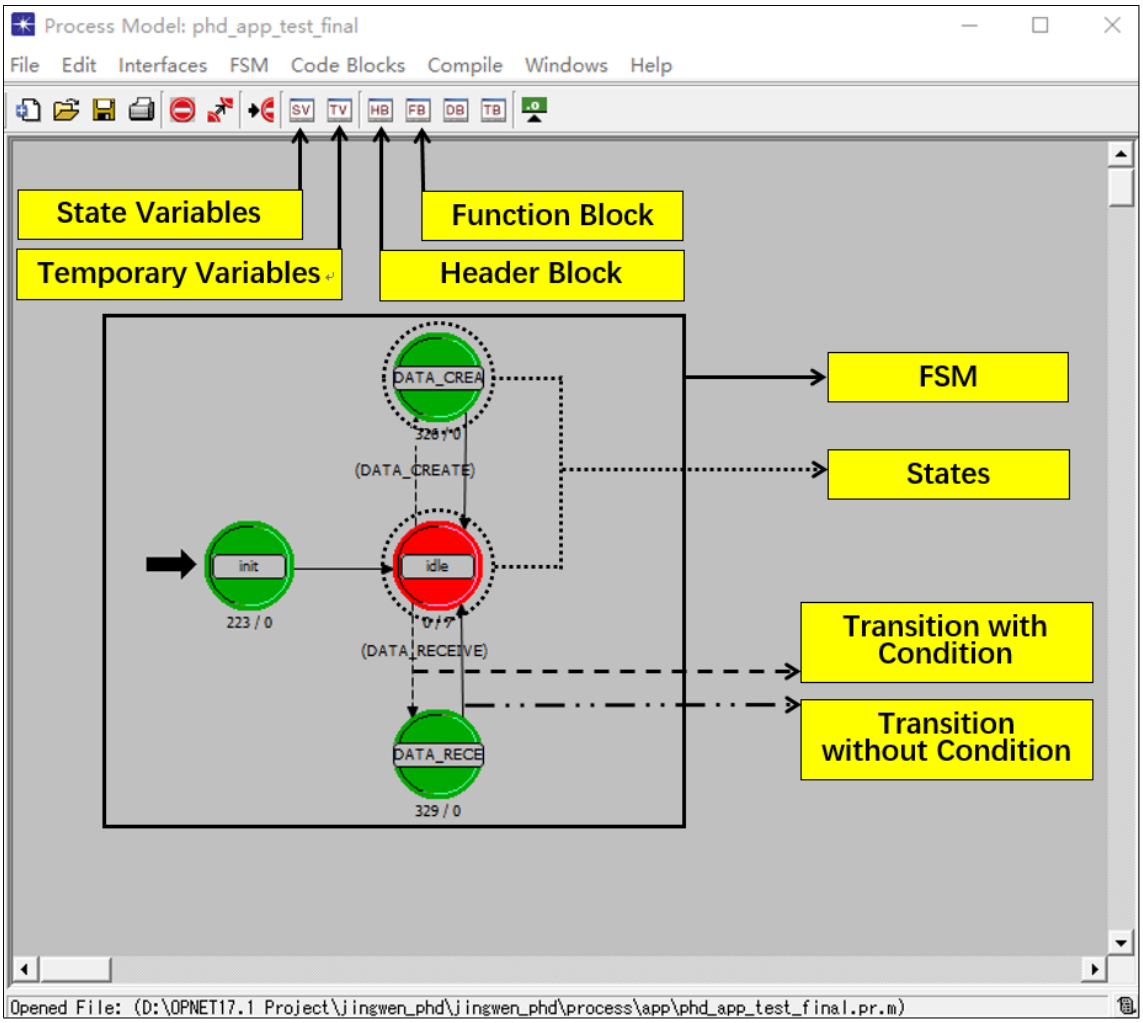


Figure 4.14: The Process Editor

- Process Model of the Application Layer

Figure 4.15 is the Process Model of the Application Layer Processor. There are four states for this process model: the "Initial State", "Data Create State", "Data Receive State" (all these three states are forced states) and the "Idle State" (an unforced state).

The green state with a black arrow is called the Initial State. This state is triggered to execute at the beginning of the simulation. It takes charge of the initial configuration of the Application Layer such as registration of the statistics and data generation initialization. After the execution of the Initial State, control is transferred to the Idle State to wait for the next event. If a data creation event happens via a self interrupt, the flow of control moves from the Idle State to the Data Create State, which is responsible for generating data according to the data

rate set in the Network Model. After the Data Create State is finished, the flow of control will be transferred to the Idle State again to wait for the next event. If a data reception event happens via a stream interrupt, flow moves from the Idle State to the Data Receive State which is in charge of data reception and collecting statistics such as throughput.

Data creation and data reception are implemented according to the above state transition procedure and this procedure is executed all the time until the simulation is terminated.

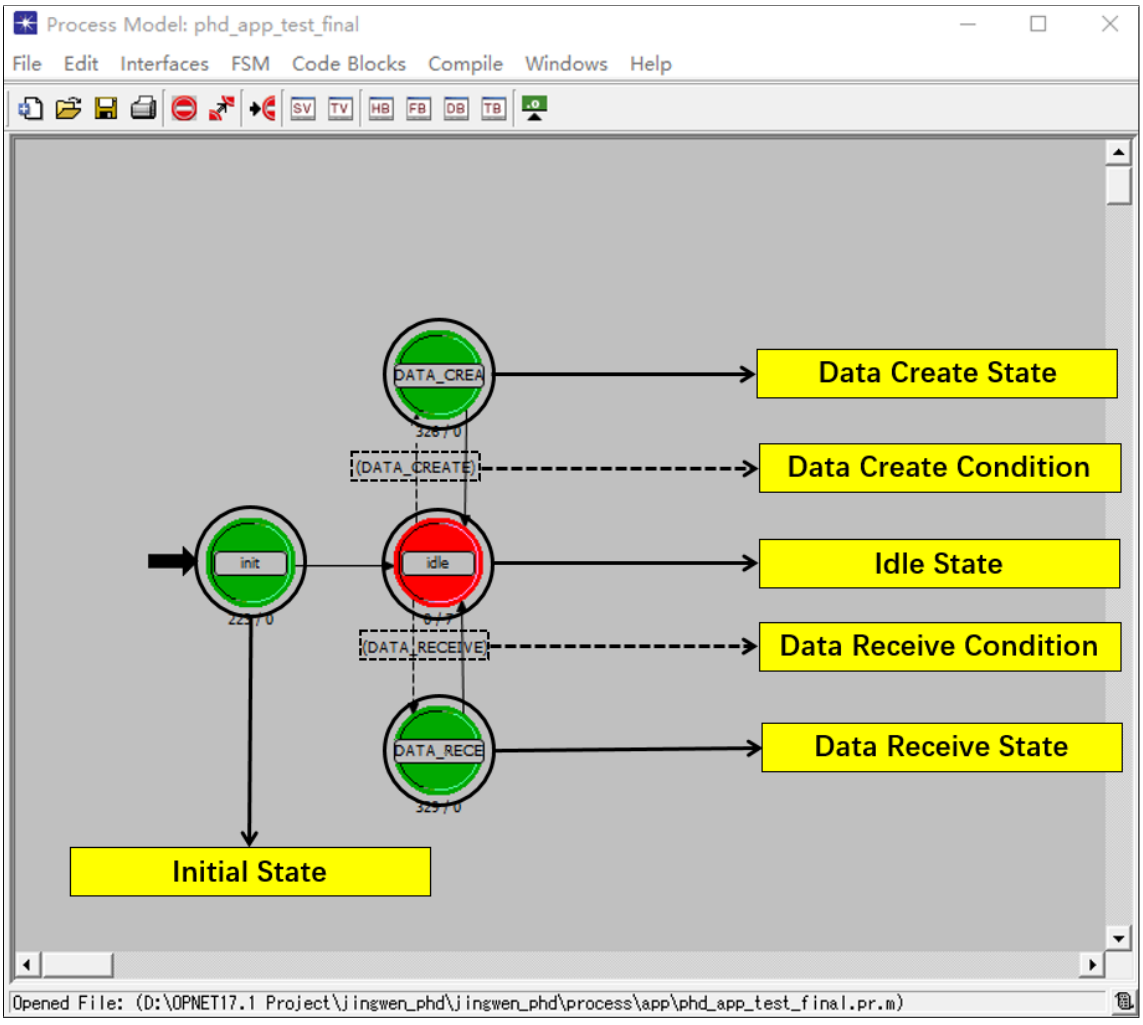


Figure 4.15: Process Model of the Application Layer Processor

- Process Model of the Network Layer

Figure 4.16 is the Process Model of the Network Layer Processor. There are two states in this process model: the "Initial State" (forced state) and the "Idle State" (unforced state). The basic function of the Process Model of the Network

Layer Processor is to decide which routing protocol should be invoked in the simulation. The selected routing protocol is executed in a Child Process which will be introduced in Section 4.2.3.2.

In the Initial State, once the simulation commences, a function called "routing_child_process_create()" is executed as shown in the Initial State panel of Figure 4.16. The detailed code of this function is implemented in the Function Block (FB) panel. In FB, a state variable called "routing_protocol_type_desicion" saved in the State Variables panel decides which routing protocol should be invoked according to the Protocol Configuration shown in Figure 4.4. In the Process Model of Network Layer Processor, several routing protocols which are used to compare with CRCPR are also implemented such as AODV, DSR, DEHAR, AODV-EHA and CWR as shown in the FB panel of Figure 4.16.

After the Initial State is finished, flow is transferred to the Idle State. Then the selected routing protocol in the simulation is invoked as a Child Process.

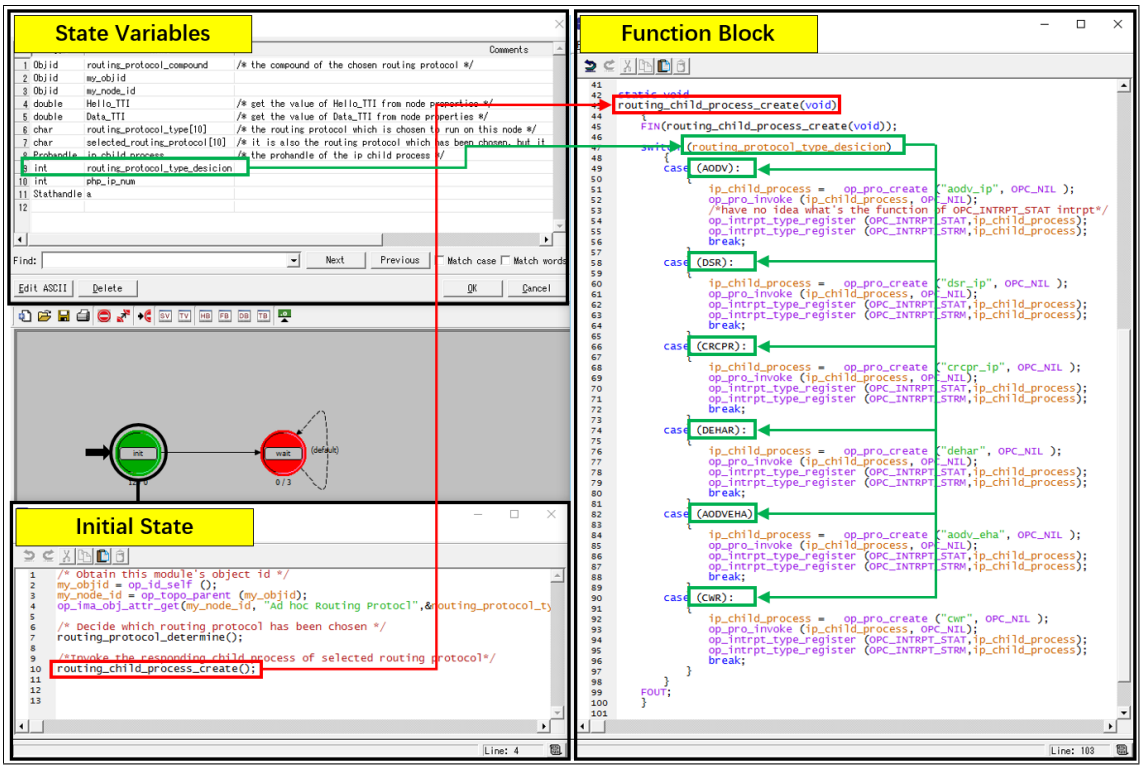


Figure 4.16: Process Model of the Network Layer Processor

4.2.3.2 Child Process

Figure 4.17 shows the Child Processes of the Network Layer Processor. Each Child Process represents one routing protocol and we only focus the CRCPR

Child Process and illustrate its implementation.

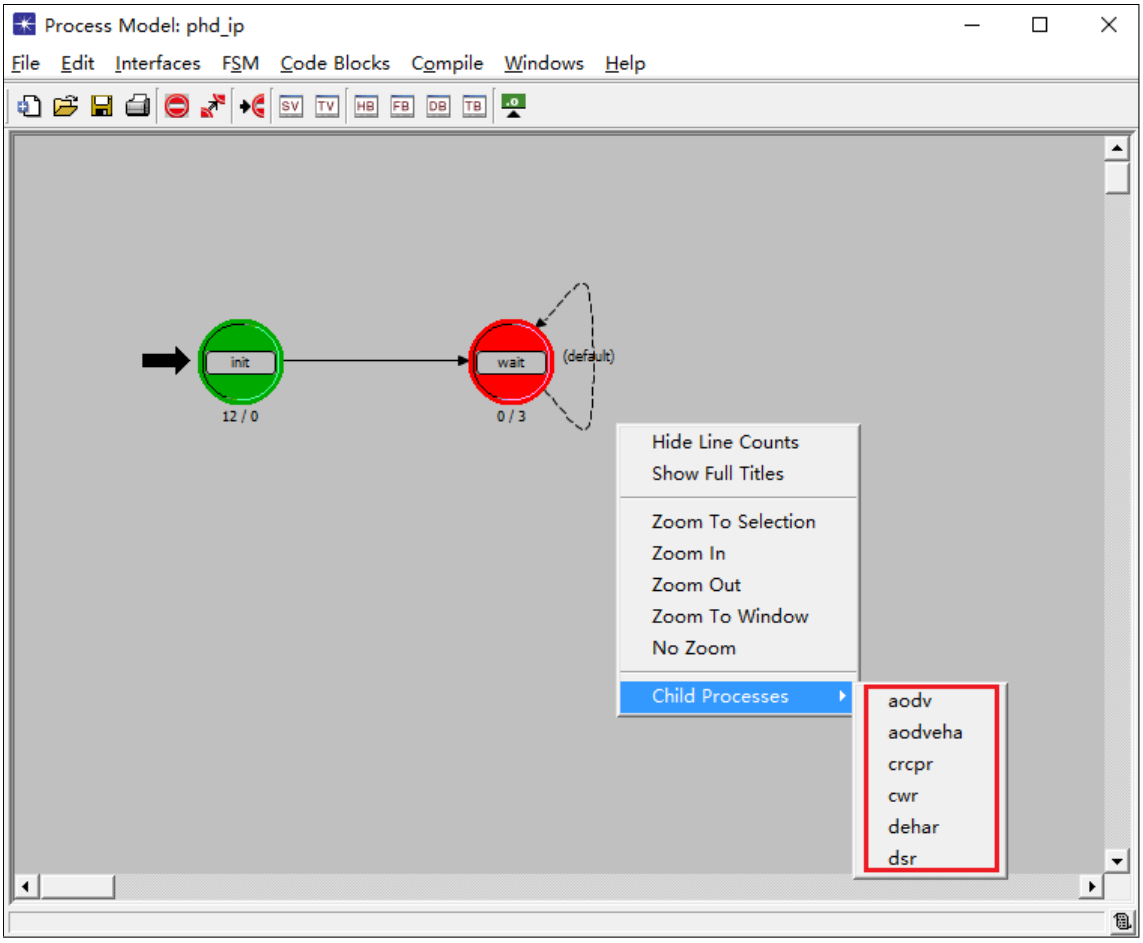


Figure 4.17: Child Process Model of the Network Layer Processor

Figure 4.18 is the Child Process of CRCPR and there are five states: the "Initial State", "CRCPR CHLO Create State", "Packet from Application State", "Packet from MAC State" and the "Idle State". The state execution and transition procedure operates as follows:

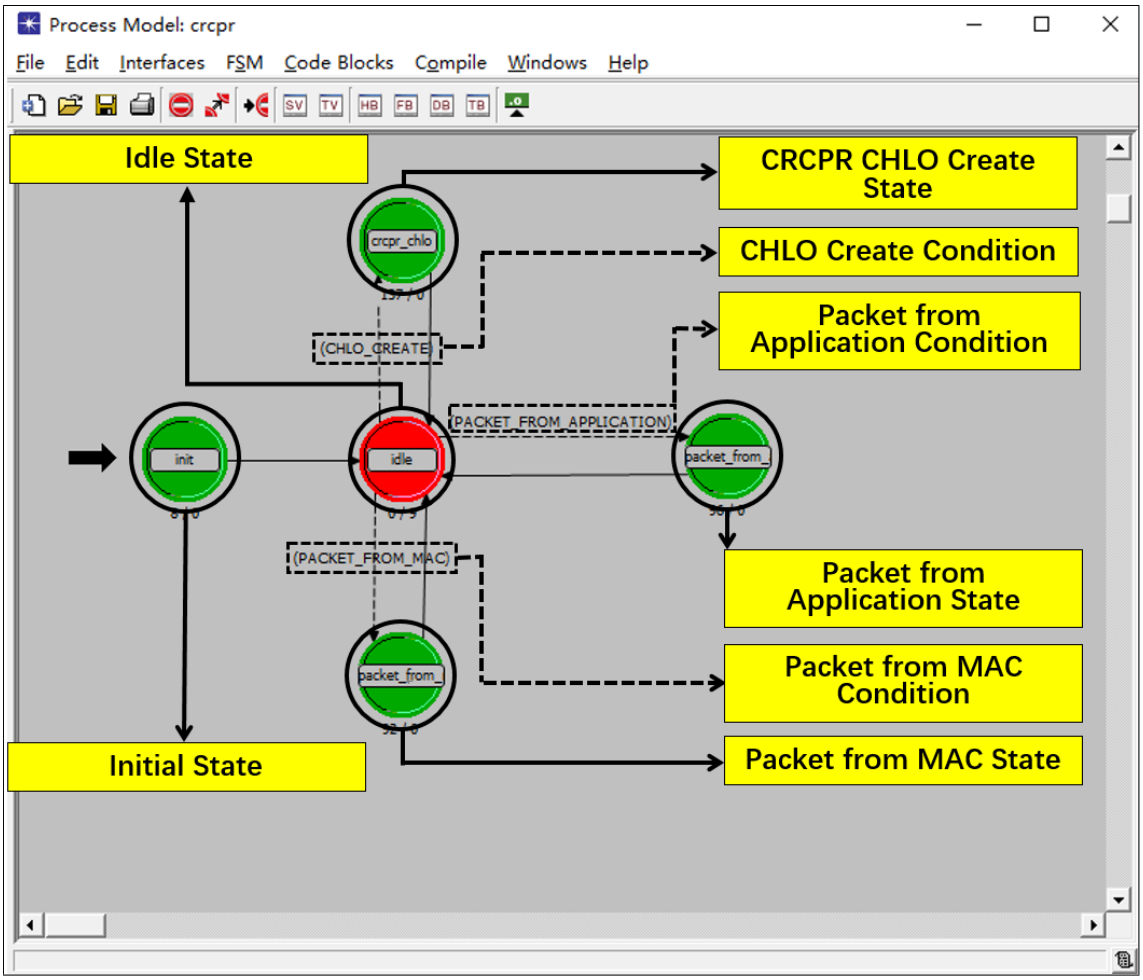


Figure 4.18: Child Process Model of CRCPR

1. At the beginning of the simulation, the Initial State is executed to initialize the memory and register variables of the statistics. After that, a self interruption is triggered for the CHLO packet broadcasting.

2. The self interrupt which satisfies the state transition condition called " CHLO Create Condition" triggers the state transition from the Idle State to the CRCPR CHLO Create State periodically. In the CRCPR CHLO Create State, a CHLO packet is created according to the CHLO design in Section 3 and Section 5.

3. If a data packet from the Application Layer is passed down to the Network Layer, the state transition condition called "Packet from Application Condition" is met. Therefore, the Packet from Application State is executed. In this state, a valid route needs to be confirmed for this data packet in the

Route Table [3] which is defined in the Header Block (HB) as shown in Figure 4.19. If a valid route is confirmed, the data will be encapsulated with the IP headers of CRCPR and passed down to MAC Layer; if there is no valid route, the route discovery procedure will be invoked and the state will be transferred to the Idle State to wait for the next event.

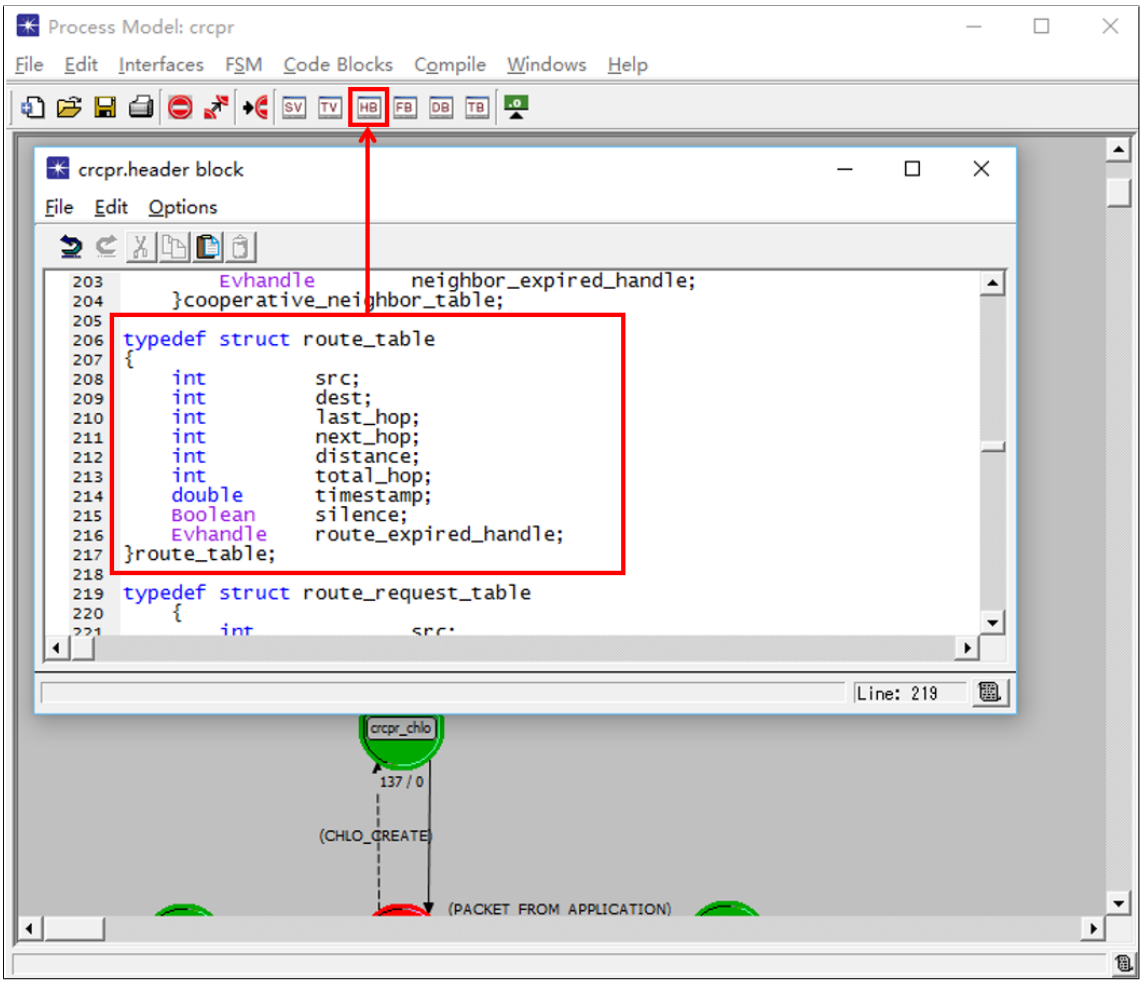


Figure 4.19: CRCPR Route Table Defined in HB

4. If a packet from the MAC Layer is passed up to the Network Layer, the state transition condition called "Packet from MAC Condition" is met and the state will be transferred from the Idle State to the Packet from MAC State. As there are many kinds of packets that can be received from the MAC Layer such as CREQ, CREP, CENF, DATA and so on, as shown in Figure 4.20, this state needs to confirm the packet type once it is received.

[3]All the other table structures such as the COP Table, Relay Table, Cooperative Neighbour Table, Route Request Table and so on are defined in the same way.

After the packet type is confirmed, the appropriate function defined in the Function Block (FB) will be executed. For example, the function in the red rectangle called "crcpr_chlo_pkt_arrival_handle()" is used to process a received CHLO packet.

The detailed design procedure for some of these functions which are responsible to process the corresponding received packets are illustrated via flow charts in Appendix A.

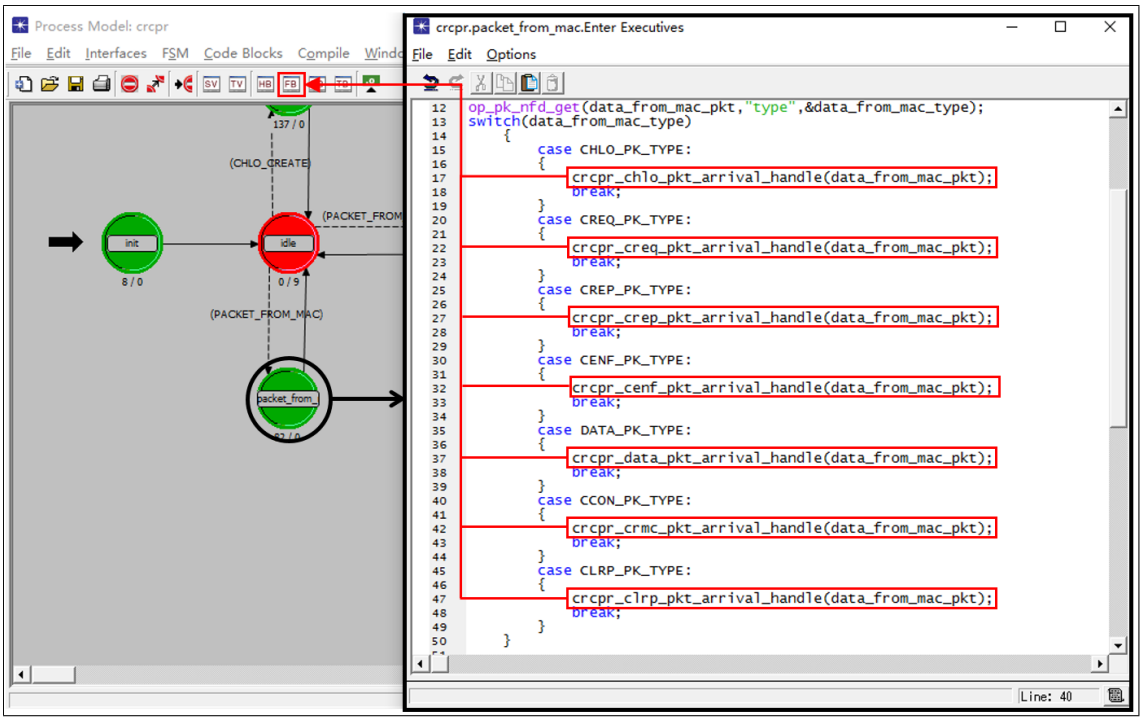


Figure 4.20: Packet Type Confirmation in the Packet from the MAC State

4.3 Validation Results

4.3.1 Node Model Validation

4.3.1.1 Scenario

As described in Section 4.2, the CRCPR architecture consists of five layers: the Application Layer, Transport Layer, Network Layer, MAC layer and Physical Layer. In order to validate the functional connections of each layer, a basic simulation scenario was set up as shown in Figure 4.21. The fixed source node is N_1 and the fixed destination node is N_2. The transmission range of each node is 30m. The simulation time is 10 minutes.

4.3.1.2 Results

The expected results for the Node Model Validation is that each layer can pass down or up the data successfully and all the data sent by the source node N_1 can be received completely by the destination node N_2.

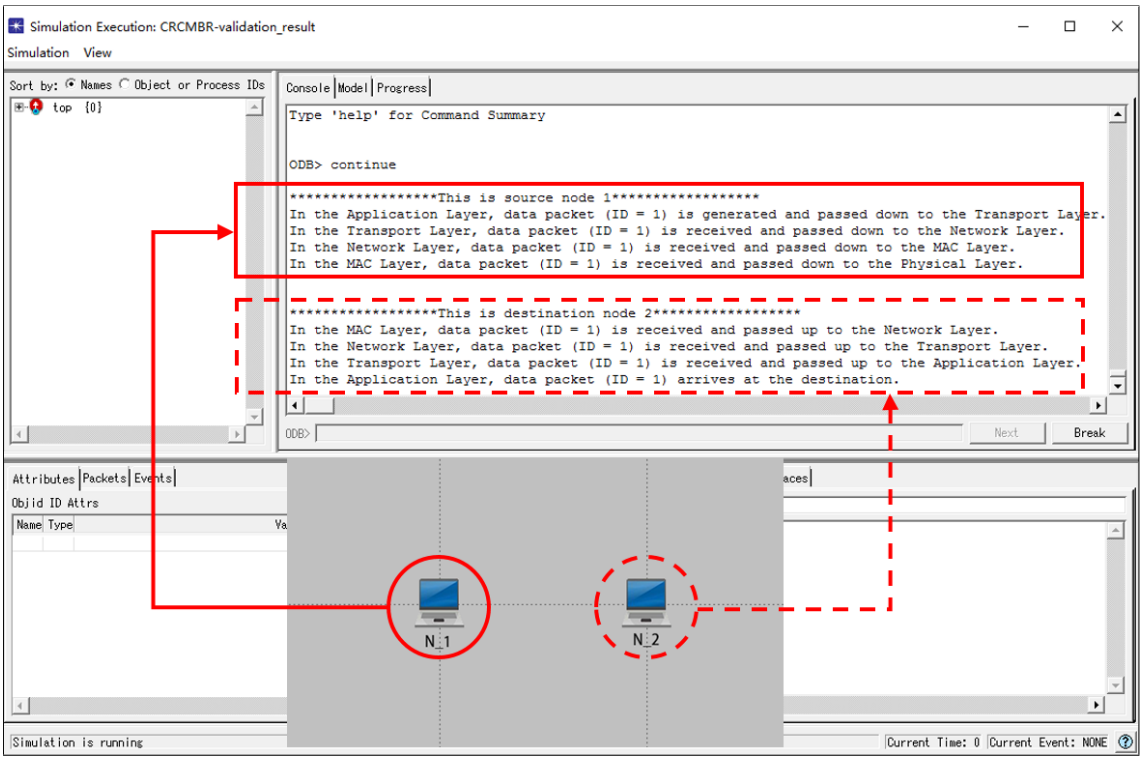


Figure 4.21: CRCPR Node Model Validation Scenario and Log

Figure 4.21 shows an example log of the source node N_1 and destination node N_2. As we can see, the data packet with ID=1 is passed down and passed up among the layers in the CRCPR Node Model successfully.

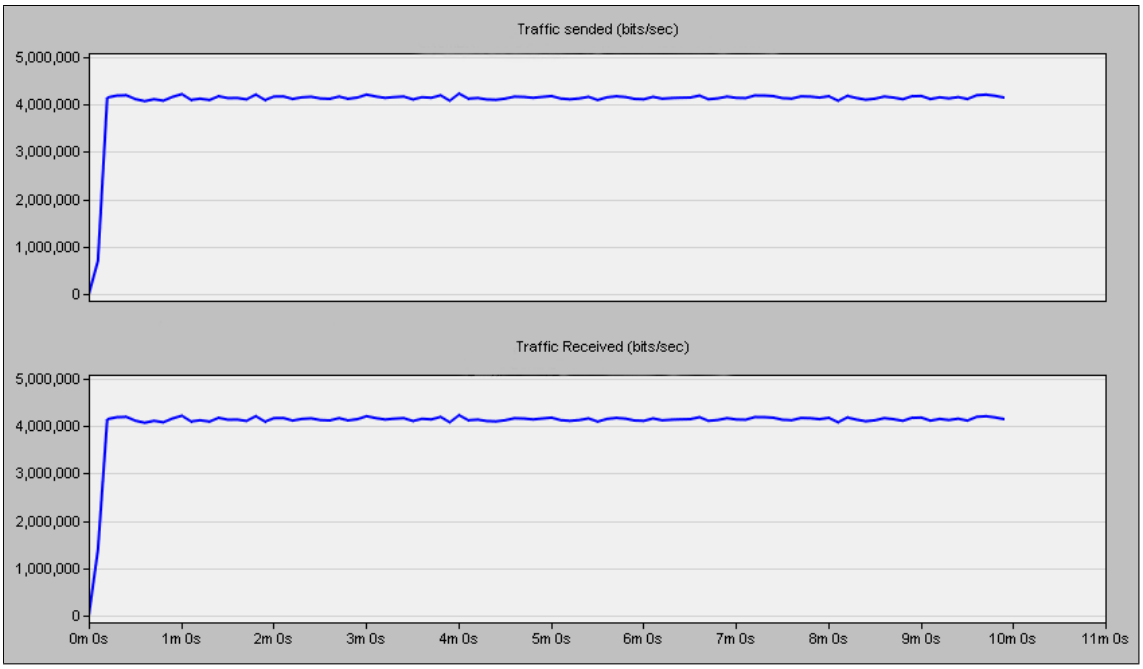


Figure 4.22: CRCPR Node Model Validation Results

Figure 4.22 shows the results about the sending traffic (the top figure) of source node N_1 and the receiving traffic (the bottom figure) of destination node N_2. As expected, the sending traffic is equal to the receiving traffic all the time during the simulation, which confirms that the data packets sent by source node N_1 are received by destination node N_2 successfully.

4.3.2 Network Model Validation

4.3.2.1 Scenario

Figure 4.23 illustrates the simulation scenario for the Network Model Validation. The source node is N_1 and the destination node is N_2. The transmission range is 30m. All the nodes are fixed excepted the colored-label nodes with *Italic* node names: N_4 and N_6. For N_4, it begins to move from the original position with coordinates (0,0) to the new position (5,0) with a constant speed 2m/s at 5 minutes into the simulation. For N_6, it begins to move from (-10,20) to the new position (-10,50) with a constant speed 2m/s at 8 minutes into the simulation.

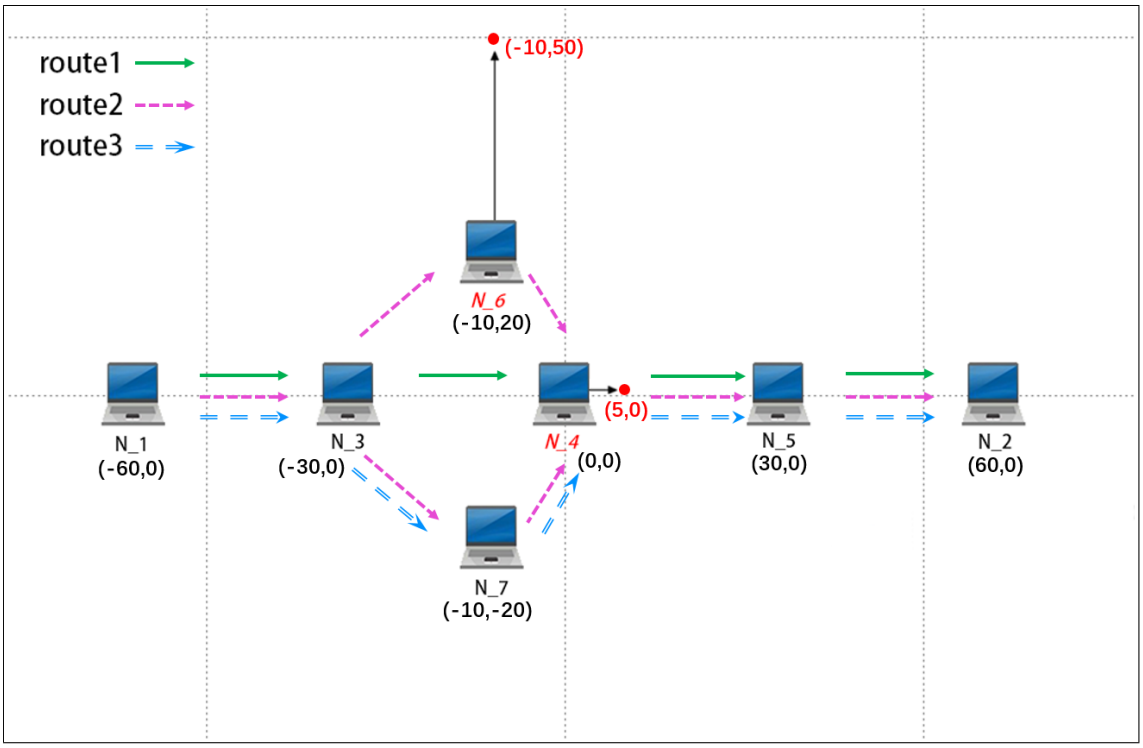


Figure 4.23: CRCPR Network Model Validation Scenario

4.3.2.2 Results

The expected results for the Network Model Validation are that before the movement of N_4, CRCPR will choose route1 to transmit data and the route for this scenario is: 1, 3, 4, 5, 2. Due to the movement of N_4 at time 5 minutes, the link between N_3 and N_4 is broken and data cannot be forwarded from N_3 to N_4 directly. However, a COP topology can be built by N_3 (Src node), N_4 (Dest node), N_6 (C Node 1) and N_7 (C Node 2), so CRCPR will choose route2 to transmit data cooperatively via two C nodes from N_3 to N_4 (The details have been provided in Section 3.3.6). In addition, N_3 and N_4 can still receive unicast hello messages from each other and the route (1, 3, 4, 5, 2) remains stable. After the movement of N_6, the COP topology does not exist but a Relay Table can be built in N_7 with Relay Neighbour 1 (N_3) and Relay Neighbour 2 (N_4), which can relay data from N_3 to N_4. In addition, the Relay Table determines the unicasting of hello messages between N_3 and N_4 and maintains the route (1, 3, 4, 5, 2). Therefore, CRCPR does not experience link breaks and chooses route3 to transmit data via the Relay Table (The details have been provided in Section 3.3.6). Throughout, the route in the simulation scenario will not be broken due

to node movements.

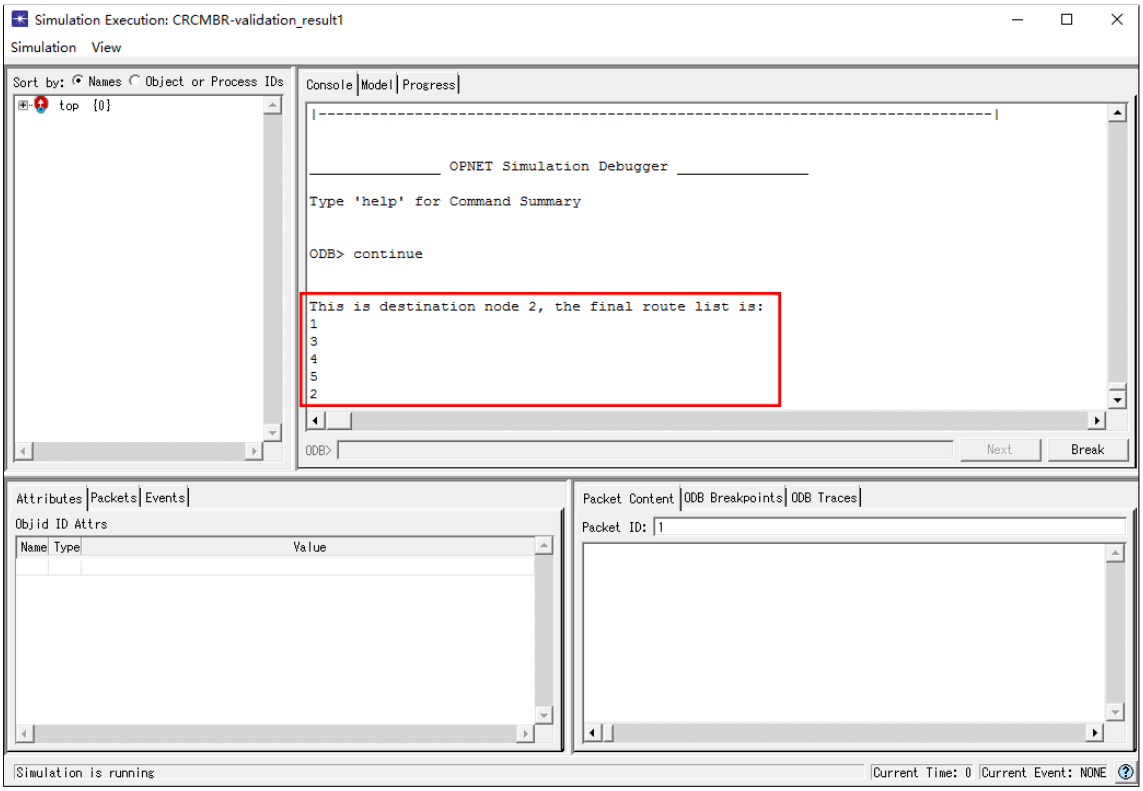


Figure 4.24: CRCPR Network Model Validation Log

Figure 4.24 shows the route selected by CRCPR during the simulation, which is the same as the predicted route.

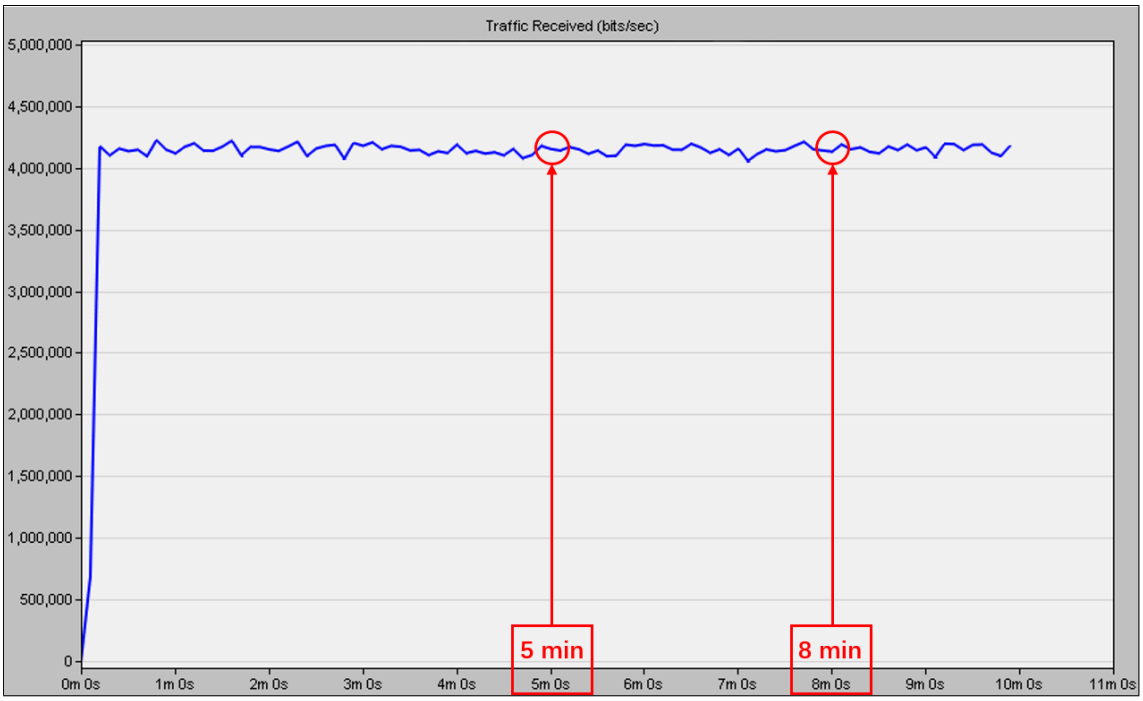


Figure 4.25: CRCPR Network Model Validation Results

Figure 4.24 illustrates traffic received at destination N_2. As we can see, at both movement positions: 5 minutes and 8 minutes into the simulation, the traffic remains stable and no link breaks happen during the whole simulation. This confirms that the COP Table and Relay Table perform well in terms of robustness against node mobility and the results are as expected.

Chapter 5

Hello Message Scheme

This chapter proposes an improved hello message broadcasting scheme named Adjust Classified Hello Scheme (ACHS) in Mobile Ad-Hoc Networks (MANETs), which cannot only be deployed into CRCPR to reduce the quantity of broadcasting hello messages, but adapt to any routing protocol with hello messages to improve broadcasting efficiency. ACHS categorizes nodes into different classes based on node roles (on the route or off the route) and properties (node mobility or nodes with special functions). Each class in ACHS can be configured with different strategies in terms of broadcasting interval and hello message format.

5.1 Introduction

Due to node mobility in MANETs, how to find a suitable route to efficiently transmit data from the source node to the destination node has been the focus of a lot of research interests [169] [170][171][172]. Discovering and maintaining neighbours via broadcasting hello messages has been regarded as an important operation in order to achieve good performance during route discovery, route reconstruction and route maintenance. However, given the limitation of device batteries, improving broadcasting efficiency becomes a key point to save transmission energy and improve the performance of the whole network. The Periodic Hello Message Scheme (PHMS) is one of the traditional methods to support neighbour table updating and maintenance. Specifically, nodes periodically advertise hello messages to their neighbours to indicate their existence. However, the fixed interval in PHMS is not suitable for real network circumstances, because a hello message with a short periodic interval may cause unnecessary congestion, while a long interval may cause slow response to network changes. For example, according to the routing protocols [33][173], the neighbour table in one node is used to verify

whether one of its neighbours is the next hop on an active route. In this case, the neighbour table helps with route establishment and link break detection. For other cases in [174][175], nodes depend on the neighbour table to forward data, which means that besides link detection, the neighbour table also contributes to the routing function. In order to meet neighbour table requirements for the above routing protocols, a short broadcasting interval for hello messages is preferable. However, a short interval can also lead to resources exhausting quickly, like the battery [176], sometimes even causing data congestion. Based on the above issues, the goal of this work is to propose a new hello message broadcasting scheme called ACHS which provides an efficient method to update neighbour tables and improve the overall network performance.

5.2 Hello Message Schemes for Protocols

In MANETs, certain hello message broadcasting schemes have been proposed already, mainly of two types: periodic broadcasting and reactive broadcasting.

5.2.1 Periodic Hello Message Scheme

Because of the simplicity of its implementation, PHMS (Periodic Hello Message Scheme) [177] is a traditional broadcasting scheme which has been widely adopted for the neighbour table updates and maintenance in MANETs. When a node receives a hello message from its neighbours, it creates a new entry or updates the corresponding entry in its neighbour table. Within a pre-defined period of time, if the node does not get any hello message from the same neighbour, the entry in the neighbour table will be deleted. The neighbour node can use the entry in the neighbour table to establish and maintain a route. The design of PHMS is based on on-demand routing protocols such as AODV [173] and ABR [178].

5.2.2 Reactive Hello Message Scheme

In RHMS (Reactive Hello Message Scheme), the nodes including the source node and the intermediate nodes only build the neighbour tables when they need to. For the source node, once the data from the Application Layer comes to the Network Layer, all the data will be buffered before the neighbour connectivity procedure

is finished. The neighbour connectivity procedure is as follows: One hello request packet will be broadcast by the source node and the maximum attempts is set as the parameter MAX_RETRIES [177]. Within the RESP_WAIT_TIME period (the RESP_WAIT_TIME period is the maximum period to wait for a valid hello response message), if no hello response message is received, the hello request packet will be rebroadcast. As long as the neighbour table is set up by receiving the hello response packet, the source node will broadcast the route request packet to discover a route. After the route is built, the data in the buffer will be sent out to the next hop. For intermediate nodes, if the hello messages from the other nodes are received, they will be triggered to broadcast hello messages. Because of node mobility, the neighbour table entry will be deleted according to the NBR_VALID_TIME (NBR_VALID_TIME is a maximum period to keep the neighbour table valid before the next hello message is received). Although RHMS reduces the broadcasting hello message frequency, buffering packets leads to long end-to-end delays.

5.3 Adjust Classified Hello Scheme

Given all of the above studies, the ACHS (Adjust Classified Hello Scheme) improves the hello broadcasting efficiency mainly from three aspects: Firstly, ACHS exploits the "Reserved" field of the hello message which categorizes nodes into different classes with different broadcasting intervals. Secondly, ACHS employs the "Hello Embed" scheme to embed a hello message into a data packet to reduce the broadcasting hello messages. Thirdly, ACHS dynamically schedules two supported hello message formats (simple hello format and rich hello format) and reduces redundant hello messages without any substantial influence on the network performance.

5.3.1 Structure Overview

ACHS supports two hello message formats for improving broadcasting efficiency: a simple hello format, shown in Figure 5.2, and a rich hello format, shown in Figure 5.3. The simple hello format is the traditional hello design used in AODV [173] and ABR [178]. Compared with the simple hello format, the rich hello format is designed by adding the field called *neighbour Address [n]* to carry the neighbour

information which is used in CRCPR and OSPF [179]. In order to dynamically decide the the hello message interval, the simple hello format and rich hello format are modified to carry more information (*Hello Interval* and *Node Velocity*) by utilizing the "Reserved" fields. The modified hello messages are shown in Figure 5.4 and Figure 5.5.

ACHS uses a classification method to categorize nodes into different classes. The node class tree is shown in Figure 5.1. According to the different classes, different broadcasting intervals will be assigned with the information of modified fields: *Interval* and *Velocity*. The example of the basic classification strategy is given in Figure 5.6. Those nodes on the route are defined as Class 1 and the nodes off the route belong to Class 2. However, in some protocols, some nodes have special functions and play an important role in improving the network performance, such as the nodes in the COP topology in CRCPR. Therefore, considering extendability, an extended classification strategy can be deployed into Class 1 to obtain two sub-classes shown in Figure 5.7: Class 1A (nodes on the route with special functions) and Class 1B (nodes on the route without special functions). The extended classification allows the ACHS design to be easily expanded to satisfy other special demands for the hello messages. How to decide different broadcasting intervals for the nodes in different classes will be introduced in Section 5.3.2 and Section 5.3.3.

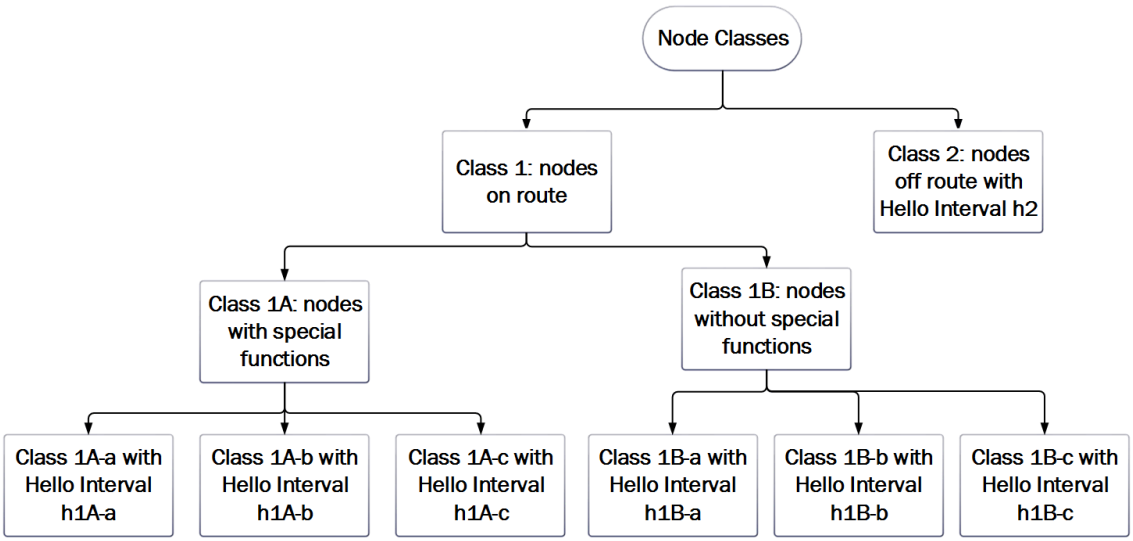


Figure 5.1: Node Class Tree in ACHS

	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	*Version*				*Type*				*Reserved*								*Length*															
4	32	*Src*																															
8	64	*Dest*																															

Figure 5.2: Simple Hello Format

	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	*Version*				*Type*				*Reserved*								*Length*															
4	32	*Src*																															
8	64	*Dest*																															
12	96	*Neighbor address [1]*																															
16	128	...																															
20	160	*Neighbor address [n]*																															

Figure 5.3: Rich Hello Format

5.3.2 Class 1: Nodes on the route

All the nodes belonging to Class 1 are on the route, which means they will forward data from the source node towards the destination node. The basic principle to improve hello message broadcasting efficiency for Class 1 is called the "Hello Embed" scheme, which means the contents of hello message can be embedded into data packets to reduce the number of broadcasting hello messages. Generally, a data packet can only be processed by the valid next hop on the route. In fact, due to the phenomenon of broadcasting transmissions in wireless communication, a data packet can be overheard by all neighbours of the sending node. The original data part can be retrieved by the valid next hop, but when the neighbours of the sending node which are not the valid next hop hear the transmission, they can extract the hello contents to update their neighbour information. If the data

	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	*Version*				*Type*				*Interval*				*Velocity*				*Length*															
4	32	*Src*																															
8	64	*Dest*																															

Figure 5.4: Modified Simple Hello Format

	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	*Version*				*Type*				*Interval*				*Velocity*				*Length*															
4	32	*Src*																															
8	64	*Dest*																															
12	96	*Neighbor address [1]*																															
16	128	...																															
20	160	*Neighbor address [n]*																															

Figure 5.5: Modified Rich Hello Format

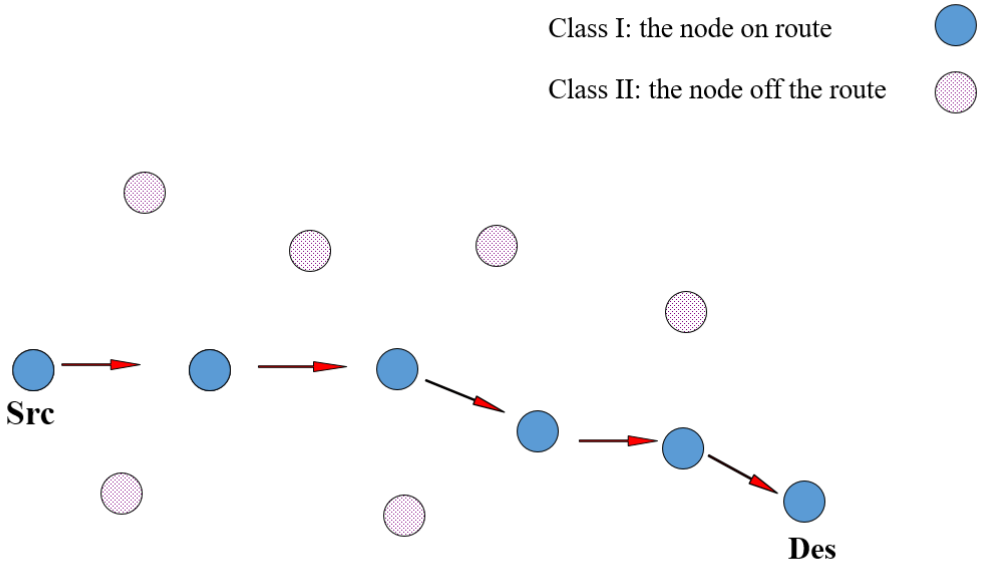


Figure 5.6: Basic Classification Strategy

sending rate is higher than the hello broadcasting interval, no more broadcasting hello messages need to be sent and the "Hello Embed" scheme is triggered. For example, this occurs when a current neighbour node moves away from the sending node or a new node moves closer to the sending node and becomes a new neighbour. If the data sending rate is lower than the hello broadcasting interval, which means the embedded hello information in the data packet cannot maintain the neighbour relationship, broadcasting hello messages are still needed. In this case, how to set the hello broadcasting interval dynamically is considered in Table 5.1. In ACHS, when the data sending rate is lower than the hello broadcasting interval, a new hello broadcasting interval under the "Hello Embed" scheme is employed according

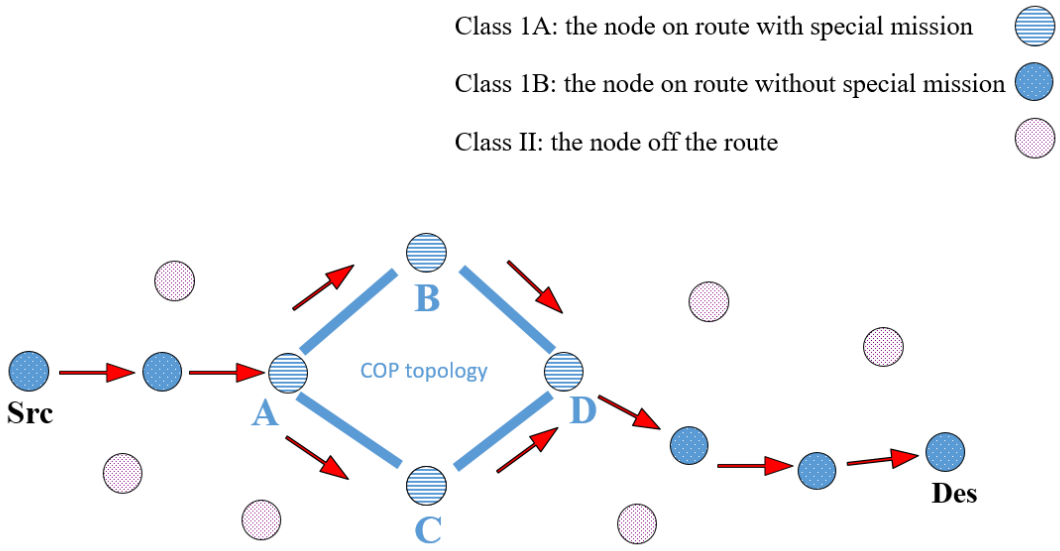


Figure 5.7: Expansive Classification Strategy

to Table 5.1. Three different hello intervals are set according to three different time ranges. t_d is defined as the time when the data with embedded hello content has been sent successfully and at this moment the broadcasting hello message which should have been sent at time t_2 is cancelled. A new hello broadcasting interval $Tchange$ will be set by comparing the threshold time T_{th} and $|t_2 - t_d|$ in different time ranges.

Table 5.1: Hello Embed Time Interval Parameter

Time Range	Time Interval from td to next hello
$T_{th1} > t_2 - t_d \geq 0$	$Tchange_1$
$T_{th2} > t_2 - t_d \geq T_{th1}$	$Tchange_2$
$t_2 - t_d \geq T_{th2}$	$Tchange_3$

For each time range, there are two thresholds: the lower decision threshold T_{th1} and higher decision threshold T_{th2}. As we can see the higher decision threshold in one time range is also the lower decision threshold in the next time range, therefore only the lower decision threshold is considered in the following discussion. Assume that after a data packet with embedded hello content is sent successfully, the broadcasting hello message can still maintain the neighbour relationship effectively, thus Equation (5.1) should be satisfied:

$$\begin{aligned} &Hello\_Interval - Lower\_Decision\_Threshold + Tchange \\ &\leq Allow\_Hello\_Loss \times Hello\_Interval \end{aligned} \tag{5.1}$$

where the left side represents the time difference between the time when the current data with the embedded hello content has been sent successfully and the time when the broadcasting hello message is about to sent; the right side represents the expire time of the neighbour table entry.

If the $Hello\_Interval$ is regarded as the minimum Δt, the lower decision threshold can be seen as $B \times Hello\_Interval$ where B is a coefficient. (For example, if the time range $T_{th1} > t_2 - t_d > 0$ is considered, 0 as the lower time threshold will be equal to $B \times Hello\_Interval$). Then the value of $Tchange$ is equal to $C \times Hello\_Interval$ where C is another coefficient. (For example, if the time range $T_{th1} > t_2 - t_d > 0$ is considered, then $Tchange_1$ will be equal to $C \times Hello\_Interval$). Therefore, Equation (5.1) can be modified to Equation (5.2)

$$\begin{aligned} &Hello\_Interval - B \times Hello\_Interval \\ &+C \times Hello\_Interval \\ &\leq Allow\_Hello\_Loss \times Hello\_Interval \end{aligned} \tag{5.2}$$

If the $Hello\_Interval$ in both sides are removed, we obtain Equation (5.3)

$$C \leq Allow\_Hello\_Loss - 1 + B \tag{5.3}$$

Equation (5.3) can be used to calculate the coefficient Conce $Allow\_Hello\_Loss$ and B have been defined. Then a new hello broadcasting interval $Tchange$ can be obtained as $Tchange_{(n=1,2,or3)} = C \times Hello\_Interval$, where n is decided by the *Time Rage* in Table 5.1.

5.3.2.1 Nodes on the route with Special Functions: Class 1A

Considering extendability, Class 1 contains two sub-classes: Class 1A (nodes on the route with special functions) and Class 1B (nodes on the route without special functions). Both of these sub-classes employ the "Hello Embed" scheme. As described before, if the data rate is higher than the broadcasting hello interval, no matter whether the simple hello format or the rich hello format is involved in the protocol, we only need to embed the hello content into data packets to maintain the neighbour relationship. However, when the data rate is lower than

the broadcasting hello interval, broadcasting hello messages are still needed. How to schedule broadcasting of simple hello messages and rich hello messages will be described separately.

In Class 1A, if the protocol only employs simple hello messages, it will not change the broadcasting schedule and only broadcasts simple hello messages; If rich hello messages are involved in the protocol, the scheduling of broadcasting simple hello messages and rich hello messages follows this principle: if the neighbour information of one node changes (a new neighbour joining or an old neighbour leaving), the node will broadcast a rich hello message. Otherwise, a simple hello messages will be sent. In order to select a specific hello interval to maintain the route and the special functions effectively, the nodes in Class 1A are further classified into three subclasses: Class 1A-a, Class 1A-b and Class 1A-c. The three subclasses are determined by a comparative result of the Determination Velocity (V_{de}) and Threshold Velocity (V_{th}) which is shown in Table 5.2. V_{th} is a scenario-selectable value [1]. Because there are three subclasses, two different values of V_{th} are needed. By comparing with V_{th}, V_{de} can be used to decide which subclass a node belongs to. We use CRCPR as an example to explain how to calculate V_{de}. In CRCPR, there are four Intermediate Nodes (INs) in a specific cooperative topology and they build a Cooperative Table to implement the special functions of cooperative transmission. Therefore, the neighbour information for the INs plays an important role and a smaller hello broadcasting interval should be set. If one of the four INs in the cooperative topology detects velocity changes in the other three nodes, it will calculate the average value as the Determination Velocity (V_{de}). The node velocity can be obtained according to Equation 5.4:

$$V = \frac{\sqrt{(X_{new} - X_{old})^2 + (Y_{new} - Y_{old})^2}}{t_{new} - t_{old}} \tag{5.4}$$

In Equation 5.4, t_{old} and t_{new} are the old observation time and new observation time respectively. X_{new} and Y_{new} are the X position and Y position of a node at time t_{new}. X_{old} and Y_{old} are the X position and Y position of a node at time t_{old}. After the velocity V of a node is calculated, it will be recorded in the revised field "Velocity" of the hello message and used by its neighbours to calculate V_{de}.

After V_{de} is calculated, the hello interval is confirmed by comparing it with

[1] Currently, V_{th} needs to be set manually. In future work, an automatic value setting scheme could be devised.

V_{th}. As the mobilities of the nodes in different cooperative topologies might not be the same, the hello interval will be decided according to Table 5.2. However, if one node belongs to more than one cooperative topology at the same time, then the hello interval will be set to the smallest one.

Table 5.2: Hello Interval Decision for Class 1A

Velocity Range	Subclass	Hello Interval
$V_{th1} \geq V_{de}$	1A-a	h1A-a
$V_{th2} > V_{de} \geq V_{th1}$	1A-b	h1A-b
$V_{de} \geq V_{th2}$	1A-c	h1A-c

A specific example is shown in Figure 5.8. Node A, B, C and D build up a cooperative topology. Node A has the velocity V_1 and it obtains the velocities of node B (V_3) and D (V_4) by a broadcast hello message. Node A can also get the velocity of node C (V_2) via a unicast hello message relayed from node B or D. Node B, C and D obtain the velocities of each other in the same way. The average velocity for each IN in cooperative topology can be calculated: $V_{de} = (V_1 + V_2 + V_3 + V_4)/4$. Finally, we can decide the hello interval for INs in the cooperative topology according to Table 5.2.

5.3.2.2 Nodes on the route without Special Functions: Class 1B

The nodes on the route without special functions will be regarded as Class 1B for hello message scheduling.

If the protocol only employs the simple hello message mechanism, it does not change the broadcasting schedule and keeps broadcasting simple hello messages; if rich hello messages are involved, the schedule for broadcasting simple hello messages and rich hello messages obey this principle: after a node has sent two simple hello messages [2] and its neighbour information changes at this moment (a new neighbour join or an old neighbour leaves), the node will broadcast a rich hello message. Otherwise, the node broadcasts a simple hello message.

Although nodes in Class 1B are not with special functions, they can still influence data transmission as they are on the route. Like Class 1A, the nodes

[2] "Two" simple hello messages is a value based on experience which can effectively reduce the need for rich hello messages as well as maintaining the special function of the routing protocol. In the future, more scenario parameters could be involved in determing the hello interval automatically.

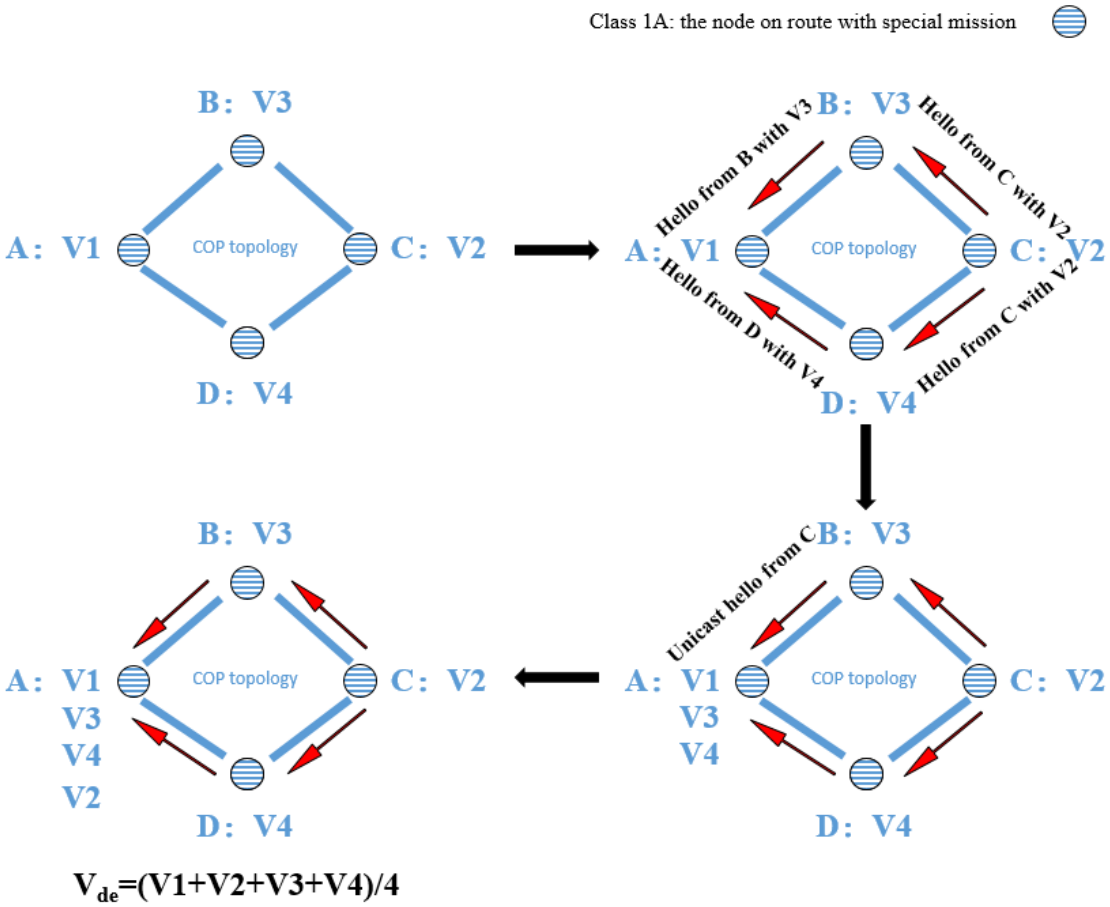


Figure 5.8: Setting Determination Velocity in CRCPR

in Class 1B are also classified into three subclasses: Class 1B-a, Class 1B-b and Class 1B-c. The three subclasses are decided by the comparison result between the Determination Velocity (V_{de}) and Threshold Velocity (V_{th}) which is shown in Table 5.3. V_{de} in Class 1B is calculated according to the following principle: A node will detect all its neighbours' velocities and choose a neighbour with the maximum velocity as the reference node at the first observing time interval[3]. If the maximum velocity changes to a larger value or changes to a smaller value in the next two time intervals [4] observed, then the average of these three speeds will be used as V_{de}. During the period of observation, if the reference node with the maximum velocity changes to another node, a new observation will be set.

[3]The first observing time will start once the nodes without special functions are selected on the route.

[4]This is a simple scheme to decide the variation trend in velocity. More than two observing intervals may lead to slow response to network changes. In future work, a more precise scheme could be explored to perform the velocity prediction.

An example shown in Figure 5.9 is used to illustrate the above process. At the first observing time interval, node B observes that the velocity of node A has the maximum value V_1 among its neighbours. So V_1 will be regarded as the comparable velocity V_c. By a similar definition, V_2, V_3, ... ,V_m is the velocity observed at the *second*, *third*, ... , m_{th} observing time intervals. If $V_{m-1} > V_c$, $V_m > V_c$ and $V_{m+1} > V_c$ or $V_{m-1} < V_c$, $V_m < V_c$ and $V_{m+1} < V_c$, then node B can obtain its own determination velocity: $V_{de} = (V_{m-1} + V_m + V_{m+1})/3$. Meanwhile, V_{m+1} will be regarded as a new comparable velocity V_c and the next observation commences.

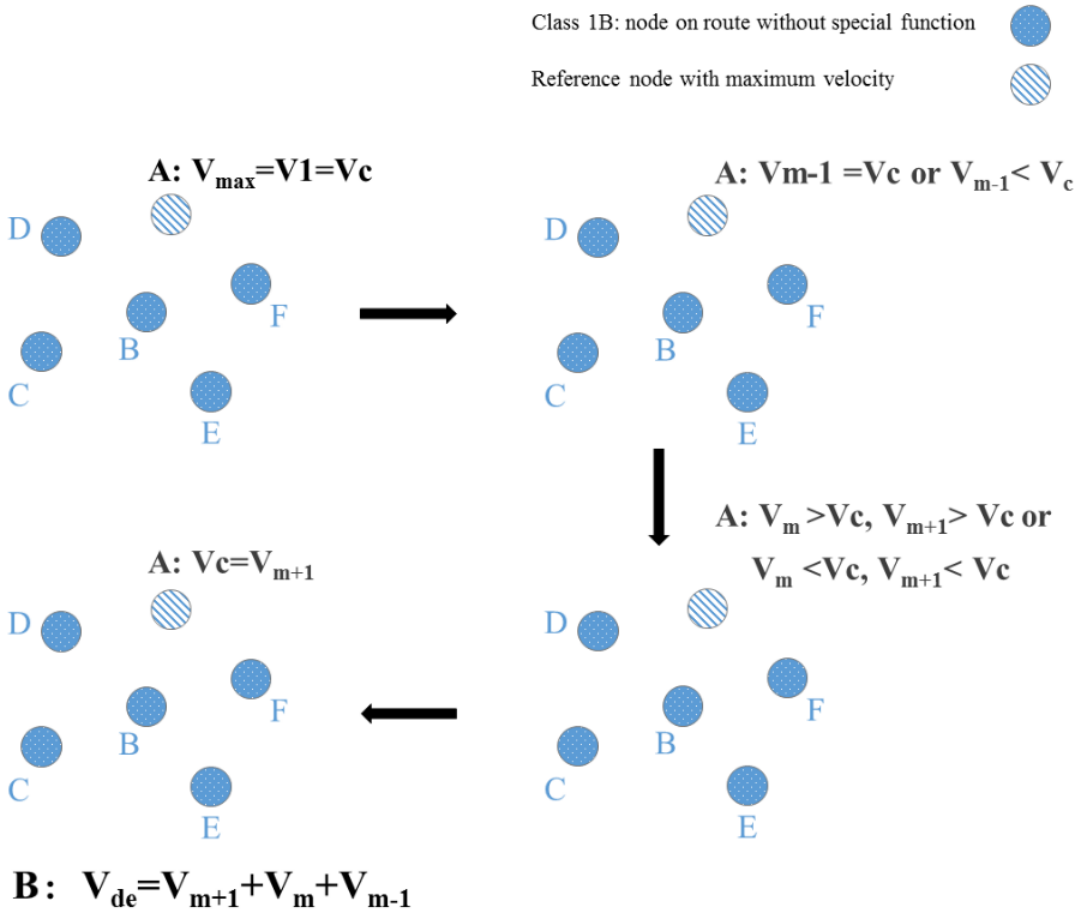


Figure 5.9: Setting Determination Velocity for Class 1B

5.3.3 Class 2: Nodes off the route

All the nodes that are not on the route are classified as Class 2. Since the movement of the nodes in this class rarely influences the route status, they have the

Table 5.3: Hello Interval Decision for Class 1B

Velocity Range	Subclass	Hello Interval
$V'_{th1} \geq V_{de}$	1B-a	h1B-a
$V'_{th2} > V_{de} \geq V'_{th1}$	1B-b	h1B-b
$V_{de} \geq V'_{th2}$	1B-c	h1B-c

lowest priority and the largest hello broadcasting interval. If the protocol only employs simple hello messages, it will not change the broadcasting schedule and keeps broadcasting simple hello messages; if rich hello messages are supported in the protocol, the schedule for broadcasting simple hello messages and rich hello messages will be according to this principle: after a node has sent four simple hello messages [5] and its neighbour information changes at this moment (a new neighbour joining or an old neighbour leaving), the node will broadcast a rich hello message for the next interval. Otherwise, a simple hello message will be broadcast. The hello interval for Class 2 is defined as $h2$ which is a scenario-selectable value [6] and longer than the broadcasting hello interval in any other class.

5.4 Summary

In this chapter, we propose an Adaptive Classified Hello Scheme (ACHS) for the routing protocols designed for MANETs. It aims at improving hello broadcasting efficiency to reduce network congestion and save energy. Basically, the design uses a classification method to adjust nodes' hello broadcasting intervals. Then, a "Hello Embed" scheme is designed for the nodes on the route to reduce unnecessary hello messages whilst still providing reasonable link break detection. As ACHS supports two hello message formats: the simple hello format and the rich hello format, therefore the scheduling of these two kinds of hello messages is explored. In summary, ACHS is an easy-deployed and extensible scheme which can be readily included in routing protocols using a hello scheme in MANETs to help improve broadcasting efficiency, but also can be adapted to different scenarios by exploring

[5] "Four" simple hello messages is a value based on experience which can effectively reduce the need for rich hello messages as well as maintaining special functions of the routing protocol. In future, more scenario parameters could be involved in deciding the hello interval automatically.

[6] Different scenarios set this value according to different principles. For example, if the mobility of a scenario is not very high, this value can be set longer to effectively reduce the broadcasting hello message volume.

more classification methods. The performance of the scheme will be accessed in Chapter 6.

Chapter 6

Analytical Comparison and Simulation Results

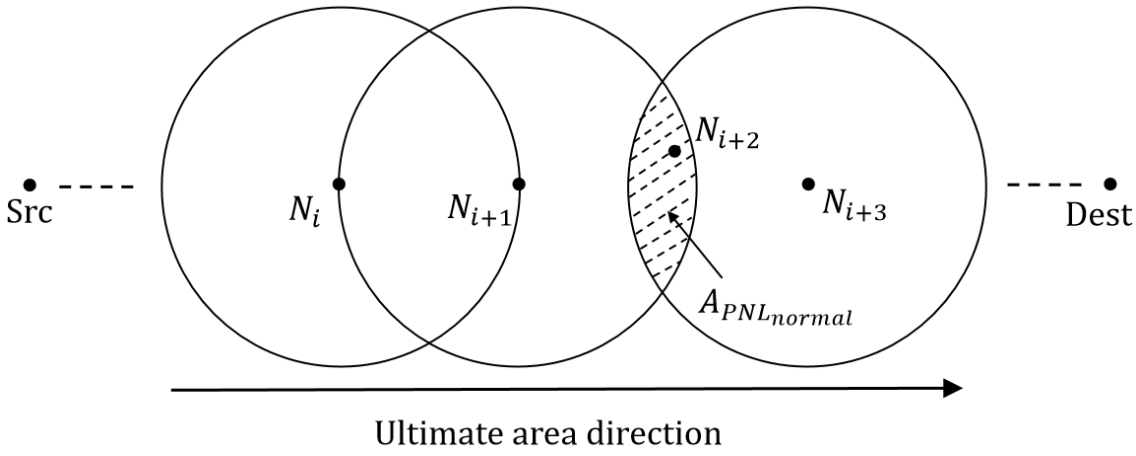
This chapter will investigate the performance of CRCPR via analysis and simulation. In the analysis, the concept of Potential Next-hop Location (PNL) is proposed to prove CRCPR can improve robustness against mobility. Using simulations, several scenarios are explored using an OpNET simulation platform. Two classic MANET routing protocols (AODV and DSR) and one cooperative routing protocol (CWR) are compared with CRCPR to test the route robustness while a further two energy-aware routing protocols (DEHAR and AODV-EHA) are simulated to evaluate the energy saving feature of CRCPR. Additionally, the performance of the Adjust Classified Hello Scheme (ACHS) is also evaluated in this chapter. Stable scenarios and mobile scenarios are set up to compare ACHS with the Periodic Hello Message Scheme (PHMS) and the Reactive Hello Message Scheme (RHMS) with the AODV protocol. By deploying ACHS into CRCPR, the hello message overhead of CWR versus CRCPR is explored as well.

6.1 Potential Next-hop Location (PNL)

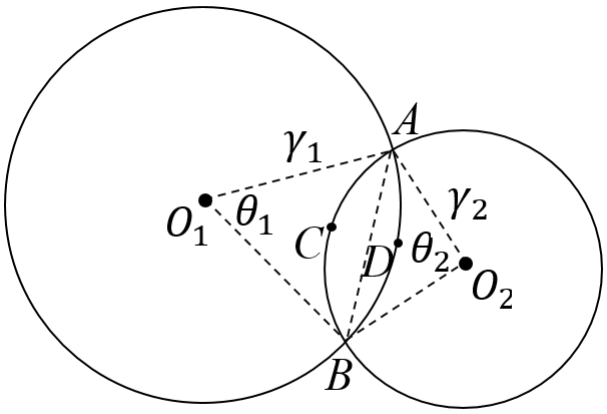
In Section 3, we have described all the procedures related to CRCPR and explained how CRCPR is able to improve robustness against mobility. In this section, we mathematically demonstrate this benefit with the help of a new concept called the Potential Next-hop Location (PNL). In order to compare a normal MANET and a COP-topology-recognized MANET, we use the same node names for the nodes located in the same position in these two types of MANET as shown in (a) of Figure 6.1 and (a) of Figure 6.2. We use (a) of Figure 6.1 as an example to demonstrate the PNL for the node N_{i+2}. Firstly, the concept of ultimate area

direction is proposed to define the PNL, which is the direction of the line from the source node to the destination node. Secondly, we identify the node N_i, node N_{i+1} and node N_{i+3} at fixed positions along the ultimate area direction. Finally, the shaded area in (a) of Figure 6.1 is the PNL for the node N_{i+2}. As we can see, the larger is the PNL between two adjacent nodes, the greater the mobility that can be supported.

The PNL in a normal MANET and a COP-topology-recognized MANET can be modeled as a mathematical problem of the intersection of circles. For the normal MANET, the PNL is an intersection of two circles as illustrated in (b) of Figure 6.1. We define r_1 and r_2 as the radius of these two circles and θ_1 and θ_2 as the angle subtended by the segment at the center. The area of the circular segments is obtained from Equation (6.1):



(a)



(b)

Figure 6.1: PNL in a Normal MANET

$$
\begin{aligned}
A_{Segment_{ADB}} =& A_{\text{Sector}_{AO_1B}} - A_{Triangle_{AO_1B}} \\
=& \int_0^{\theta} \int_0^{r} \tilde{r} d\tilde{r} d\tilde{\theta} - {}^{1}\!/_{2} \overline{AO_1 BO_1} \sin\theta = {}^{r^2}\!/_{2}(\theta - \sin^{-1}\theta)
\end{aligned} \tag{6.1}
$$

Applying the above result to our case, we obtain

$$
A_{PNL} = {}^{r_1^2}\!/_{2}(\theta_1 - \sin\theta_1) + {}^{r_2^2}\!/_{2}(\theta_2 - \sin\theta_2) \tag{6.2}
$$

As $r_1 = r_2 = r$ and $\theta_1 = \theta_2 = \theta$, so we get

$$
A_{PNL_{normal}} = r^2(\theta - \sin\theta) \tag{6.3}
$$

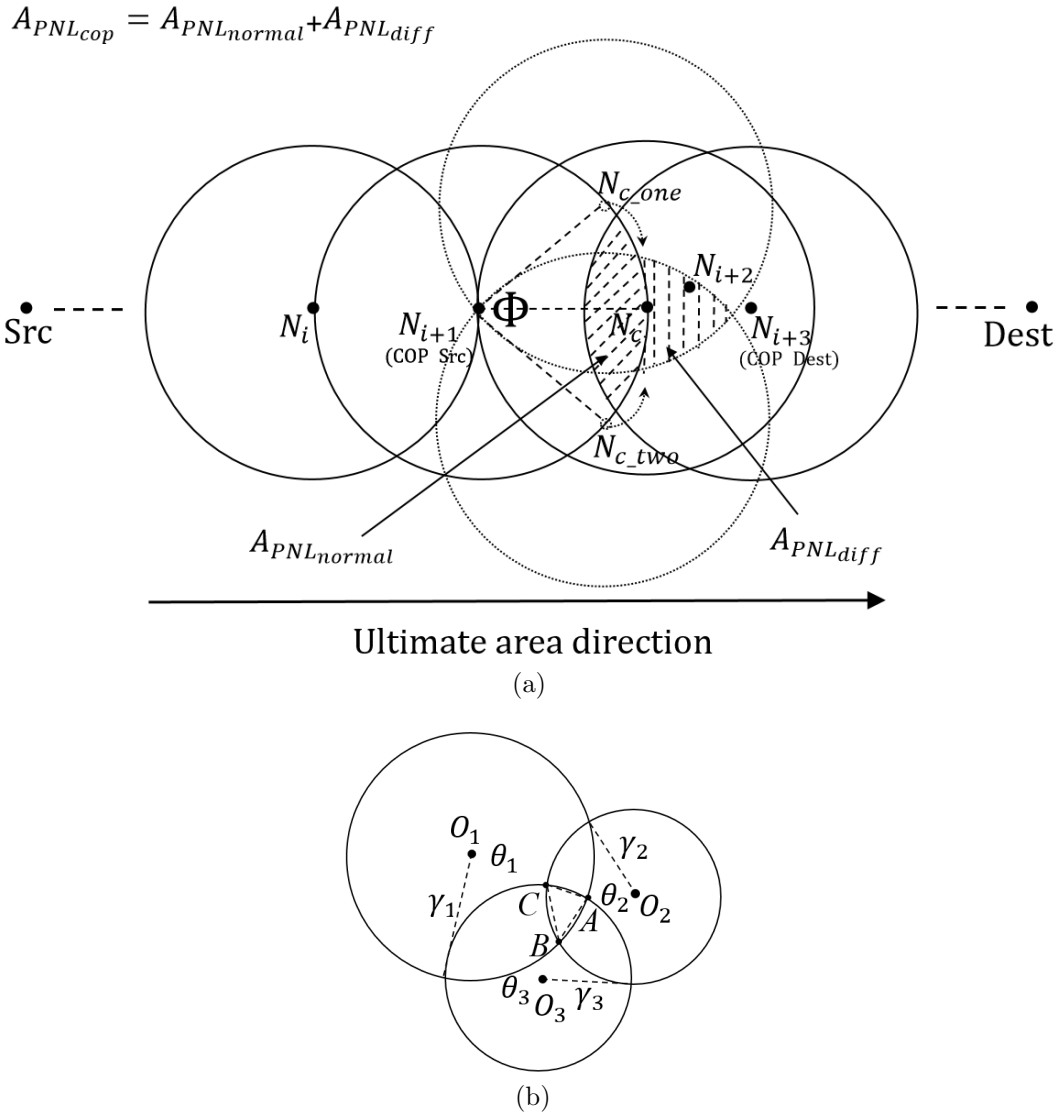


Figure 6.2: PNL in a COP-topology-recognized MANET

The PNL for a COP-topology-recognized MANET is shown in (a) of Figure 6.2 and it is the sum of $A_{PNL_{normal}}$ and $A_{PNL_{diff}}$, where $A_{PNL_{diff}}$ can be obtained by using the intersection area of two circles with centers of the node N_{c_one} and node N_{c_two} and subtracting the intersection area of three circles with centers of the node N_{c_one}, node N_{c_two} and node N_{i+1}. Thus the key point to obtain the PNL for the COP-topology-recognized MANET is to calculate the intersection of the three circles (circular triangle) in (b) of Figure 6.2.

We use the same mathematical definition and method as (b) of Figure 6.1, thus the area of circular triangle $A_{ct_{ABC}}$ can be calculated by Equation (6.4):

$$\begin{aligned} A_{ct_{ABC}} = {}& {}^{1}\!/_{4}\sqrt{(\overline{AB}+\overline{BC}+\overline{AC})(\overline{BC}+\overline{AC}-\overline{AB})} \\ & \times\sqrt{(\overline{AB}+\overline{AC}-\overline{BC})(\overline{AB}+\overline{BC}-\overline{AC})} \\ & +\textstyle\sum_{k=1}^{3} r_k^2 \sin^{-1}\frac{c_k}{2r_k} - \frac{c_k}{4}\sqrt{4r_k^2-c_k^2} \end{aligned} \tag{6.4}$$

The angle Φ in (a) of Figure 6.2 will change from 0 to π, so we can calculate PNL mathematically as Equation (6.5), (6.6), (6.7):

$$A_{PNL_{cop}} = \left\{ \begin{array}{cc} A_{PNL_{normal}} + A_{diff}, & 0 < \Phi < \frac{2}{3}\pi \\ A_{PNL_{normal}}, & \frac{2}{3}\pi < \Phi < \pi \end{array} \right\} \tag{6.5}$$

$$A_{PNL_{normal}} = r^2\left(\frac{2\pi}{3} - \sin\frac{2\pi}{3}\right) \tag{6.6}$$

$$\begin{aligned} A_{PNL_{diff}} = {}& r^2\left[(\pi-\Phi) - \sin(\pi-\Phi)\right] \\ & -\tfrac{1}{4}\times 2r\times\sin(\tfrac{\pi}{3}-\tfrac{\Phi}{2})\times\sqrt{4r^2-\left[2r\times\sin(\tfrac{\pi}{3}-\tfrac{\Phi}{2})\right]^2} \\ & -2\times\left(r^2\sin^{-1}\tfrac{1}{2} - \tfrac{r}{4}\sqrt{3r^2}\right) - r^2\sin^{-1}\tfrac{2r\times\sin(\frac{\pi}{3}-\frac{\Phi}{2})}{2r} \\ & +\tfrac{2r\times\sin(\frac{\pi}{3}-\frac{\Phi}{2})}{4}\sqrt{4r^2-\left[2r\times\sin(\tfrac{\pi}{3}-\tfrac{\Phi}{2})\right]^2} \end{aligned} \tag{6.7}$$

In order to illustrate the difference of the PNL in a normal MANET and a COP-topology-recognized MANET, we show the PNL area changes between these two types of MANET in Figure 6.3. The x-axis in Figure 6.3 is defined as the angle Φ which is at the vertex N_{i+1} enclosed by $\overline{N_{i+1,}N_{C\_ONE}}$ and $\overline{N_{i+1,}N_{C\_TWO}}$ in

(a) of Figure 6.2; the y-axis is defined as the area of the PNL. We simulate three different transmission ranges for each MANET: 50m, 80m and 100m.

As we can see, for a given transmission range, the smaller the angle Φ is, the larger the PNL of a COP-topology-recognized MANET will be. This means a COP-topology-recognized MANET can support more mobility than a normal MANET. Furthermore, as the transmission range becomes larger, the advantage of supporting mobility in a COP-topology-recognized MANET becomes greater. We illustrate this further in the following sections by comparing the performance of AODV, DSR and CRCPR.

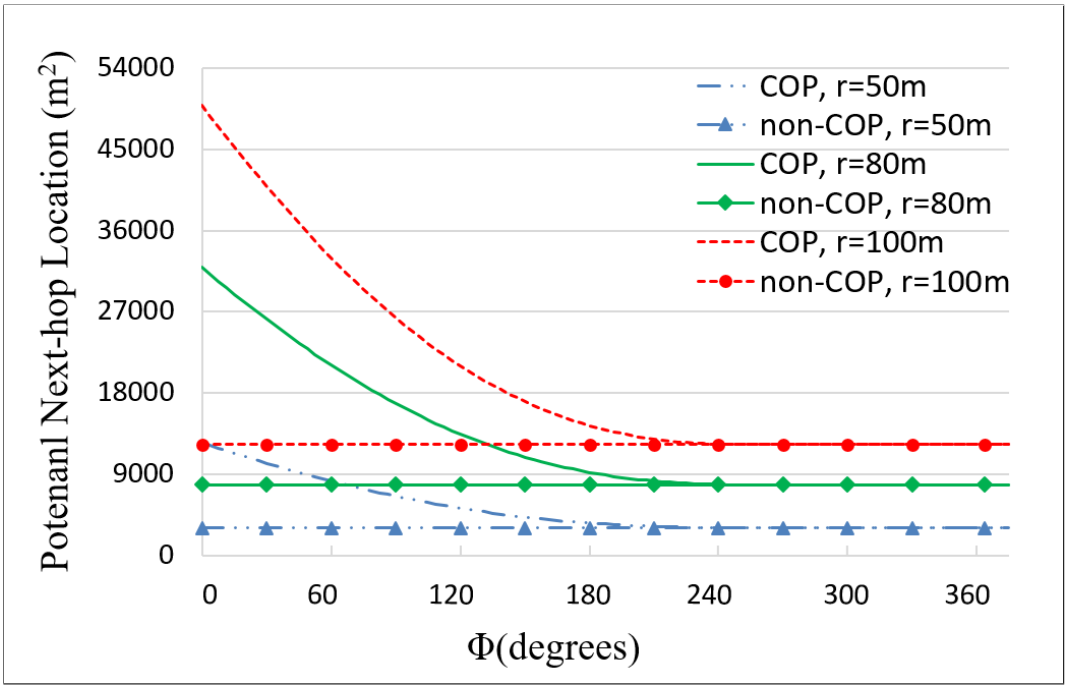


Figure 6.3: PNL Comparison in a Normal MANET and a COP-topology-recognized MANET

6.2 CRCPR versus AODV and DSR

Similar to the work [101], where the authors compare cooperative and non-cooperative routing protocols to evaluate the performance, two classic non-cooperative routing protocols, AODV[40] and DSR [41], are chosen here to compare with CRCPR. As with [88], AODV is chosen as one of our baselines because AODV is widely adopted and its operation is well understood by the research community. Furthermore, DSR is selected as the other baseline as DSR caches back-up routes against link breaks due to mobility, which is similar to CRCPR in terms of route

robustness.

6.2.1 Scenario

Employing an experimental setup similar to [101], a simulation environment is configured as an area of size 1000 meters by 1000 meters. In order to estimate the link break probability we employ the same Random Walk Mobility Model [1] as proposed in [164]. The random trajectories are recorded for each node providing repeatability to ensure comparisons are fair. As a typical MANET comprises less than 100 nodes [180] [181] [182] [183] [184], two network sizes are considered: a 25-node and 50-node case. In the 25-node scenario, we have one pair transmitting with the number of random mobile nodes increasing from one to five. In the 50-node scenario, two pairs of transmitting nodes are set up with the number of random mobile nodes increasing from one to eight. The speed for each mobile node in both scenarios is uniformly distributed [0,2] (m/s), which is the same configuration as [164]. Each scenario runs for 20 minutes simulation time with 10 random seeds to avoid the influence of correlation effects. Figure 6.4 gives one example scenario for 50 nodes with eight mobile nodes. The source nodes are *N_2*, *N_3* and the destination nodes are *N_11*, *N_10*. The differently colored-label nodes with *Italic* node names represent the mobile nodes and each mobile node is randomly chosen to move within its corresponding rectangular region which is randomly decided as well.

[1] Referred to in Section 2.2.0.4, the mobile nodes in Random Walk Mobility Model move from their current location to a new location according to a randomly selected velocity.

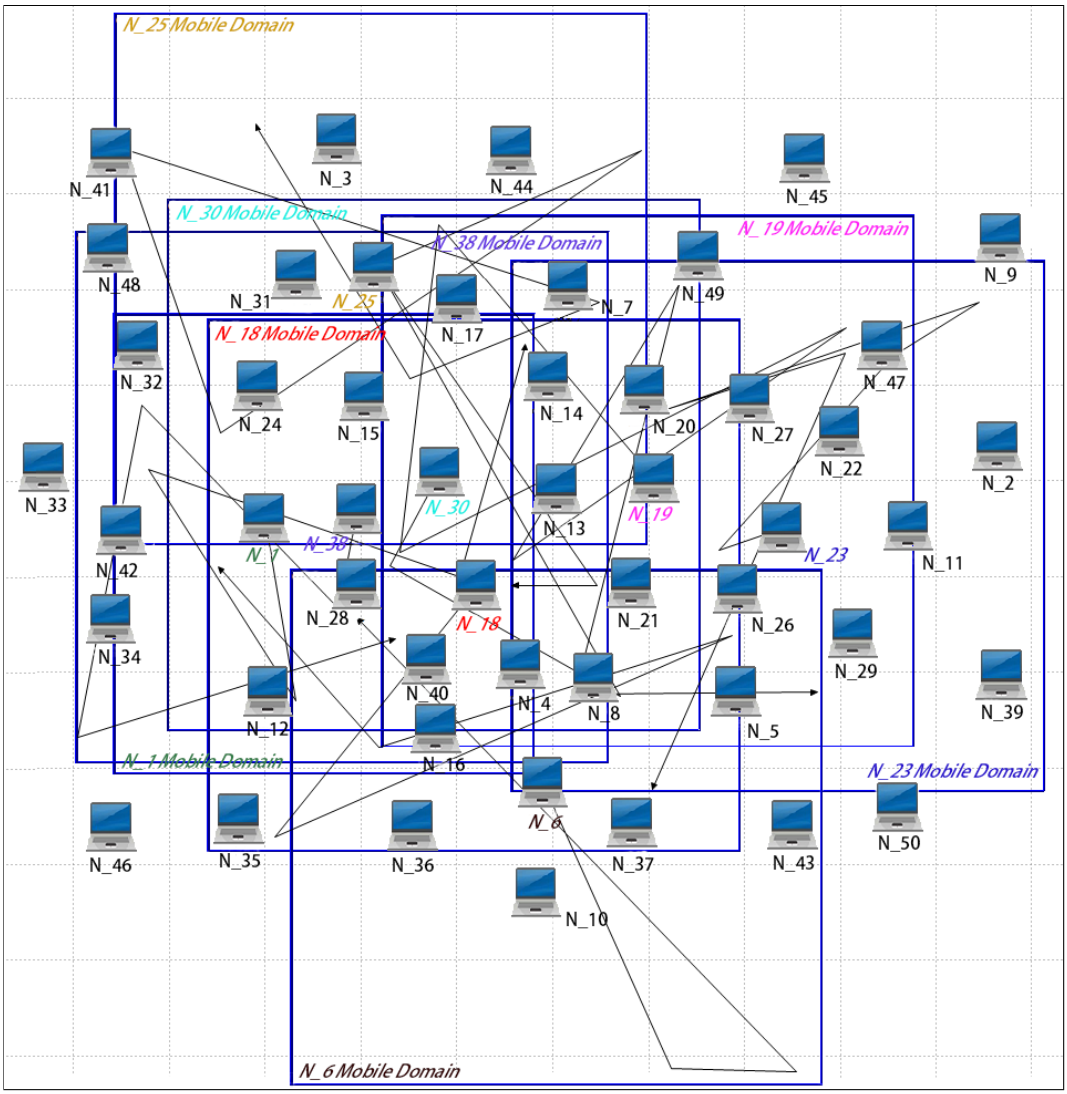
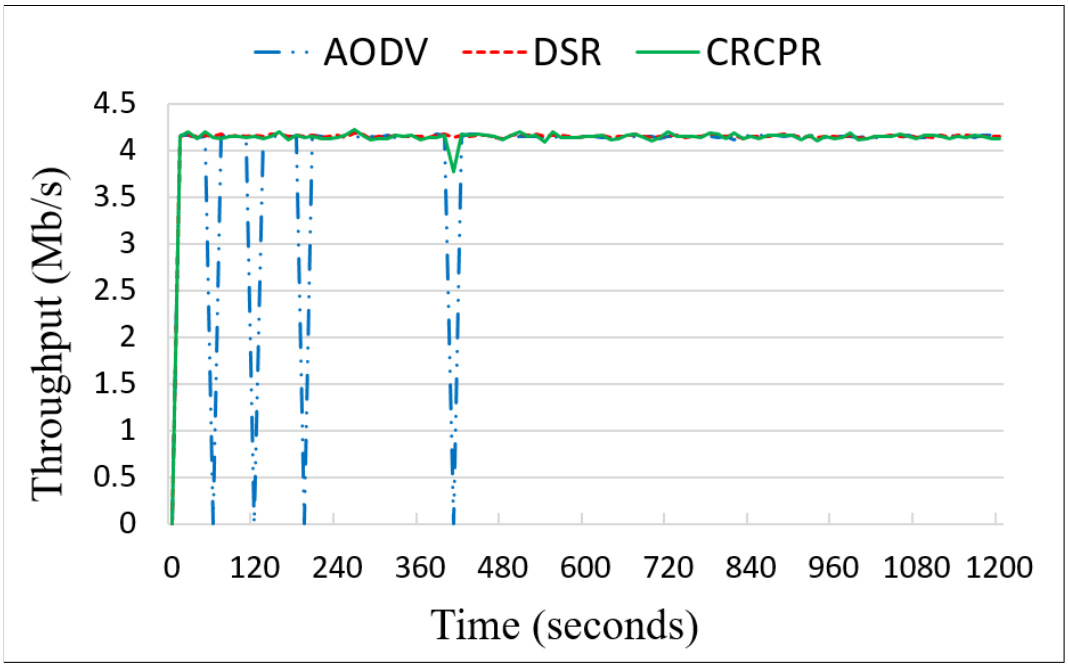


Figure 6.4: Example Scenario for 50 Nodes

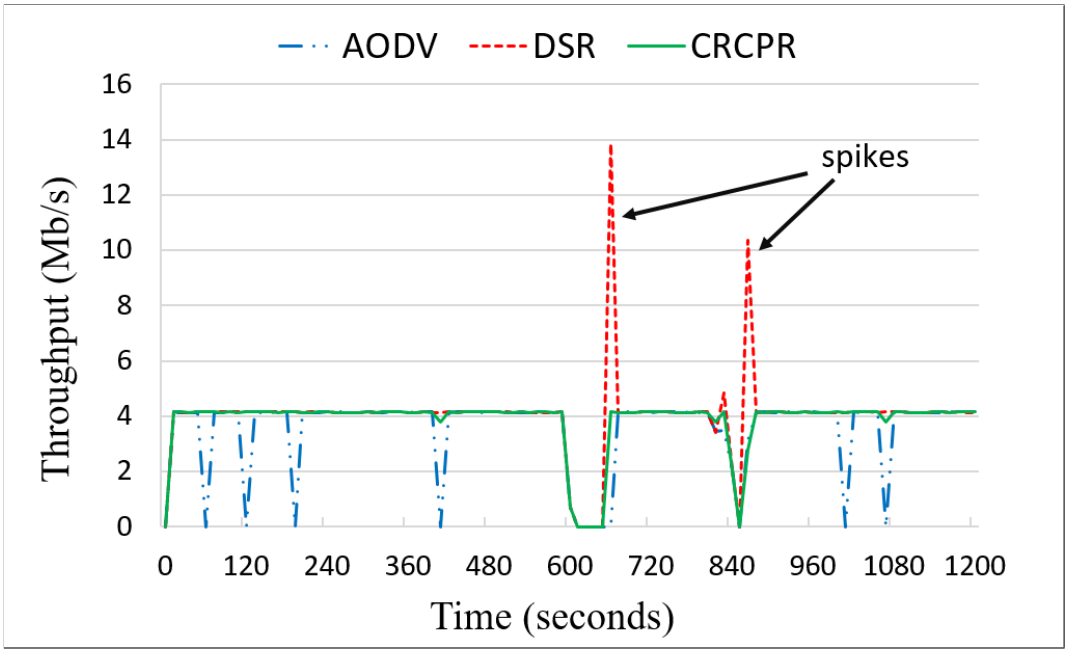
6.2.2 Results

6.2.2.1 Throughput

All the throughput data shows instantaneous values, which clearly illustrates the link breaks. Both (a) and (b) in Figure 6.5 show the throughput at the destination node with four and five mobile nodes in the 25-node scenario.



(a) 4 Mobile Nodes



(b) 5 Mobile Nodes

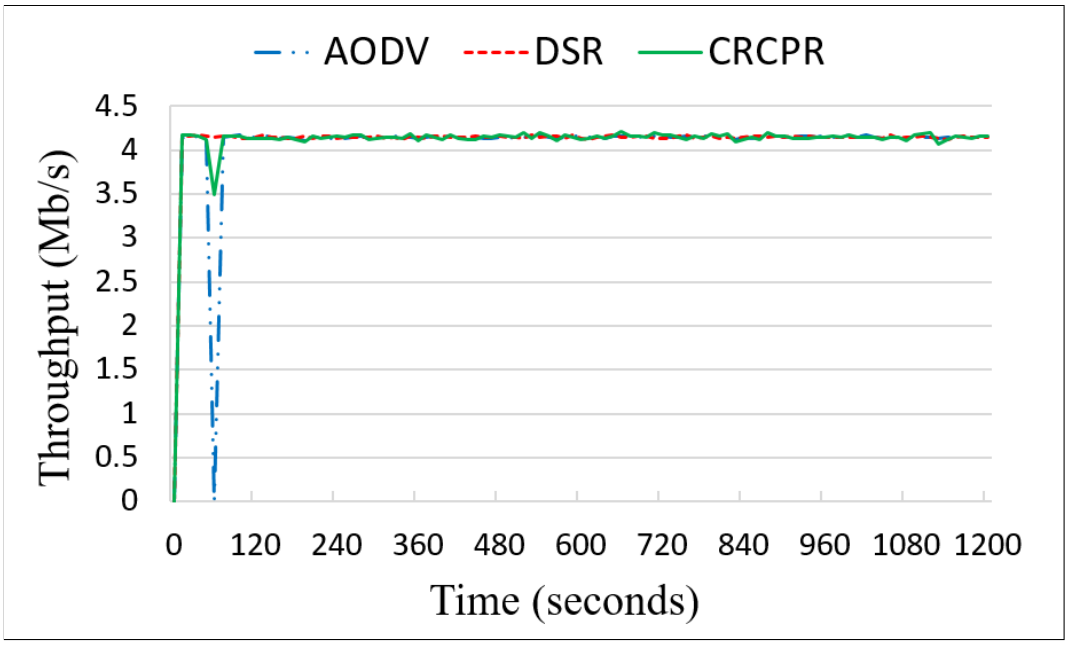
Figure 6.5: Typical Throughput in the 25-node Scenario with 4 / 5 Mobile Nodes

In the 25-node scenario, when there are 4 mobile nodes, as with (a) in Figure 6.5, AODV suffers link breaks, shown as sudden drops in throughput, and has to re-establish a route for transmission by broadcasting CREQ packets. As DSR has cached routes, it just changes to the back-up route and does not need to conduct route discovery again. CRCPR uses the route enhancement relay feature (where possible), so the movement only changes the transmission from the COP mode

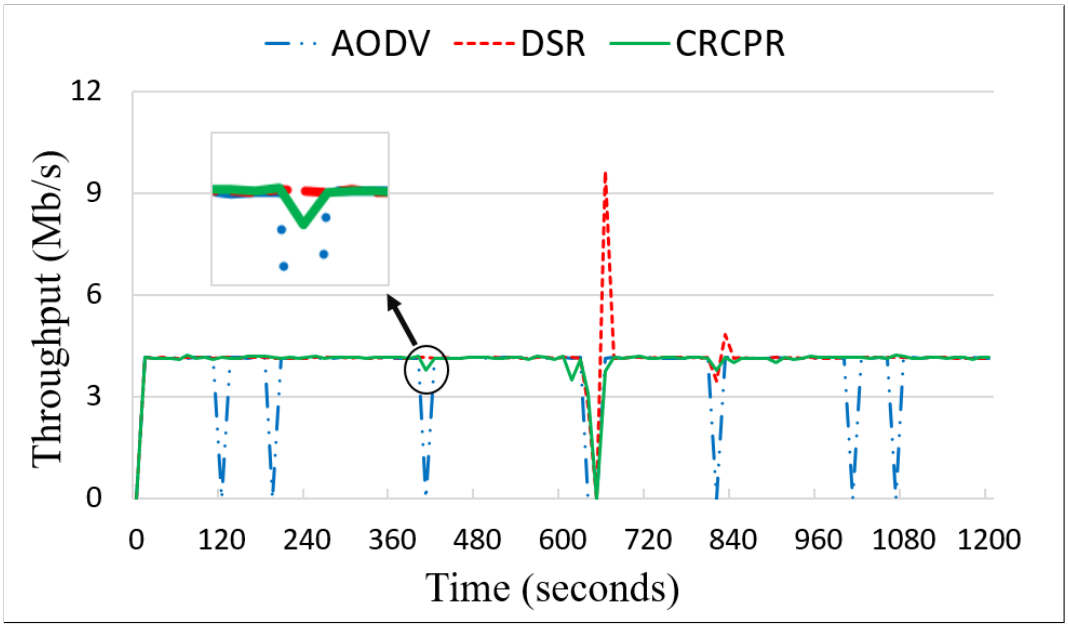
to Relay mode without suffering any link breaks when the first three link breaks happen in this instance.

Once the number of mobile nodes increases to five, the link breaks happen more frequently, as seen in (b) of Figure 6.5. During the interval between 600s and 700s, all three protocols lose the route. Under the DSR scheme, once the route is reconstructed, all buffered messages in the Network Layer are sent leading to the spikes in throughput. For AODV and CRCPR, the Network Layer has no data back-up scheme and just discards the packets when no route exists. Overall, CRCPR performs much better than AODV in terms of link break outages and throughput.

Figure 6.6 shows the throughput with five and eight mobiles in the 50-node scenario. As the number of mobile nodes increases from 5 to 8, link breaks happen more frequently as shown in (a) to (b) in Figure 6.6. Meanwhile, there is a small drop in throughput with CRCPR, which is emphasized in the enlarged area. This drop indicates a link break similar to AODV, but because we employ a local repair mechanism, as in ABR [44] described in Section 3.3.8, CRCPR can restore the route more quickly and maintain high throughput. Therefore, CRCPR has much better performance than AODV in terms of reduced link breaks and route recovery. With more random mobile nodes, DSR with its many cached routes performs well but requires considerable resource to maintain the back-up routes.



(a) 5 Mobile Nodes



(b) 8 Mobile Nodes

Figure 6.6: Typical Throughput in the 50-node Scenario with 5 / 8 Mobile Nodes

6.2.2.2 Number of Link Breaks

For the 25-node scenario, the link break frequency of the three protocols is shown in Figure 6.7. In this case,when the number of random mobile nodes increases from 1 to 3, CRCPR has the same performance as DSR. With 4 and 5 random mobile nodes, DSR has the best performance due to its cached routes. However, the frequency of link breaks for CRCPR remains at least 40% lower than AODV.

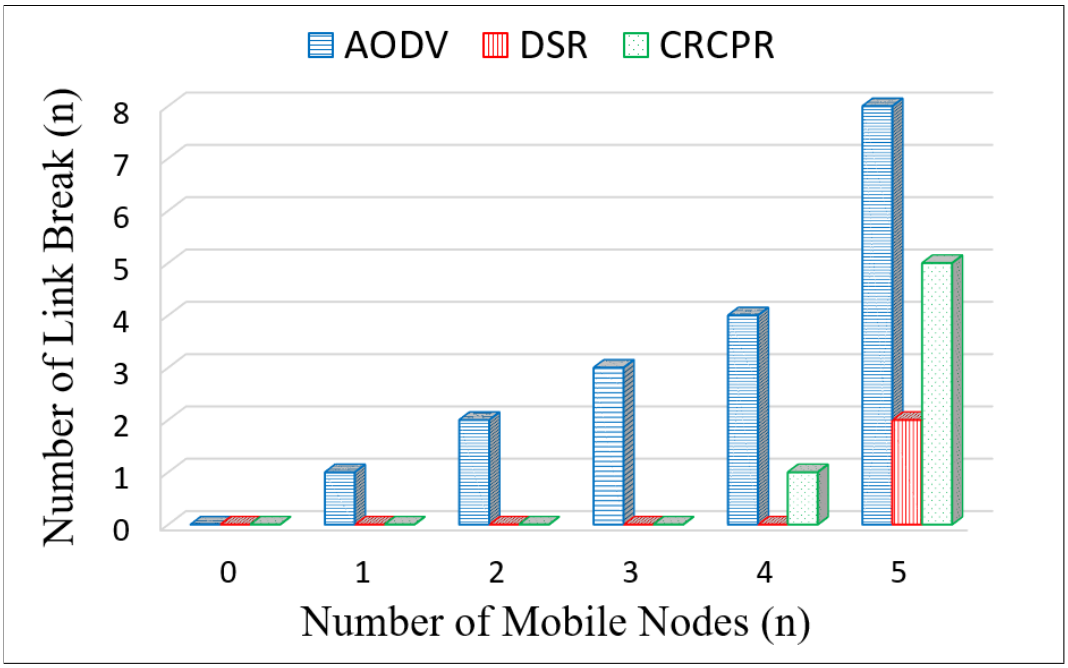


Figure 6.7: Link Break Frequency in the 25-node Scenario

For the 50-node scenario, Figure 6.8 shows that DSR has the best performance due to its use of cached routes but with high memory cost. In this case, when the number of random mobile nodes increases from 1 to 5, CRCPR has the same performance as AODV. With 6, 7 and 8 random mobile nodes, the frequency of link breaks for CRCPR remains at least 50% lower than AODV. Nevertheless, in both the 25-node and 50-node scenarios, CRCPR greatly outperforms AODV.

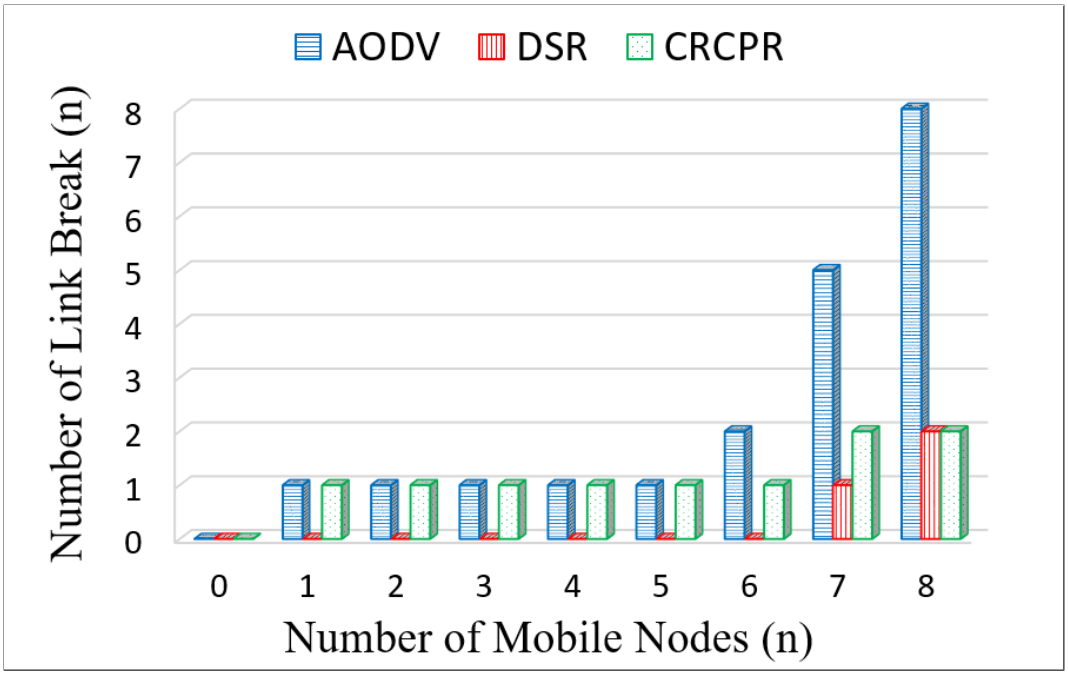


Figure 6.8: Link Break Frequency in the 50-node Scenario

6.2.2.3 Power Consumption

Figure 6.9 and Figure 6.10 shows the power consumption for the 25-node and 50-node scenarios, respectively. All the results for power consumption are normalized according to Equation (6.8), where pp is the message *Processing Power* per bit, t_p is the *Transmission Power* per bit, b_{total} is the total bits including the data from the Application Layer, control messages from the Network and MAC Layers, as well as the message header of data message added by the Network and MAC Layers, b_{data} is only the data bits from the Application Layer and con is a selectable scaling coefficient, which can guarantee the normalized power in the range [0,1].

$$P_n = \frac{2}{\pi} \times \arctan\left[con \times e^{\left(\frac{t_p \times b_{total} + pp \times b_{total}}{t_p \times b_{data} + pp \times b_{data}}\right)}\right] \tag{6.8}$$

In the 25-node case, CRCPR has the best performance whilst DSR has the worst. The reason is that once a COP topology is selected for a route or a COP topology is formed locally during the transmission, more than 40% power will be saved relative to the non-cooperative transmission case according to [110]. DSR consumes more energy due to the retransmission mechanism.

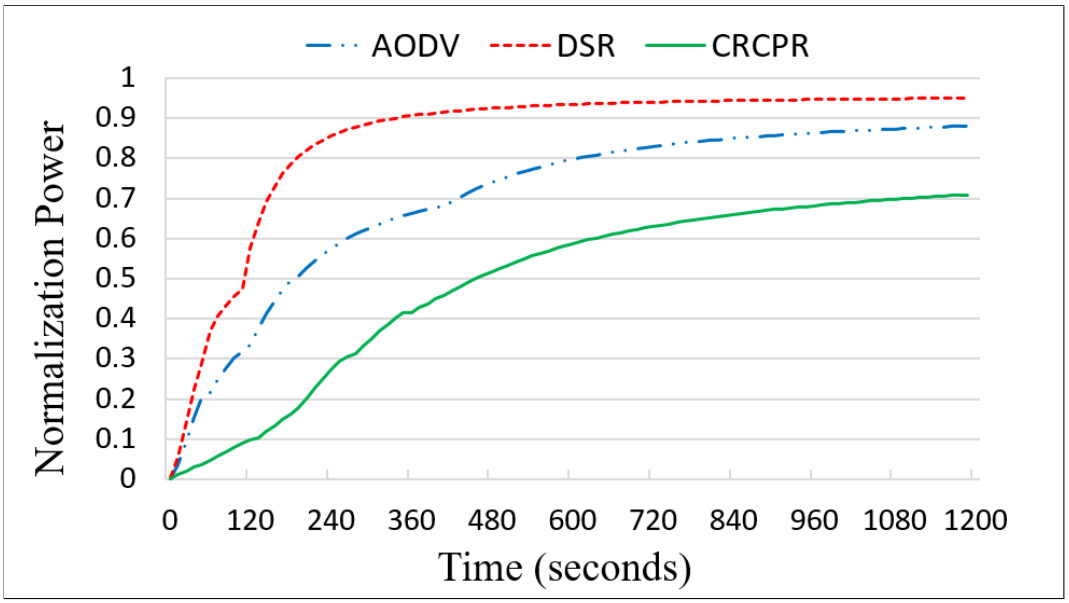


Figure 6.9: Power Consumption in the 25-node Scenario

For the 50-node scenario, with more nodes in the area, there are more opportunities for COP topology-based route establishment. Therefore, more energy can be saved by cooperative communication.

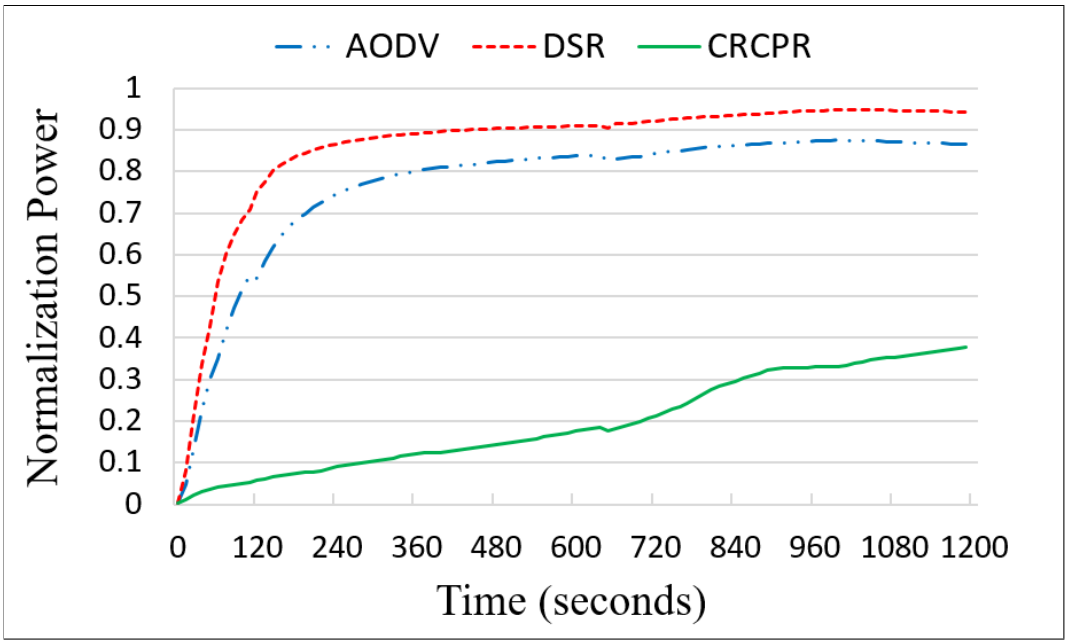


Figure 6.10: Power Consumption in the 50-node Scenario

6.3 CRCPR versus CWR

CRCPR utilizes several table structures, namely: the Cooperative Neighbour Table, the COP Table and the Relay Table to reduce control overhead and enhance resilience to mobility; the "Recruit-and-Transmit" scheme is adopted in CWR [131] for the same purpose. In order to investigate the mobility handling of CRCPR and CWR, we analyze the performance from two aspects: throughput and the number of link breaks. As described in Section 3.3.9, CRCPR employs a new route selection algorithm which can utilize the Routing Matrix Values (RMV) to estimate the energy and link break probability at the same time when it chooses a route. In order to analyze the performance of the route selection scheme in CRCPR, network lifetime is considered.

6.3.1 Scenario

Experiments are set to compare mobility handling and energy consumption of CRCPR and CWR. In order to make a fair comparison, we use the scenario adopted for CWR in work [131]. In our work, we scale down the network from 7 rows × 21 columns of nodes to 3 rows × 7 columns as shown in Figure 6.11, which is efficient for mobility handling investigations. The straight-line distance between two adjacent fixed nodes is 20m and transmission range of each node is 30m. *N_1*

as the source node is located in the first column of the middle row and *N_24* as the destination node is located in the same row. The number of mobile nodes (with an *Italic* node name in Figure 6.11) in the simulation scenario increases from 4 to 8 across five simulations. In each simulation, we run 30 trials for a given number of mobile nodes. The mobile nodes are deployed randomly. The speed of each mobile node is uniformly distributed [0,2] (m/s) and the movement direction is along the arrow in Figure 6.11. The simulation duration is set to 20 minutes.

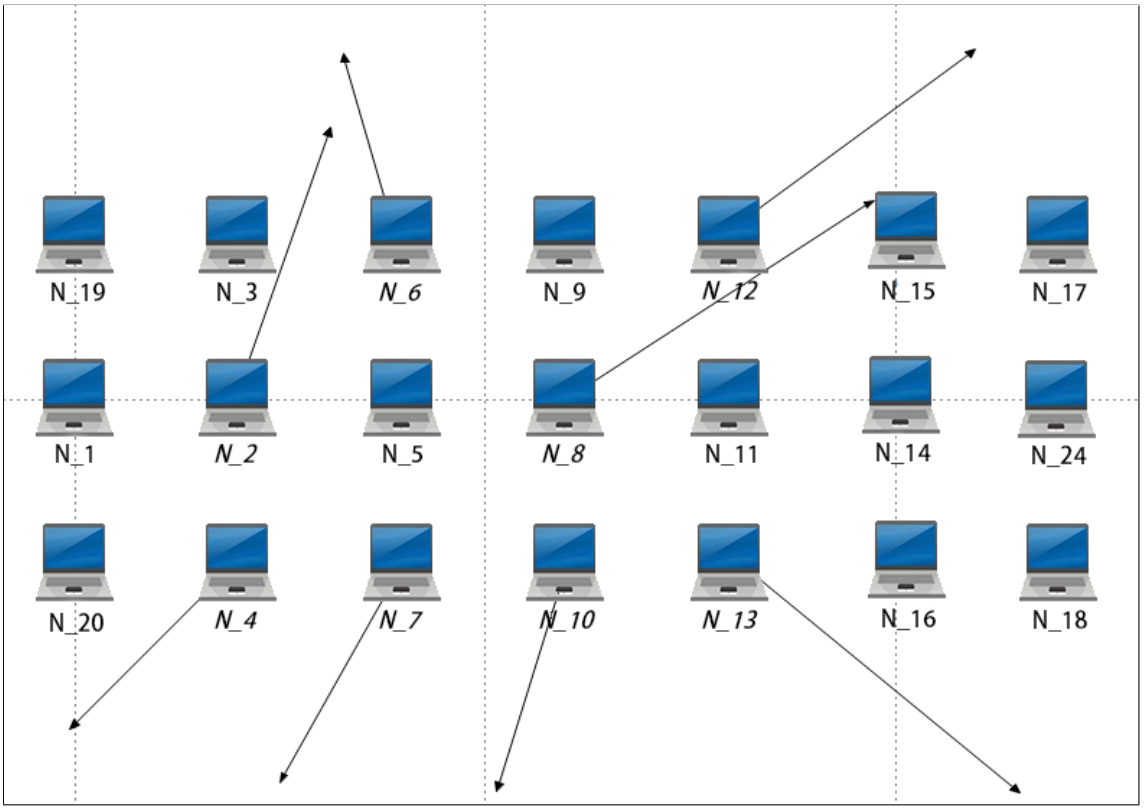
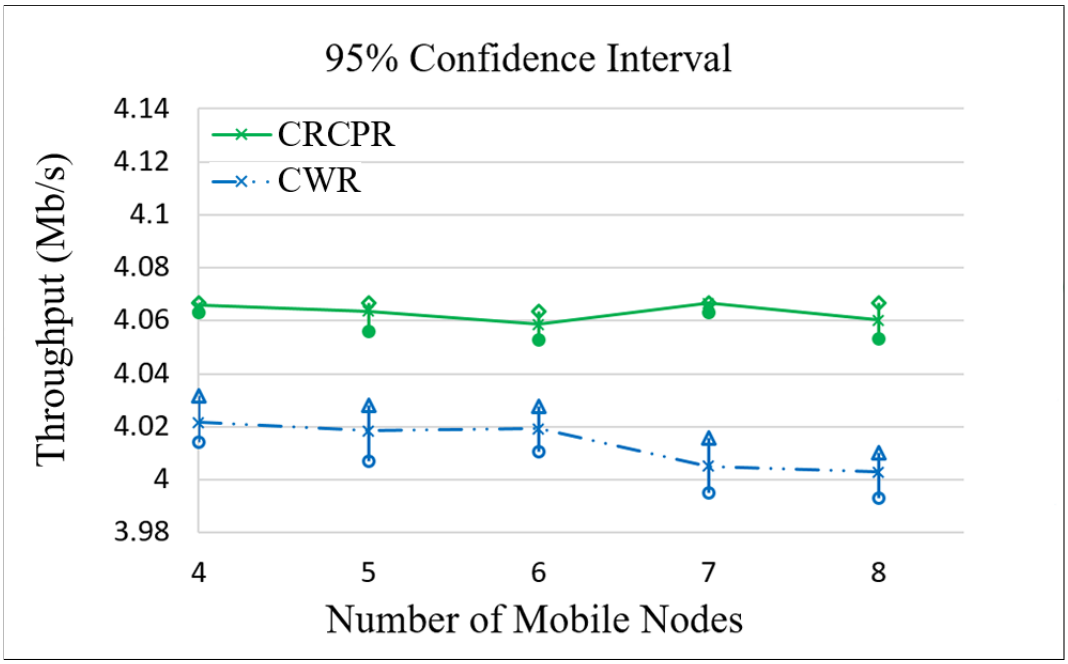


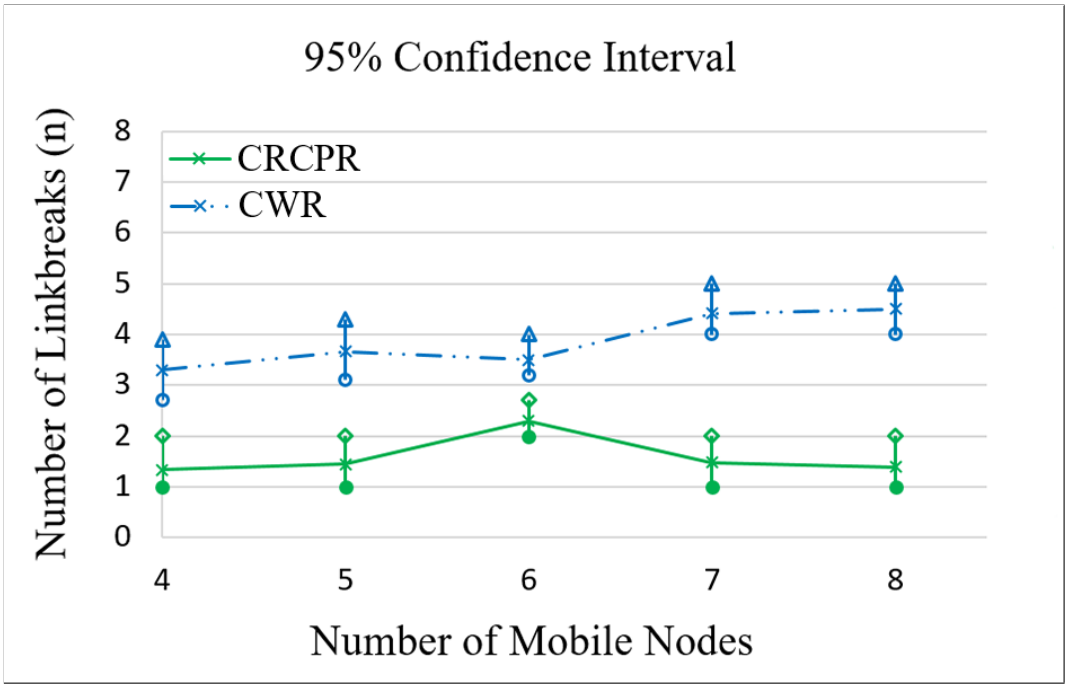
Figure 6.11: Mobility Handling Scenario with 8 Mobile Nodes

6.3.2 Results

6.3.2.1 Mobility Handling



(a) Throughput



(b) Number of Link Breaks

Figure 6.12: Resilience to Mobility of CWR and CRCPR

We provide the 95% confidence intervals when showing the results. From (a) of Figure 6.12, we can see the throughput of CWR decreases with increasing

number of mobile nodes, whereas the value of throughput for CRCPR remains more stable. The better performance of CRCPR is because it can utilize the COP topology to improve the robustness against node mobility while CWR has no mobility awareness. If one cooperative (C) node in the COP topology happens to be a mobile node, it does not lead to a link break as explained in Section 3.3.6. Furthermore, the route selection criteria of CRCPR tries not to select a node with high link break probability, so a more stable route will be selected in CRCPR than the "shortest path" route selected by CWR. On the other hand, the number of link breaks also reflects the mobility handling effectiveness of CRCPR, which is shown in (b) of Figure 6.12. As the number of mobile nodes increases, the number of link breaks of CRCPR remains at least 50% lower than for CWR, which results from the route recovery process being invoked less frequently and improves the network throughput.

6.3.2.2 Network Lifetime

Network lifetime is defined as the network duration when the first node in the network experiences energy drain out. For CWR, none of the nodes possess a Energy Harvesting (EH) capability. Therefore, in order to make the fair comparison, we deactivate the EH function of CRCPR. The same simulation configuration used for assessing the mobility handling is implemented to investigate the route selection performance of CWR and CRCPR. More precisely, instead of deploying mobile nodes, we now deploy Energy Restricted (ER) nodes, which have lower energy than the other nodes. All the ER nodes are named with a red *Italic* font and selected randomly. The number of ER nodes increases from 1 to 6 in six simulations. Figure 6.13 shows the EH-disabled scenario with 6 ER nodes. In each simulation, we run 30 trials for a given number of ER nodes.

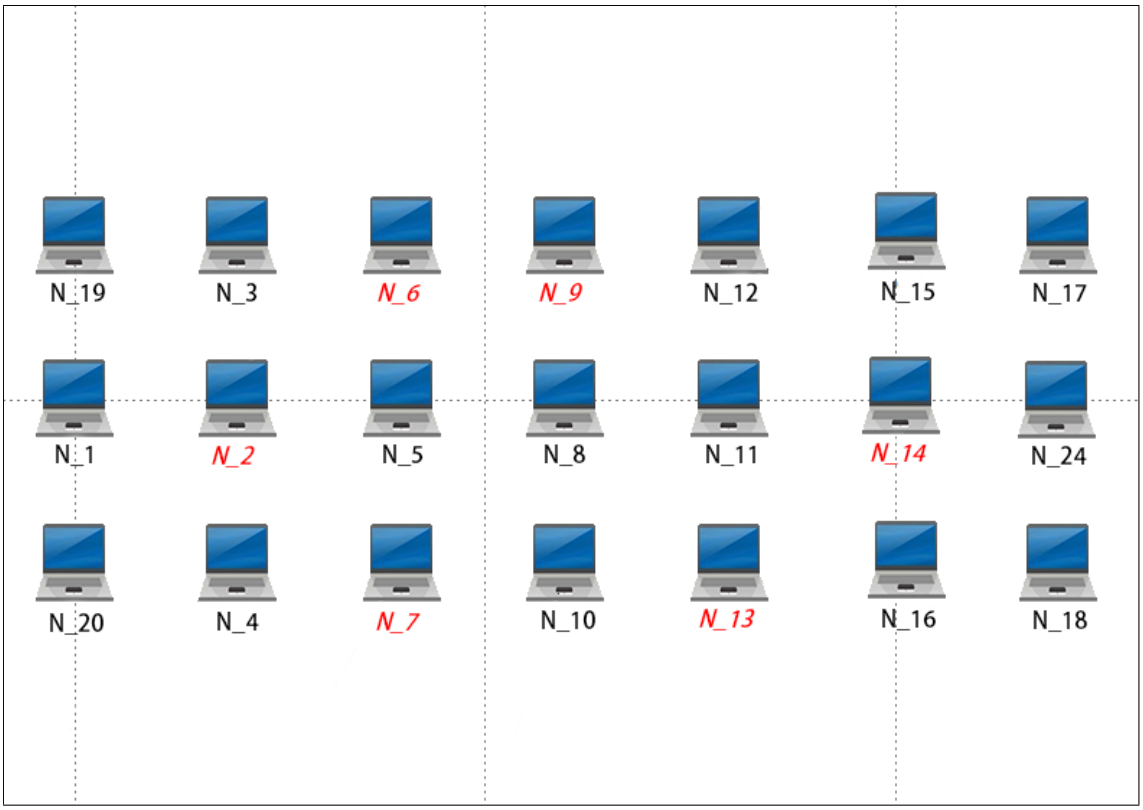


Figure 6.13: EH-disabled Scenario with 6 ER Nodes

In Figure 6.14, with increasing ER nodes, network lifetime of both CWR and CRCPR deceases. However, when the number of ER nodes is lower than 3, the network lifetime performance of CRCPR remains stable. The reason is the route selection scheme in CRCPR avoids choosing ER nodes to form the final route, where possible, and leads to at least 40% higher network lifetime performance. When the number of ER nodes becomes greater, i.e. 6 ER nodes in this scenario, CWR and CRCPR exhibit similar network lifetime. This is because when more ER nodes are present in the scenario, it becomes much harder for the CRCPR route selection scheme to find a route without ER nodes. Therefore, both CWR and CRCPR show the similar lifetime performance.

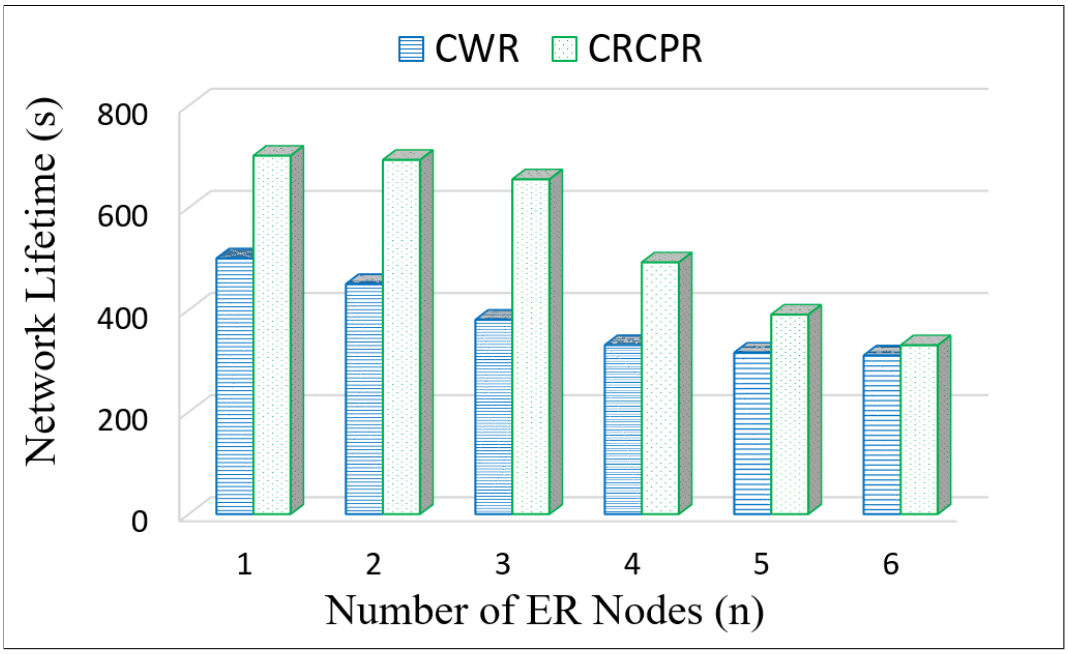


Figure 6.14: Network Lifetime without EH

6.4 CRCPR versus DEHAR and AODV-EHA

To investigate economic energy consumption feature, we employ two other routing protocols that take into account the energy harvesting potential of nodes: DEHAR [151] and AODV-EHA [152]. DEHAR introduces a concept called "energy distance" which is encoded from spatial distance and makes the real distance related to the energy status (how much energy can be harvested from the surroundings). The route with the shortest energy distance will be selected as the final route. AODV-EHA considers the energy harvesting ability of all nodes and tries to find the route with least transmission cost by replacing "hop count" with "energy count". Energy count can be obtained by predicting the average transmission cost to forward a data message successfully from the sending node to the receiving node.

6.4.1 Scenario

The scenario is set up 21 nodes arranged into 3 rows × 7 columns as shown in Figure 6.15. The source node and destination node are *N_1* and *N_24*, respectively. All the nodes are static. The energy restricted (ER) nodes are named with a green *Italic* font and selected randomly. The number of ER nodes increases from 1 to 6

across six simulations. In each simulation, we run 30 trials for a given number of ER nodes. This scenario does not only reflect the different performance of CRCPR, DEHAR and AODV-EHA in terms of network lifetime, but also confirms the economic energy consumption feature of cooperative communication in CRCPR.

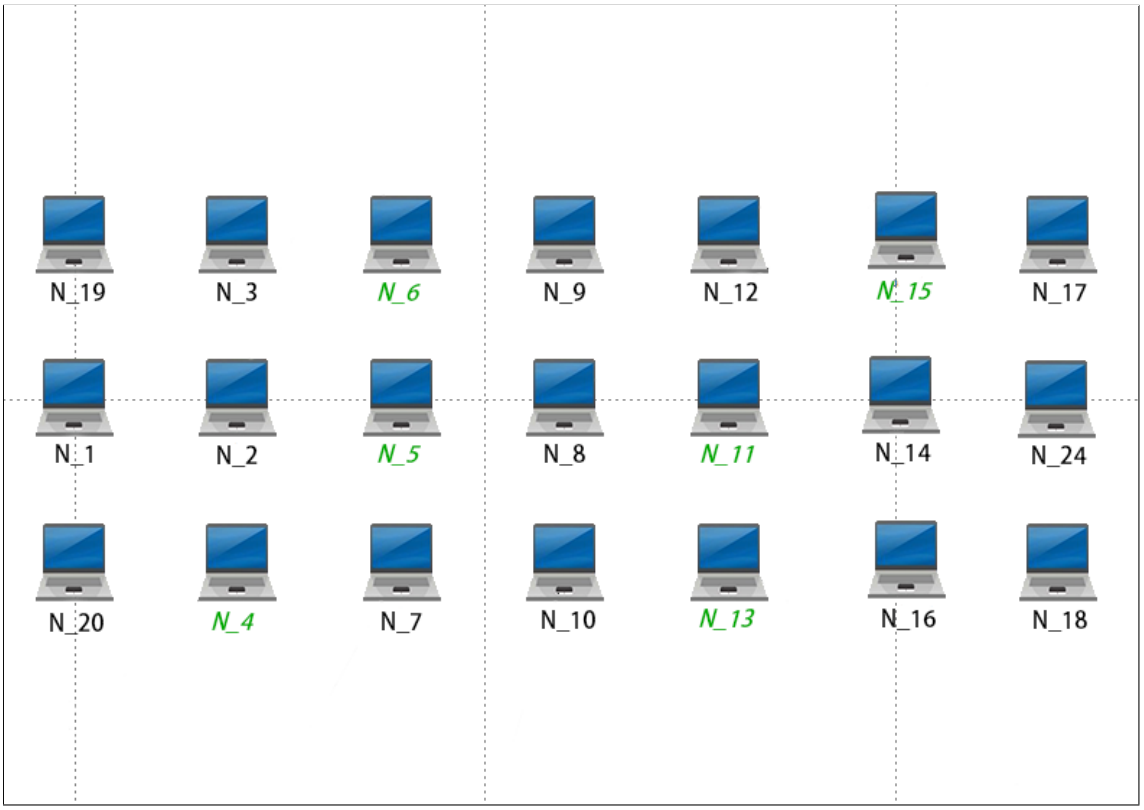


Figure 6.15: EH-eabled Scenario with 6 EH Nodes

6.4.2 Results

6.4.2.1 Network Lifetime

As shown in Figure 6.14 in Section 6.3.2.2, the highest network lifetime level is around 700 seconds if the energy harvesting feature is not activated while as shown in Figure 6.16 below, the highest network lifetime level can reach 1800 seconds if the energy harvesting feature is activated.

With increasing number of ER nodes, the lifetime of all three protocols becomes shorter but CRCPR has the best performance at all times. When the number of ER nodes is less than 3, CRCPR's performance remains stable due to its route selection mechnism which avoids choosing energy-restricted nodes along the final route. In this scenario, when the number of ER nodes is greater than 3, although

it is difficult for CRCPR to avoid ER nodes along the final route completely, the lifetime is still 80% longer than for the other two energy harvesting protocols. The reason is that CRCPR can utilize cooperative diversity to save energy during transmissions.

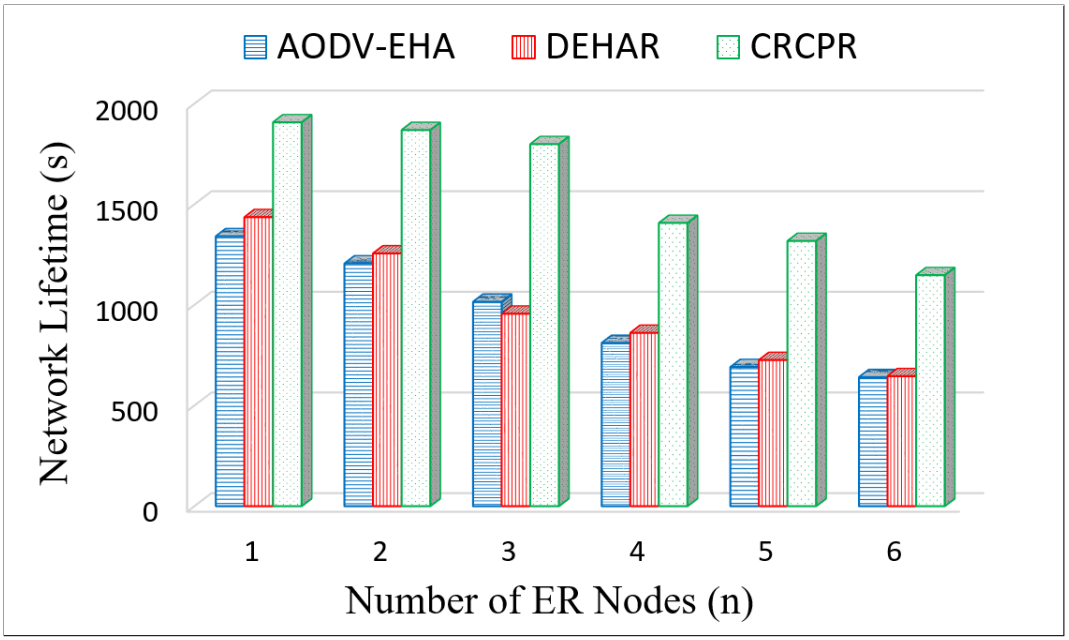


Figure 6.16: Network Lifetime with EH

6.5 ACHS versus PHMS and RHMS under AODV

6.5.1 Static Scenario

Figure 6.17 shows a scenario with static nodes to evaluate the proposed ACHS (Adjust Classified Hello Scheme) in this work compared to another two hello schemes: PHMS (Periodic Hello Message Scheme) and RHMS (Reactive Hello Message Scheme), while employing AODV as a Network Layer routing protocol. In this scenario, data is routed from N_1 to N_2. The transmission range of each node is 30m. The arrows show the route established by AODV. Other key parameters are shown in Table 6.1. These parameters are set as suggested in [40]. The number of nodes in the scenario increases from 20, 25 to 30 in order to analysis average end-to-end delay and hello efficiency.

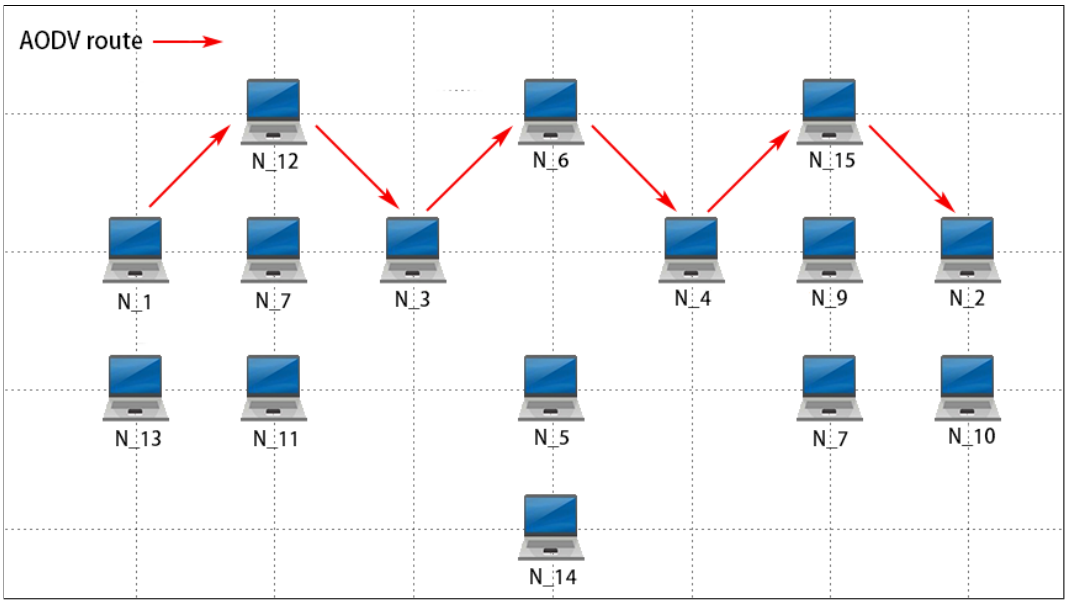


Figure 6.17: Static Scenario

Table 6.1: ACHS Parameters for Stable Scenario

Date Interval	1s
PHMS Hello Interval	1s
RHMS Hello Interval	1s
ALLOW HELLO LOSS	2

6.5.2 Results

6.5.2.1 End-to-End Delay

The results of average end-to-end delay in different network scales are shown in Figure 6.18. The performance of ACHS is almost the same as PHMS and obviously lower than for RHMS. The reason is that in RHMS, nodes only start broadcast hello messages when they need to send data so that it will take more time to build the neighbour table before discovering the route. Therefore, ACHS and PHMS have a better performance than RHMS in the terms of the average end-to-end delay.

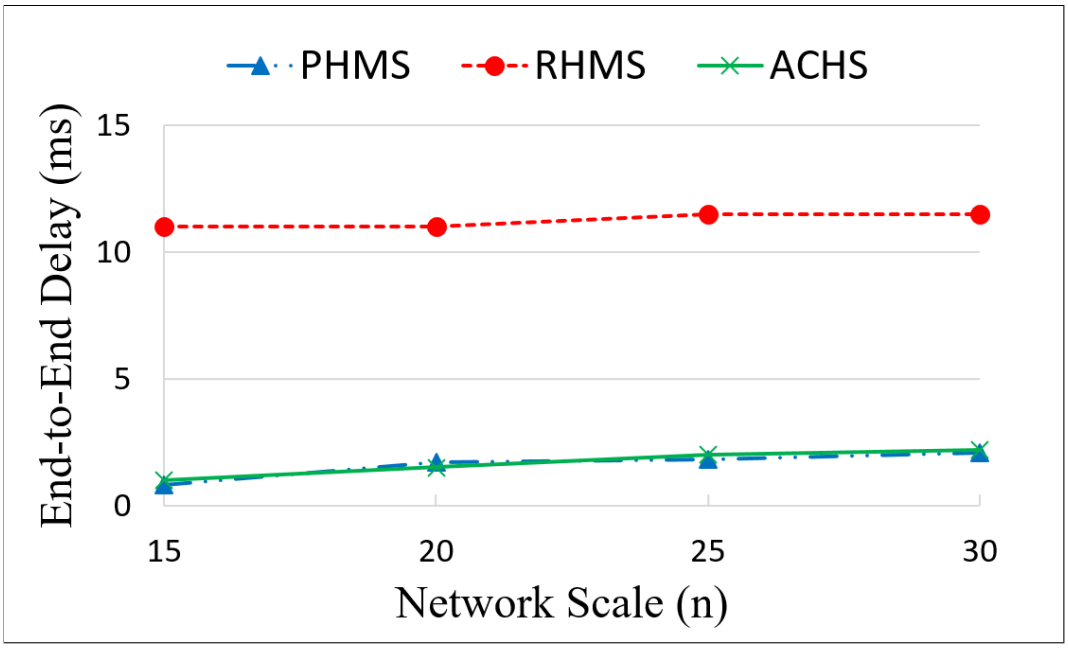


Figure 6.18: End-to-End Delay in Static Scenario

6.5.2.2 Hello Efficiency

Hello efficiency is defined as the ratio of data receiving rate to the hello message broadcasting rate as shown in Equation (6.9), which can be used to evaluate the network performance considering the broadcasting of hello messages and the throughput.

$$Hello\_efficiency = \frac{Throughput}{Hello\_message\_rate} \tag{6.9}$$

In Figure 6.19, as the number of nodes increases, the trend of average hello efficiency for ACHS is decreasing from 0.16 to 0.09 while for PHMS and RHMS, the average hello efficiency is decreasing from 0.07 to 0.04. This is because the source node maintains the same data transmission rate but the number of broadcast hello messages grows. Since RHMS has no hello message transmission before data transmission, its efficiency is typically 10% better than the PHMS. Regarding ACHS, it embeds the content of hello messages into data messages, so the number of transmitted hello messages is the lowest among these three methods and the average hello efficiency is typically 200% higher. Therefore, compared with PHMS and RHMS, ACHS is shown to offer the best performance in terms of average hello efficiency.

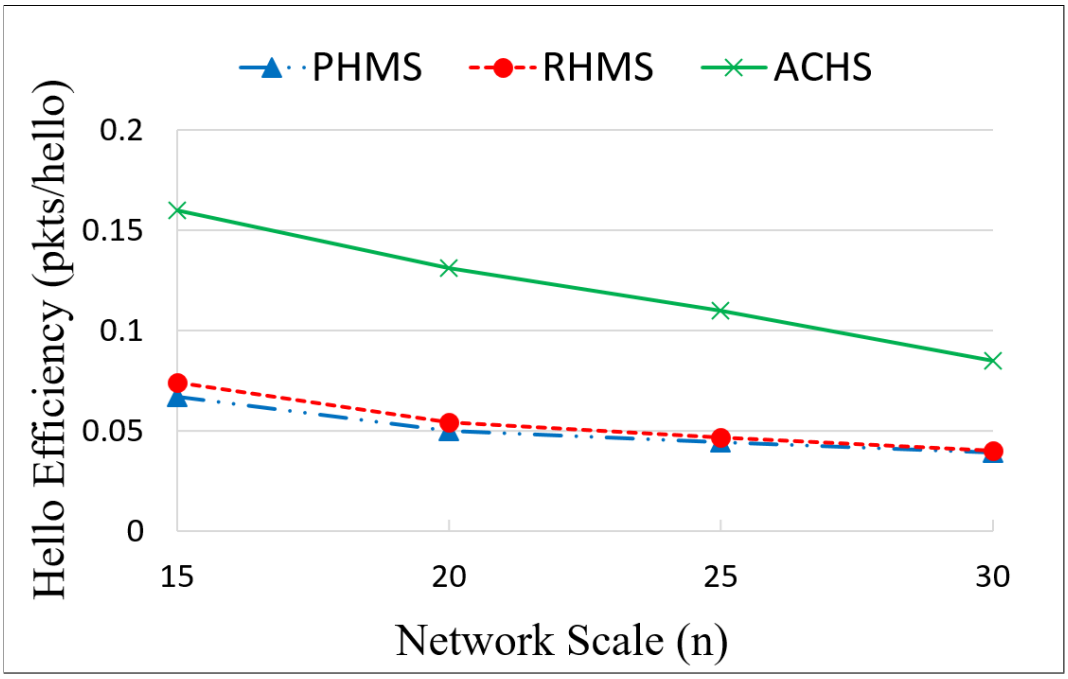


Figure 6.19: Hello Efficiency in Static Scenario

6.5.3 Mobile Scenario

Figure 6.20 shows a mobile scenario with 50 nodes. AODV is simulated to route data from N_1 to N_2. The transmission range of each node is 30m. The pink arrows show the initially established route. The blue arrows indicate the reconstructed route after a link break. All nodes are static except the nodes with the green arrows. The moving nodes are configured to move with constant acceleration of $0.1m/s^2$ at the beginning of the simulation along the direction of their green arrows. For this scenario, we run three simulations with three different initial velocities for the moving nodes: 0.1m/s, 0.3m/s and 0.5m/s. The other parameters are set with the same values as in the static scenario. The end-to-end delay and hello efficiency are analyzed.

In order to perform the classification for the ACHS scheme, the thresholds are set in accordance with Table 6.2:

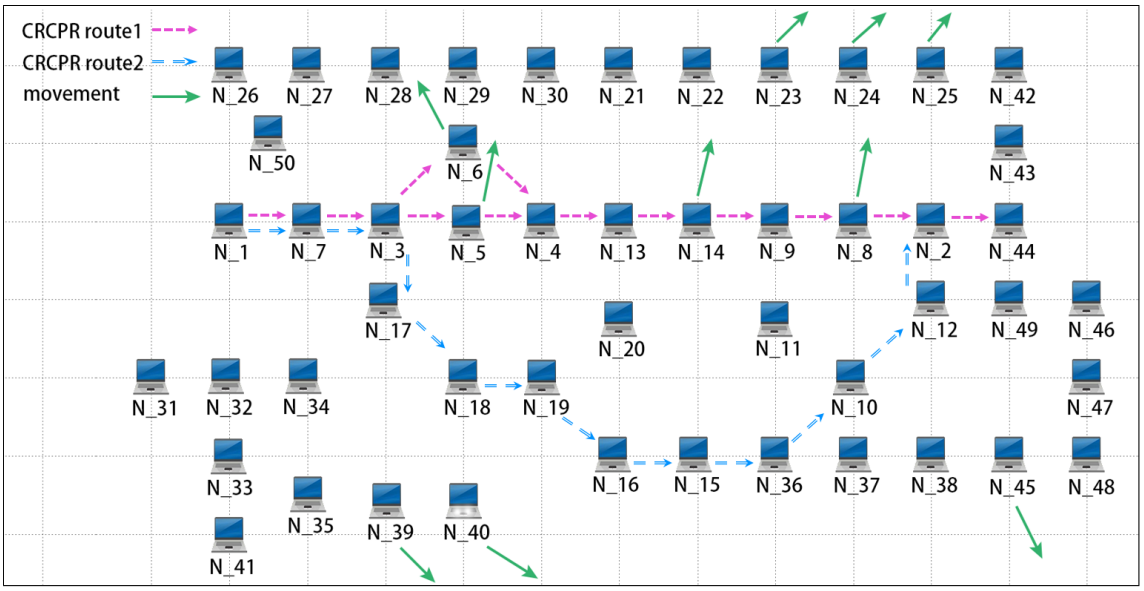


Figure 6.20: Moving Scenario

Table 6.2: ACHS Parameters for Mobile Scenarios

$V_{th1'}$	0.3m/s
$V_{th2'}$	0.8m/s
$h1B-a$	1.5s
$h1B-b$	1.35s
$h1B-c$	1.1s

6.5.4 Results

6.5.4.1 End-to-End Delay

Figure 6.21 indicates the average end-to-end delay for the mobile scenario. The results are similar to the static scenario. The average end-to-end delay of RHMS is the highest due to its on-demand design. So the performance of ACHS and PHMS is around 80% better than RHMS.

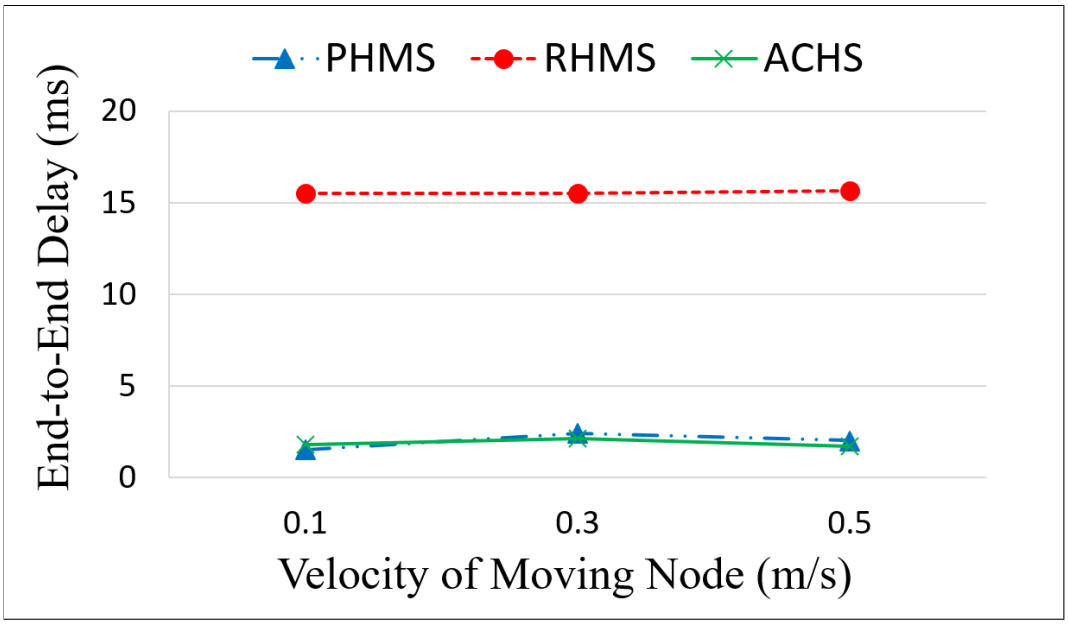


Figure 6.21: End-to-End Delay in the Mobile Scenario

6.5.4.2 Hello Efficiency

Figure 6.22 provides the average hello efficiency results. The reason why PHMS has the worst performance is that the hello broadcasting interval is fixed during the transmission and PHMS does not consider the link situation (link break or not) and node roles (on the route or not) in the network. So the highest number of broadcast hello messages results in the lowest hello efficiency. For RHMS, when a link break happens, the broadcast hello messages will be suspended until the route is recovered. Fewer broadcast hello messages results in a better hello efficiency than PHMS. As with ACHS, some nodes in the scenario will be classified differently resulting in different hello broadcasting intervals, which leads to a lowest volume of broadcast hello messages. Therefore, in terms of average hello efficiency, the performance of ACHS is 80% better compared with PHMS and 20% better compared with RHMS.

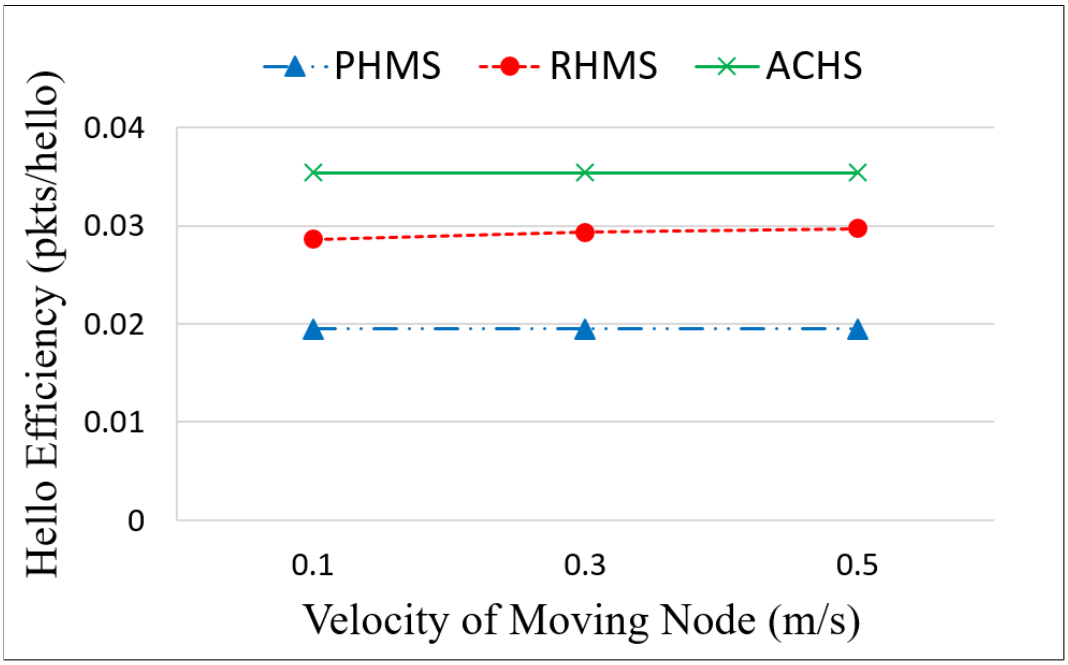


Figure 6.22: Hello Efficiency in the Mobile Scenario

6.6 CRCPR with ACHS versus CWR

The Cooperative Neighbour Table, COP Table and Relay Table in CRCPR require hello messages to be relayed except for normal broadcasting. Also, some additional information like the *NSN Addr List* field needs to be carried compared to a traditional routing protocol. These add to the complexity of CRCPR and the relayed hello messages increase the control overhead in the network. Therefore, ACHS is introduced into CRCPR [185] to compensate for the additional cost. The hello message overhead of CWR versus CRCPR is thus explored.

6.6.1 Scenario

This experiment explores the number of hello messages in CWR and CRCPR relative to the network density. All the nodes are randomly deployed in an area of size 300 meters by 300 meters. In order to realize different network densities, we increase the number of nodes (network scale) in this fixed area from 15 to 55 in 5 steps (10 nodes are added in each step), giving five simulations. For each simulation, we repeat the experiment 30 times with different random node-deployment seeds. The transmission range is 30m. The hello message interval for CWR and CRCPR is set to 1s which is the same as used in AODV [40]. The

simulation duration is set to 20 minutes. Figure 6.23 shows the scenario with 55 nodes.

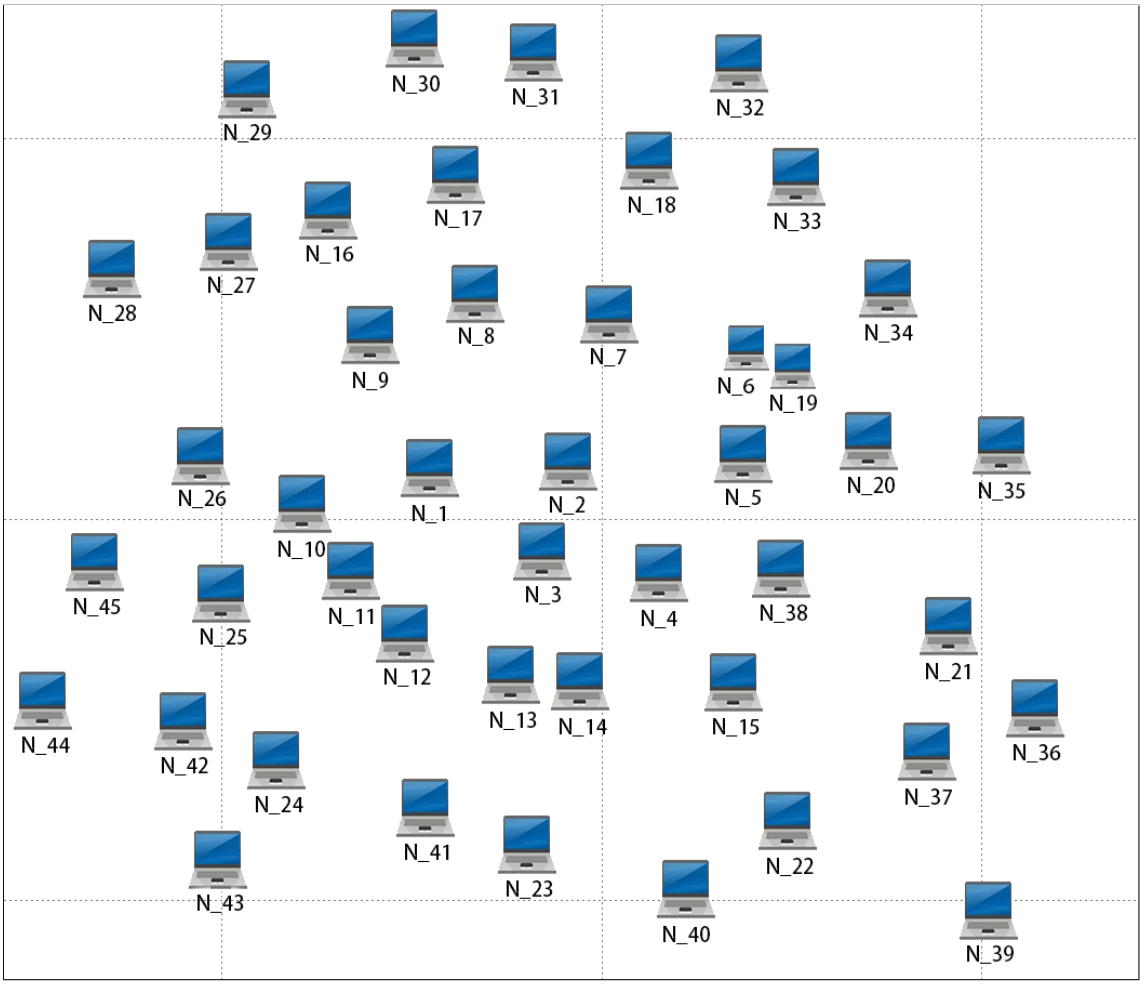


Figure 6.23: Hello Message Enhancement Scenario with 55 Nodes

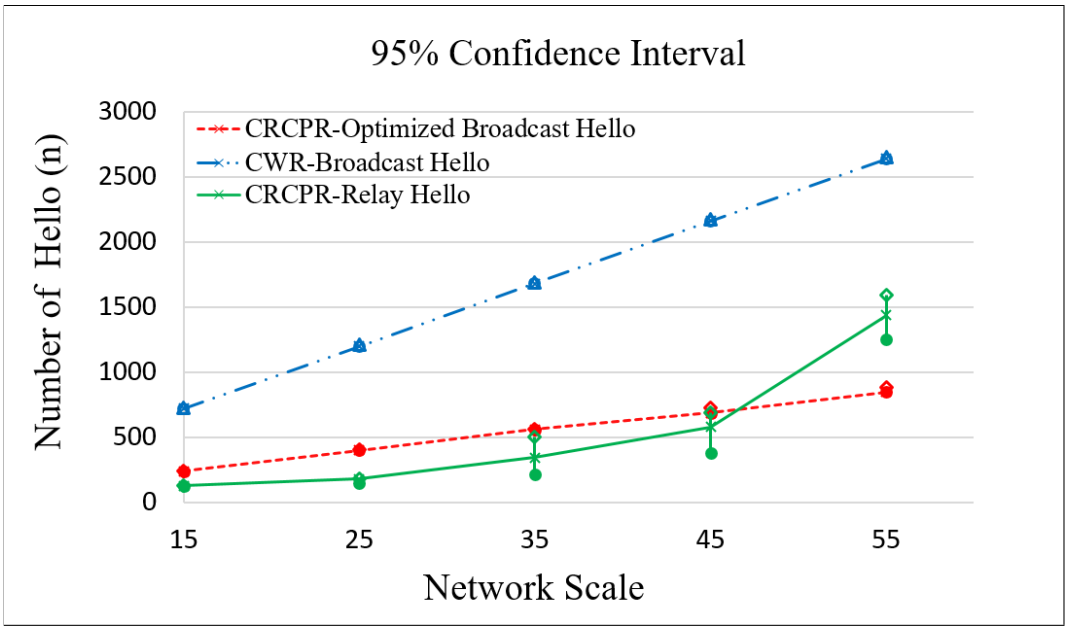
6.6.2 Results

6.6.2.1 Hello Message Enhancement

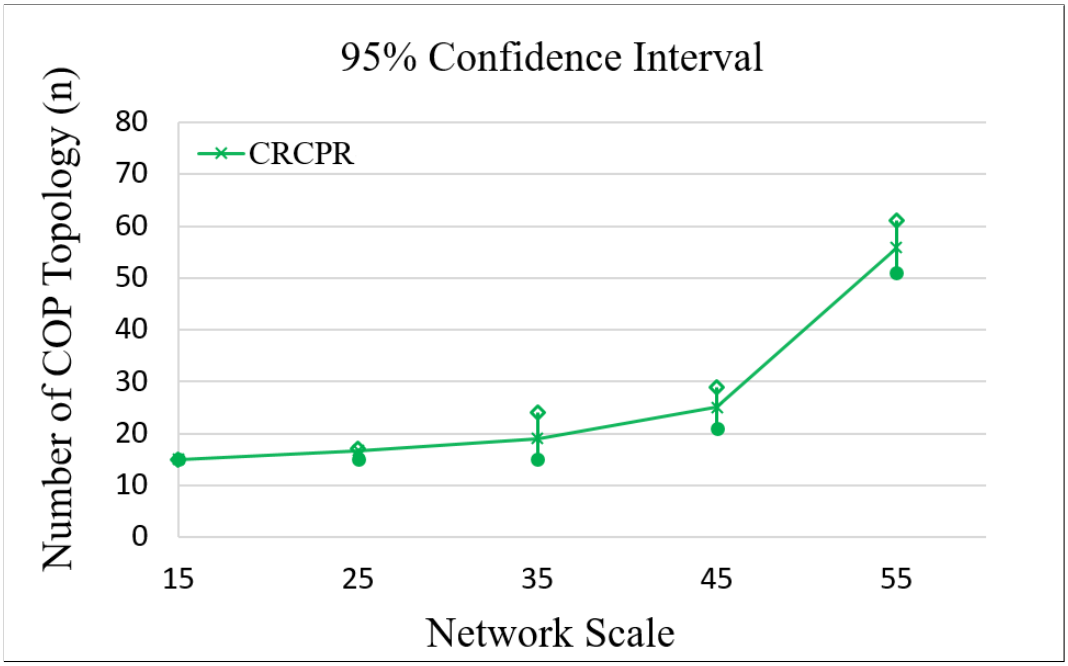
The 95% confidence intervals are provided in the results. From (a) of Figure 6.24, the increment of the number of nodes (network scale) from 15 to 55 in the fixed area is used to realize different network densities.

For CRCPR, the number of broadcast hello messages will increase from 200 to 800 and the number of relay hello messages (unicast hello message) will increase from 100 to 1400 in average. For CWR, the number of broadcast hello messages will increase from 700 to 2700. The increment of the number of hello messages in both protocols is because more nodes in the scenario leads to more hello messages

being sent to maintain the neighbour relationship. However, the broadcast hello message quantity of CRCPR is much lower than CWR. Even if relay hello messages are included, the total number of hello messages in CRCPR is still less. The increasing trend in terms of the number of relay hello messages is caused by the increasing number of COP topologies in CRCPR as the network density increases, which is illustrated in (b) of Figure 6.24. From the results of Figure 6.24, we can conclude that although the Cooperative Neighbour Table, COP Table and Relay Table increase the overhead of CRCPR in terms of hello message transmissions, the ACHS scheme can ameliorate this problem and reduce the number of hello messages without impacting on the overall network performance.



(a) Number of Hello Message



(b) Number of COP Topology

Figure 6.24: Hello Message Enhancement

6.7 Conclusion

In this chapter, we have explored the performance of CRCPR via analysis and simulation. In terms of route robustness, CRCPR shows the best performance compared with two classic MANET routing protocols (AODV and DSR) and one cooperative routing protocol (CWR). In addition, based on the energy harvesting design and new route selection criteria, CRCPR provides a longer network lifetime

compared with two energy-aware routing protocols (DEHAR and AODV-EHA). Although the hello message is modified in CRCPR to support table structures: the Cooperative Neighbour Table, the COP Table and the Relay Table, the control overhead is controlled by the new Adjust Classified Hell Scheme, which provides efficient Neighbour Table updates and typically improve the overall network performance.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

Current infrastructure-based wireless networks like cellular networks and WLANs are no longer suitable for some emerging applications such as IoT. Although MANETs have been recognized as a popular approach for new applications, certain constraints impact on their performance, e.g. the limited communication range of mobile nodes, link breaks due to node mobility and the restricted power supply.

Cooperative communication in MANETs has become an appealing topic as it can improve energy efficiency. However, some issues still remain such as the lack of a systematic designed cooperative routing scheme (including route discovery, route reply, route enhancement and cooperative data forwarding), the use of cooperative communication for mobility resilience, and route selection (jointly considering energy consumption, energy harvesting ability and link break probability).

Driven by the above concerns, we have introduced a cross-layer routing protocol called "Constructive Relay based Cooperative Routing (CRCPR)" based on cooperative communication to support emerging environments in MANETs. By exploiting cooperative communication with information held in a COP Table data structure, energy consumption during transmissions can be significantly reduced. Additionally, by employing a relay principle based on a Relay Table, CRCPR provides greater robustness against node mobility induced link breaks. A new route selection scheme utilizing Routing Matrix Values (RMV) from the Physical/MAC layer, such as the residual energy, energy harvesting ability and link break probability, help to determine the final route.

In order to improve broadcasting efficiency by reducing network congestion

and saving energy, a scheme called "Adaptive Classified Hello Scheme (ACHS)" is designed. As an easy-to-deploy and extensible scheme, ACHS can be readily included in MANET routing protocols using a hello mechanism to improve broadcasting efficiency, but can also be tailored to different scenarios by exploring alternative classification methods.

CRCPR explicitly covers route discovery, route selection, route reply, route maintenance, route enhancement and cooperative data forwarding and is evaluated within an Opnet-based simulation environment. This framework can be readily integrated with existing lower layer mechanisms to improve the performance of MANETs.

7.2 Future Work

Several aspects of this work are worthy of further exploration, as described as follows:

1. In Section 3.3.2.2, when we consider the self-management of the COP topology in CRCPR. The COP Src chooses a suitable entry from its COP Table list and places this entry in a CCON message to notify the proper cooperative nodes to participate in cooperative transmissions. Currently, the suitable entry in the COP Table is identified according to its creation time: the newly created entry is regarded as the most appropriate one to be placed in the CCON message. However, more parameters could be involved in deciding a suitable COP Table entry such as the velocity of the INs, their residual energy, the stability of COP topology and so forth.

2. In order to utilize the well-understood frame sychronization techniques of lower layers, the COP Possibility Detection Algorithm in CRCPR detects and forms a four-node COP topology (one COP Src, one COP Dest and two Cooperative nodes) as in Section 3.3.2.3. This approach is not restrictive. According to the analysis in Section 3.3.6, the link break happens between the COP Src and COP Dest only when all cooperative nodes moves out the COP topology. Thus, if more than two cooperative nodes were to be involved in providing cooperative communication in the future, the robustness against mobility would become even stronger.

3. When we describe the Adjust Classified Hello Scheme in Section 5.3, several scenario-selectable parameters are set manually based on experience such as the Threshold Velocity (V_{th}), the hello interval for Class 2 $h2$ and so on. In future, more scenario factors could be involved in calibrating these parameters automatically. For example, if the residual energy of a node is low, its hello interval could become longer; if the velocity change of a node is high, the observation interval could be reduced to provide a quicker response to network changes.

4. In Section 3.3.9, when we decide the Link Break Degree, p_0 is a key parameter to distinguish between an "extremely unstable" link and an "extremely stable" link. Currently, p_0 is a scenario-selectable parameter based on experience, which can effectively reduce the likelihood of link breaks along the route. In future, different factors could be used to calibrate this parameter automatically. For example, if the velocity of a moving node were to significantly change, p_0 could be automatically updated accordingly.

Appendix A

Flow Chart of Packet Reception

Appendix A shows the flow charts of different packet receptions in CRCPR. As described in Section 4.2.3.2, the implementation of the packet reception *functions* in the Child Process can be programmed according to the following flow charts. All the shaded boxes in the flow charts represent packet discard.

A.1 CREQ Packet Processing Procedure

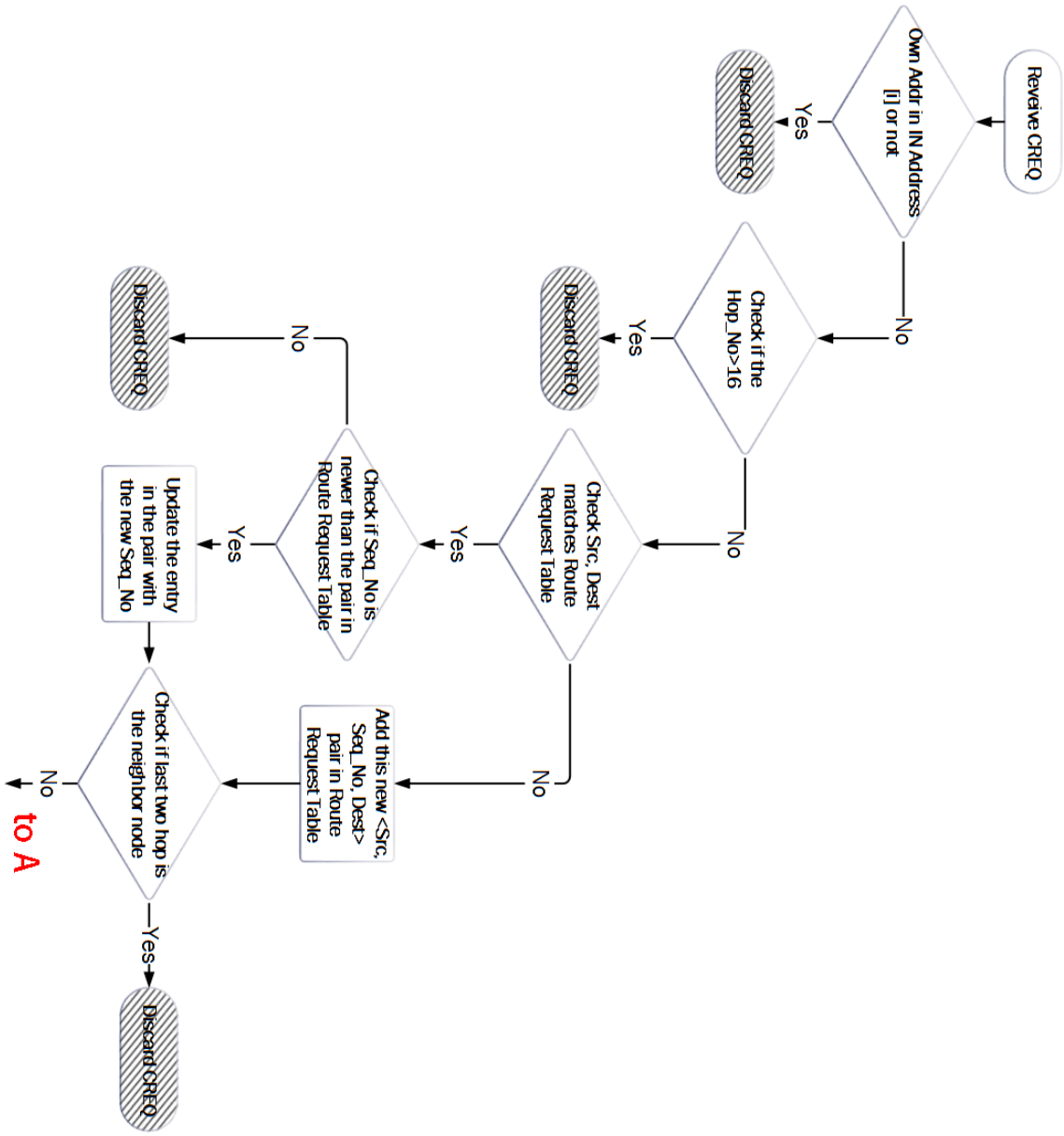


Figure A.1: CREQ Packet Processing Procedure (1/3)

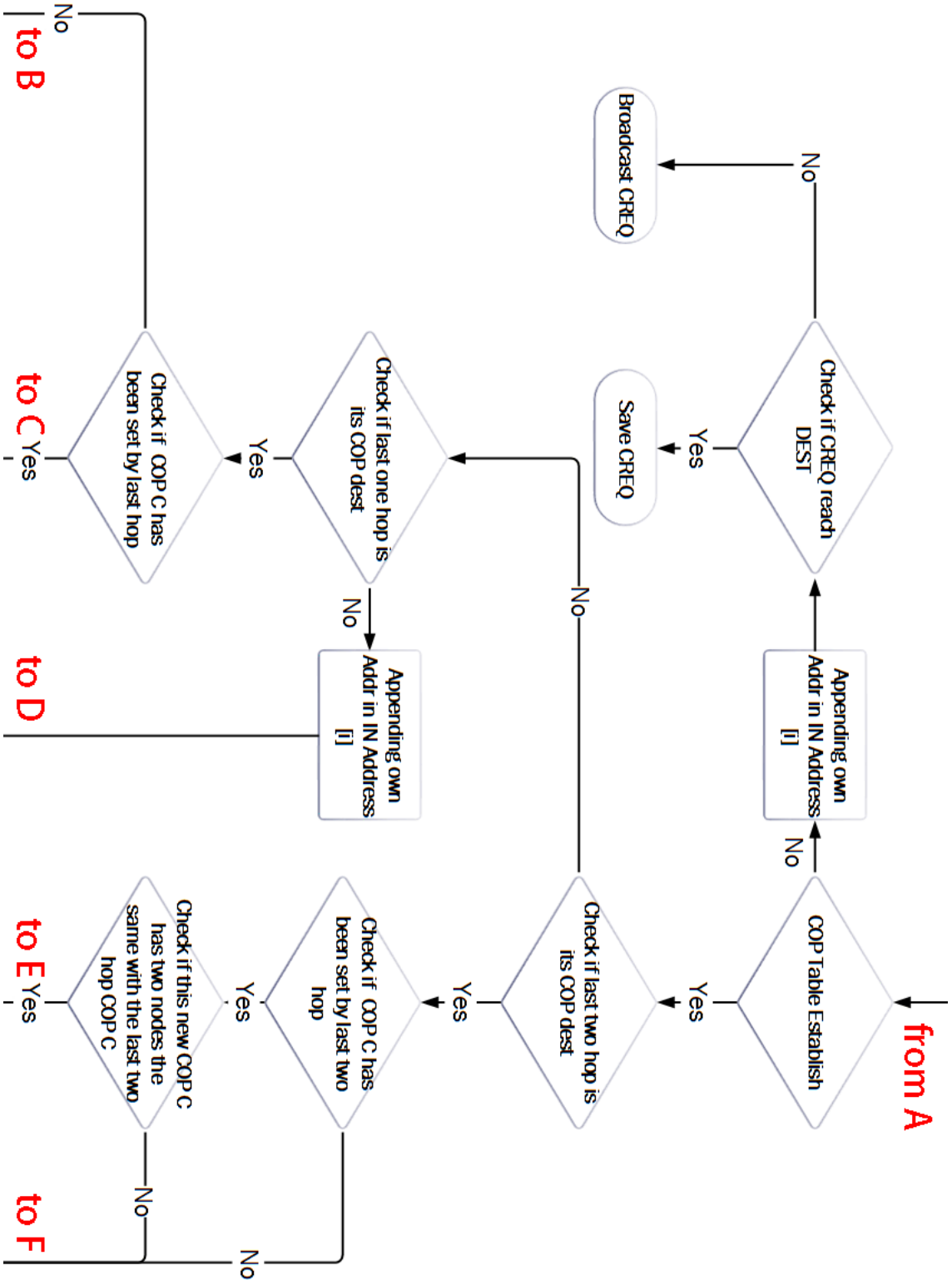


Figure A.2: CREQ Packet Processing Procedure (2/3)

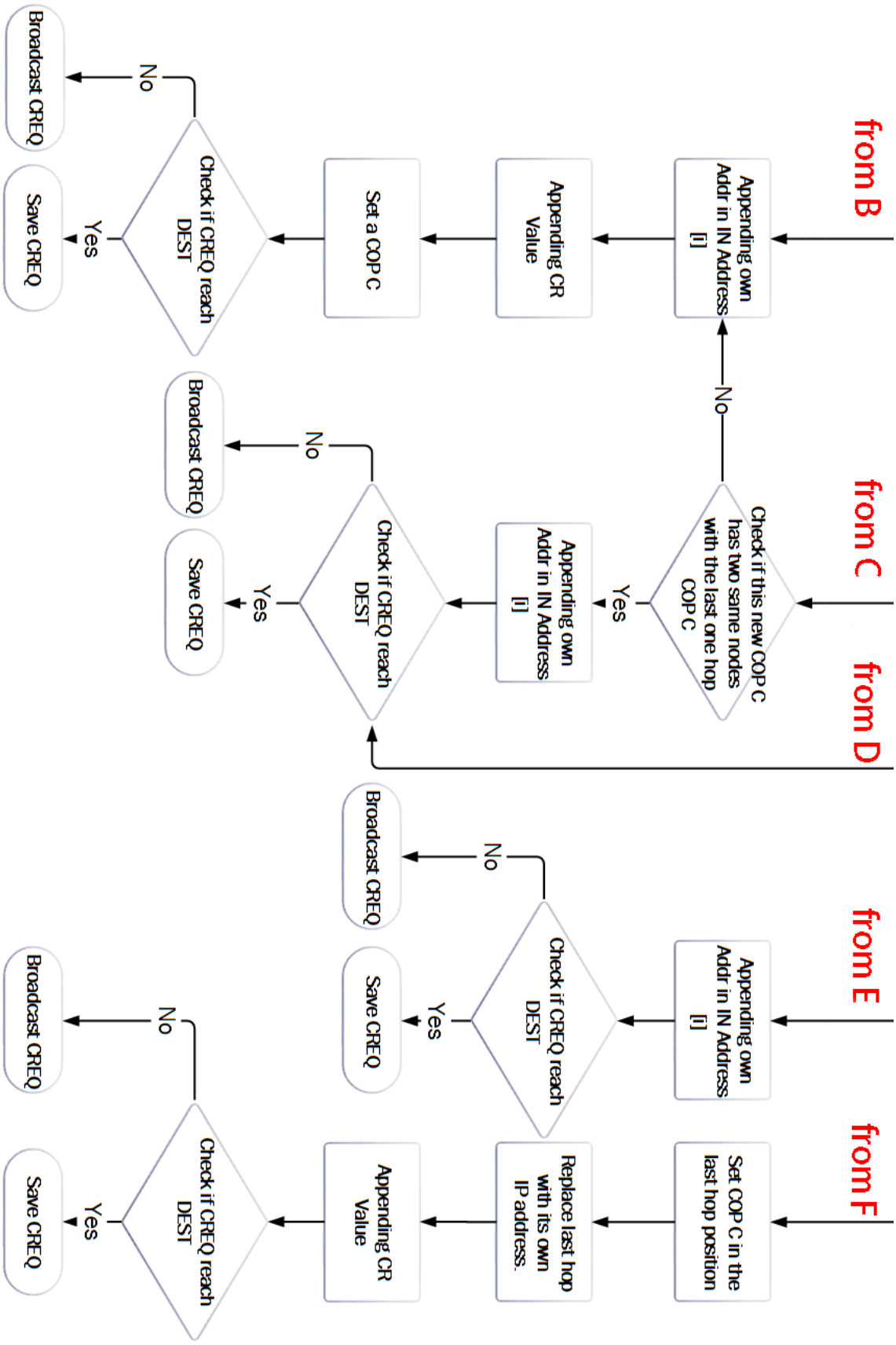


Figure A.3: CREQ Packet Processing Procedure (3/3)

A.2 CREP Packet Processing Procedure

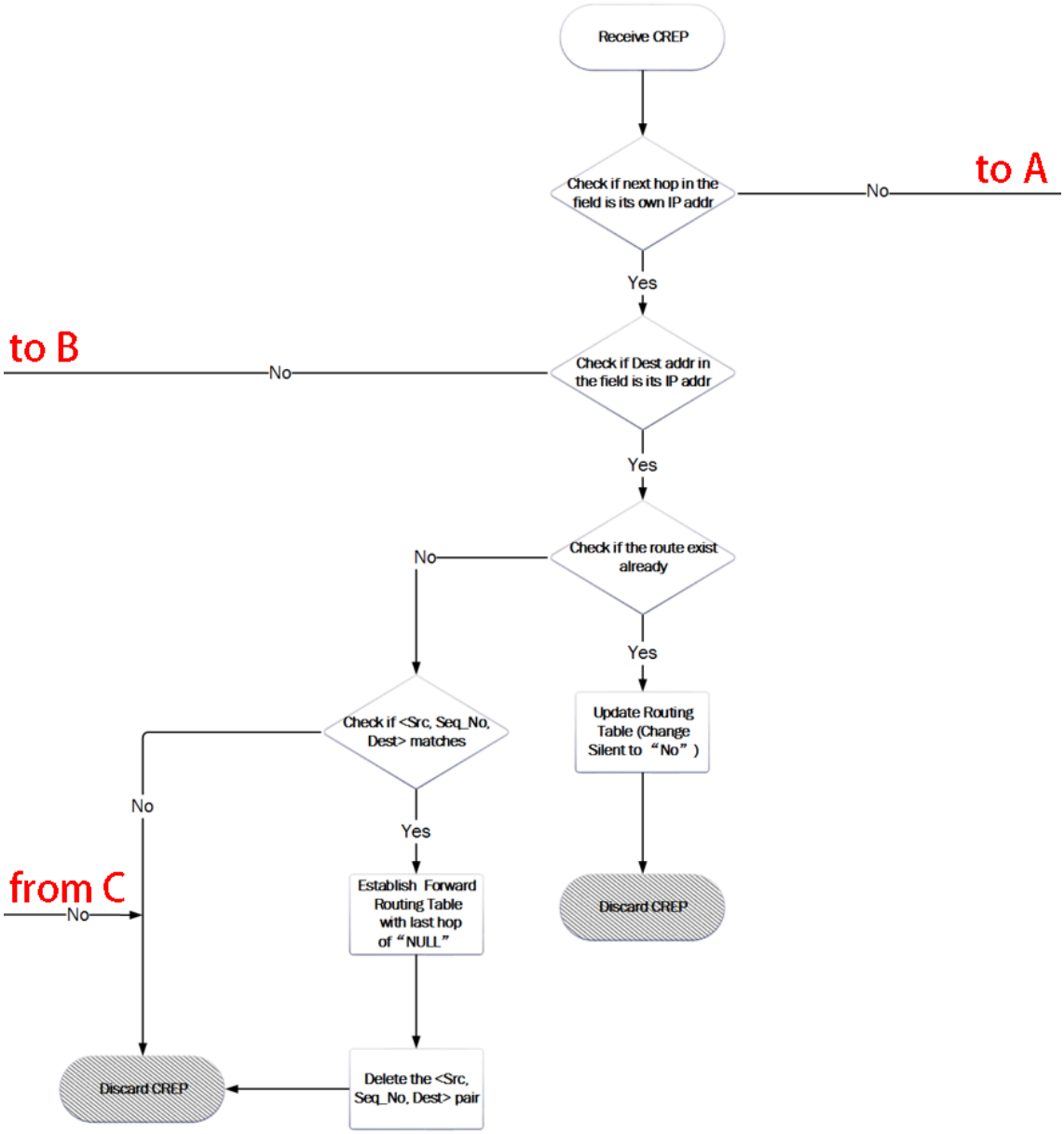


Figure A.4: CREP Packet Processing Procedure (1/3)

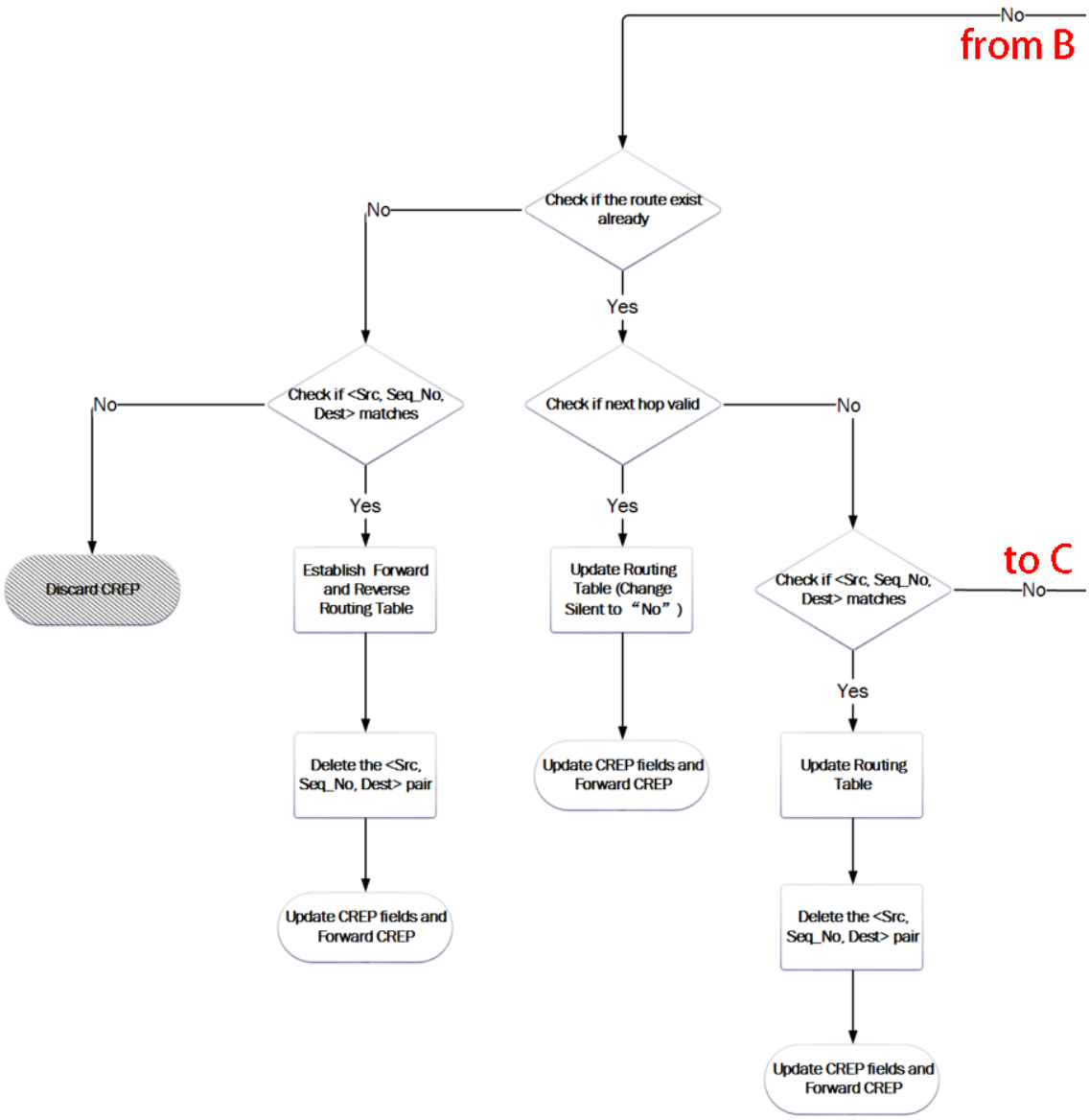


Figure A.5: CREP Packet Processing Procedure (2/3)

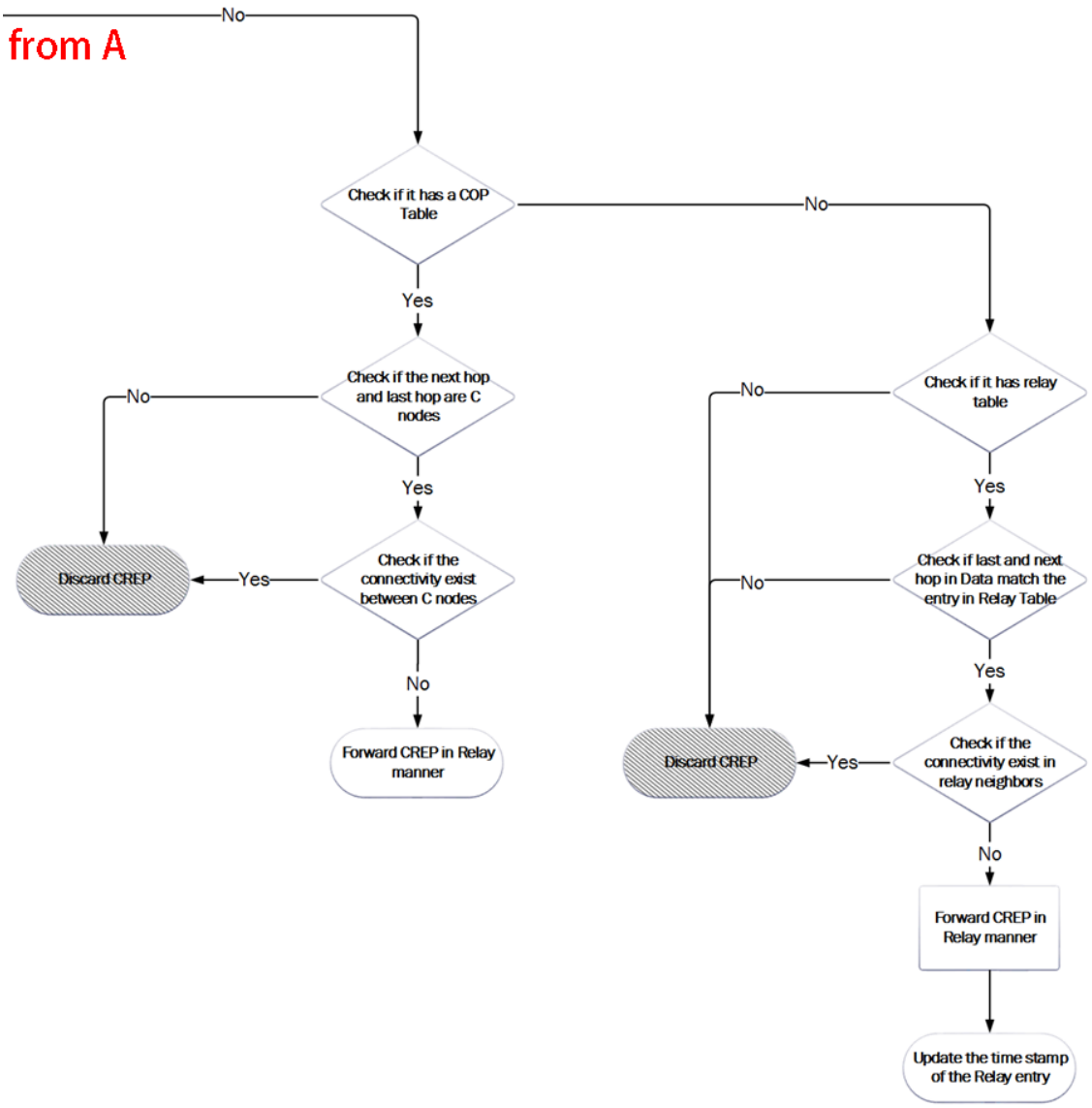


Figure A.6: CREP Packet Processing Procedure (3/3)

A.3 DATA Packet Processing Procedure

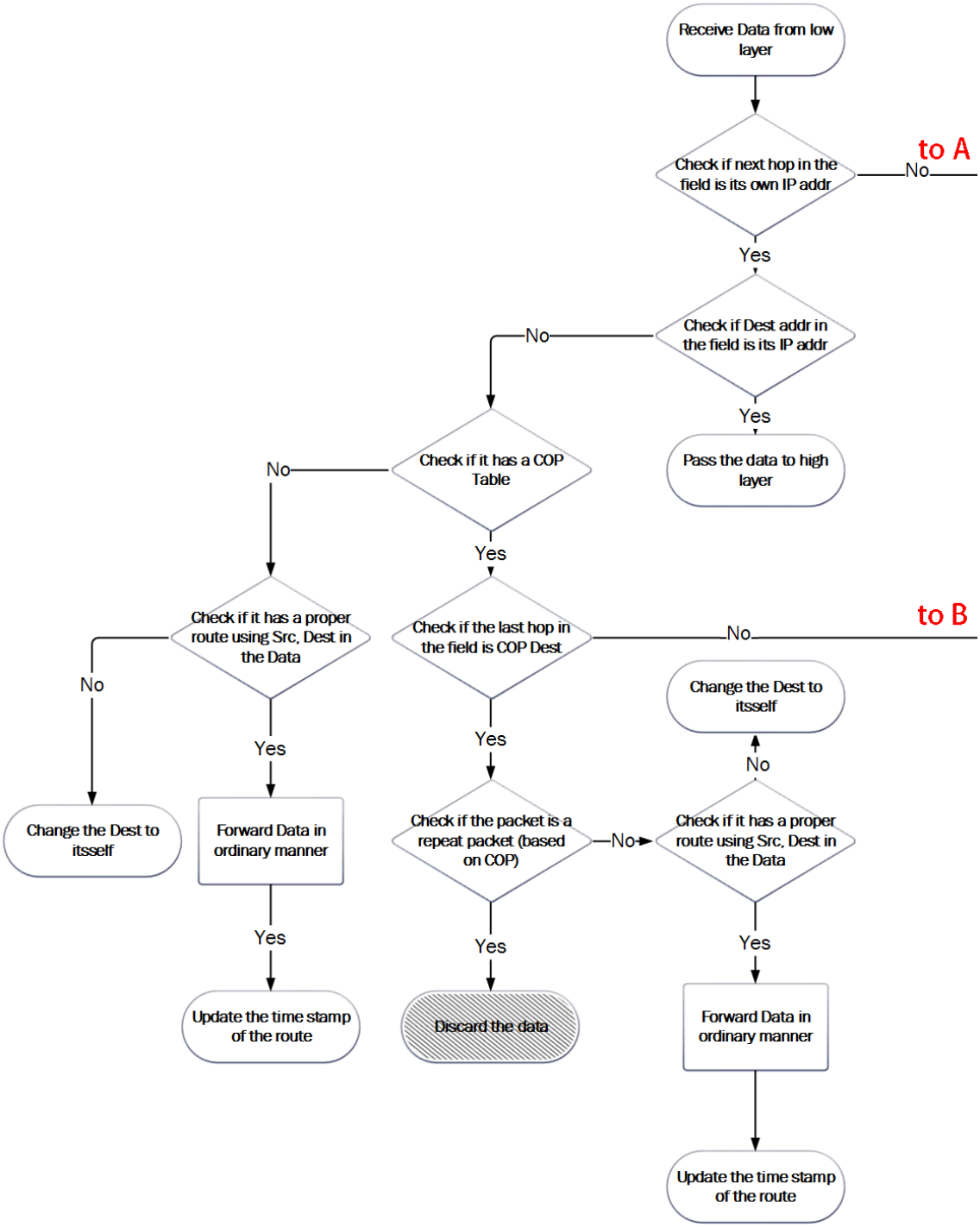


Figure A.7: DATA Packet Processing Procedure (1/3)

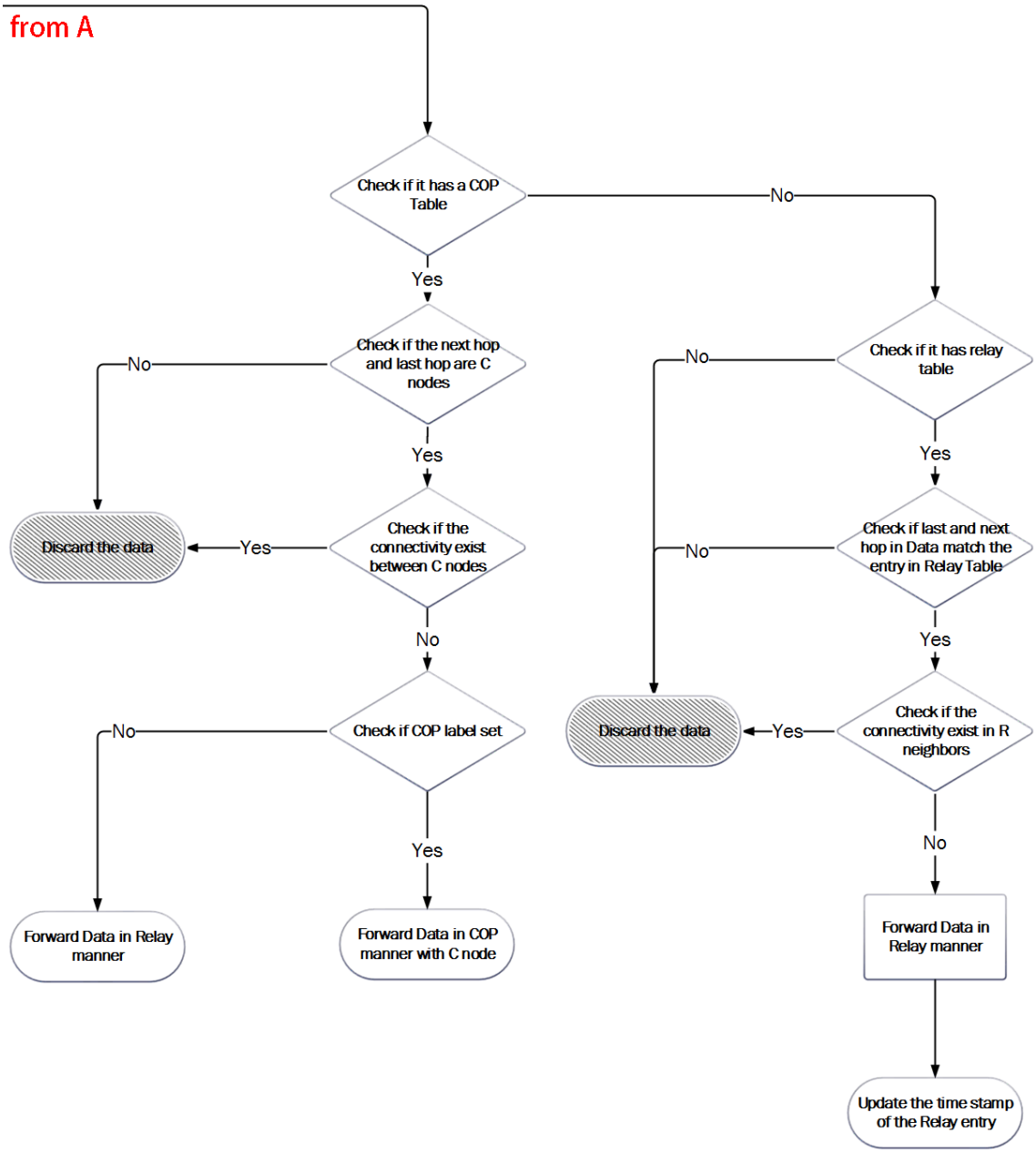


Figure A.8: DATA Packet Processing Procedure (2/3)

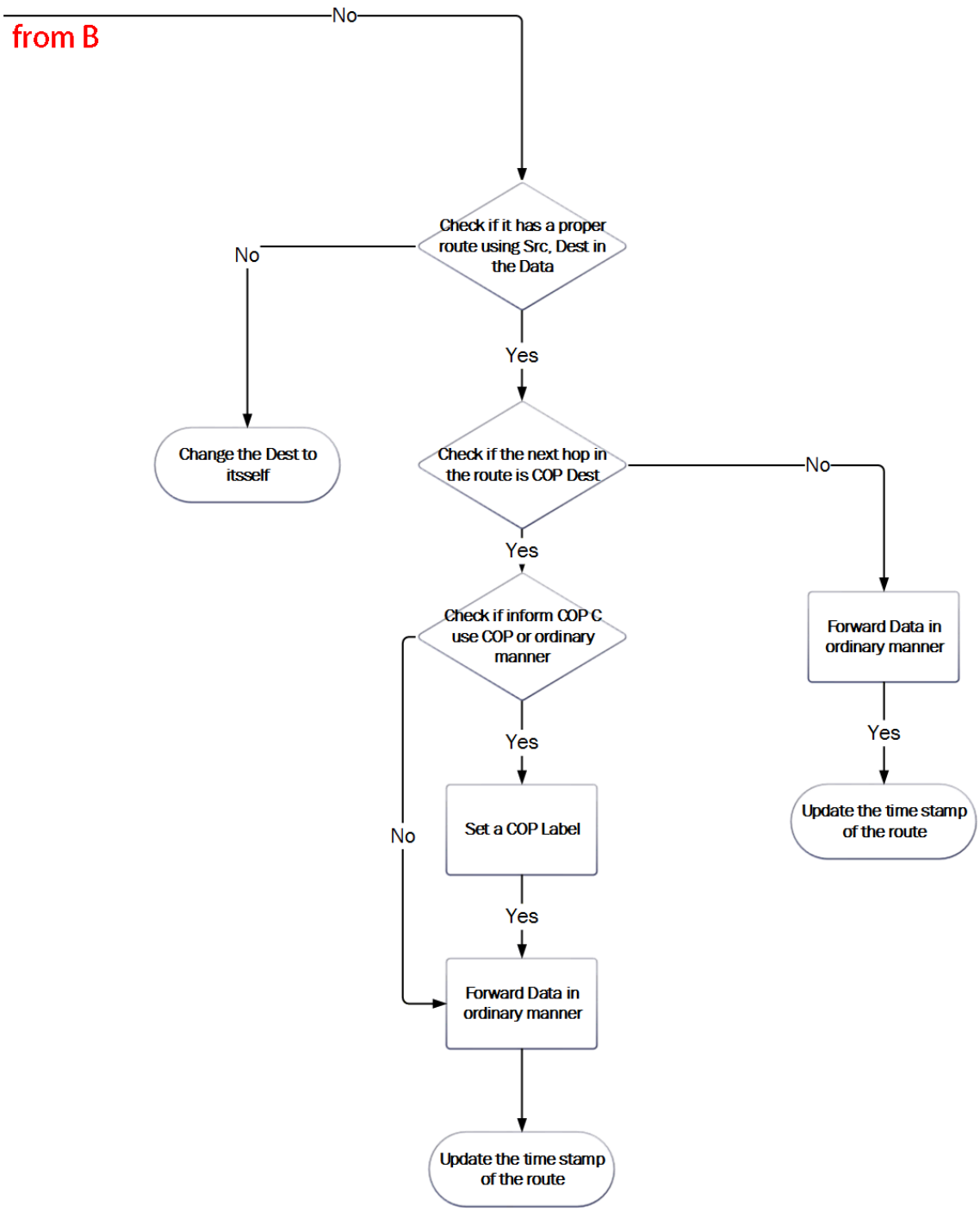


Figure A.9: DATA Packet Processing Procedure (3/3)

A.4 CENF Packet Processing Procedure

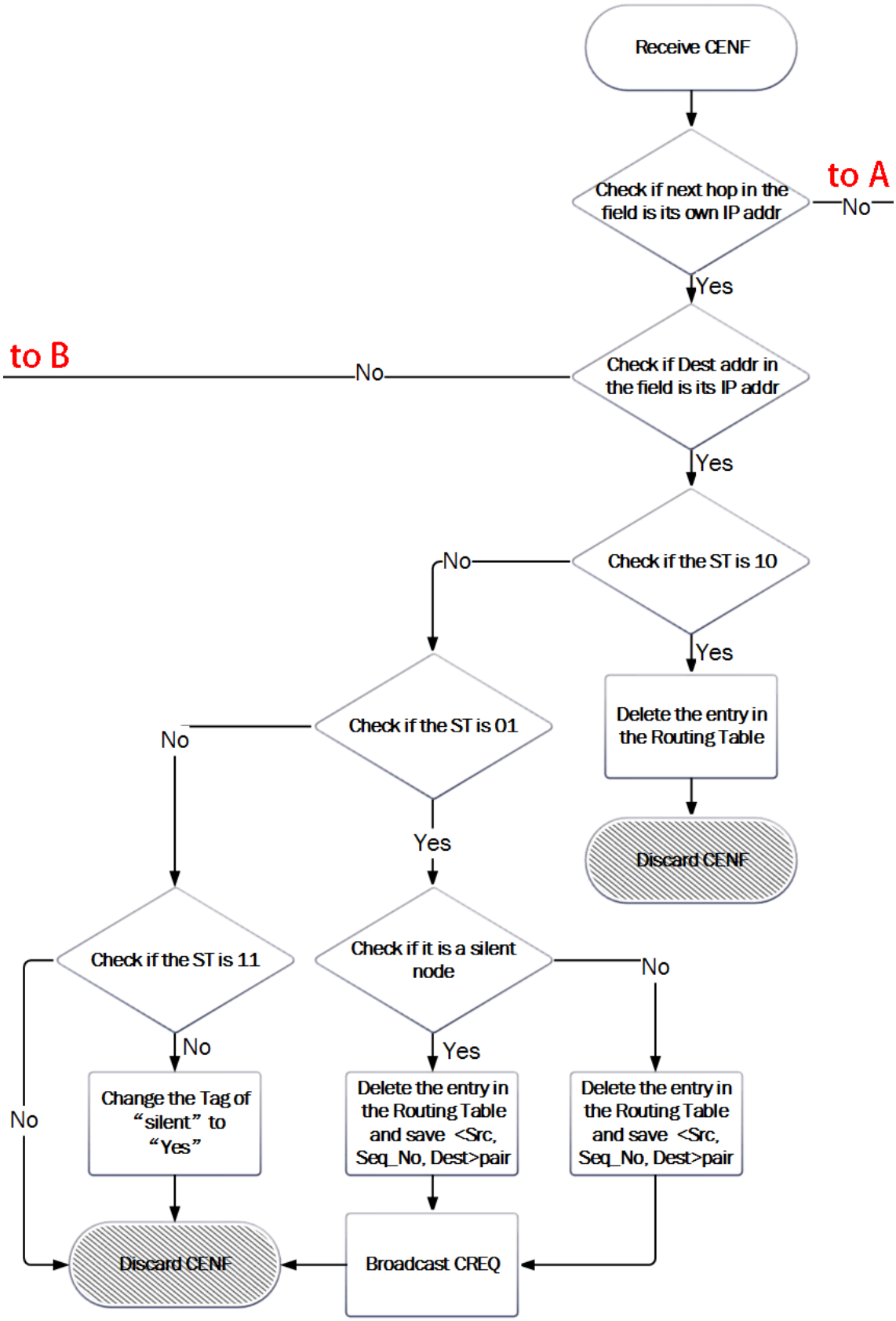


Figure A.10: CENF Packet Processing Procedure (1/4)

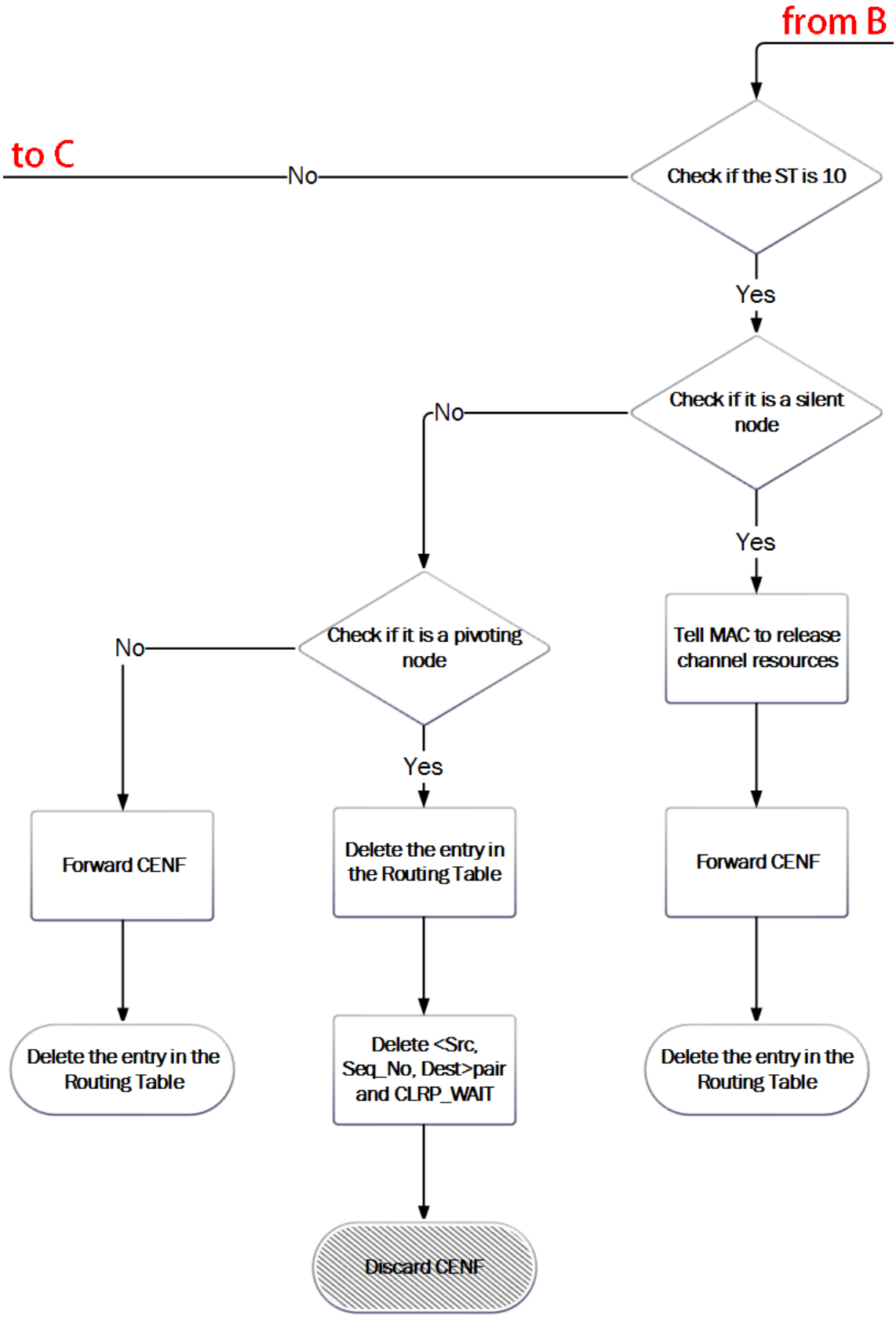


Figure A.11: CENF Packet Processing Procedure (2/4)

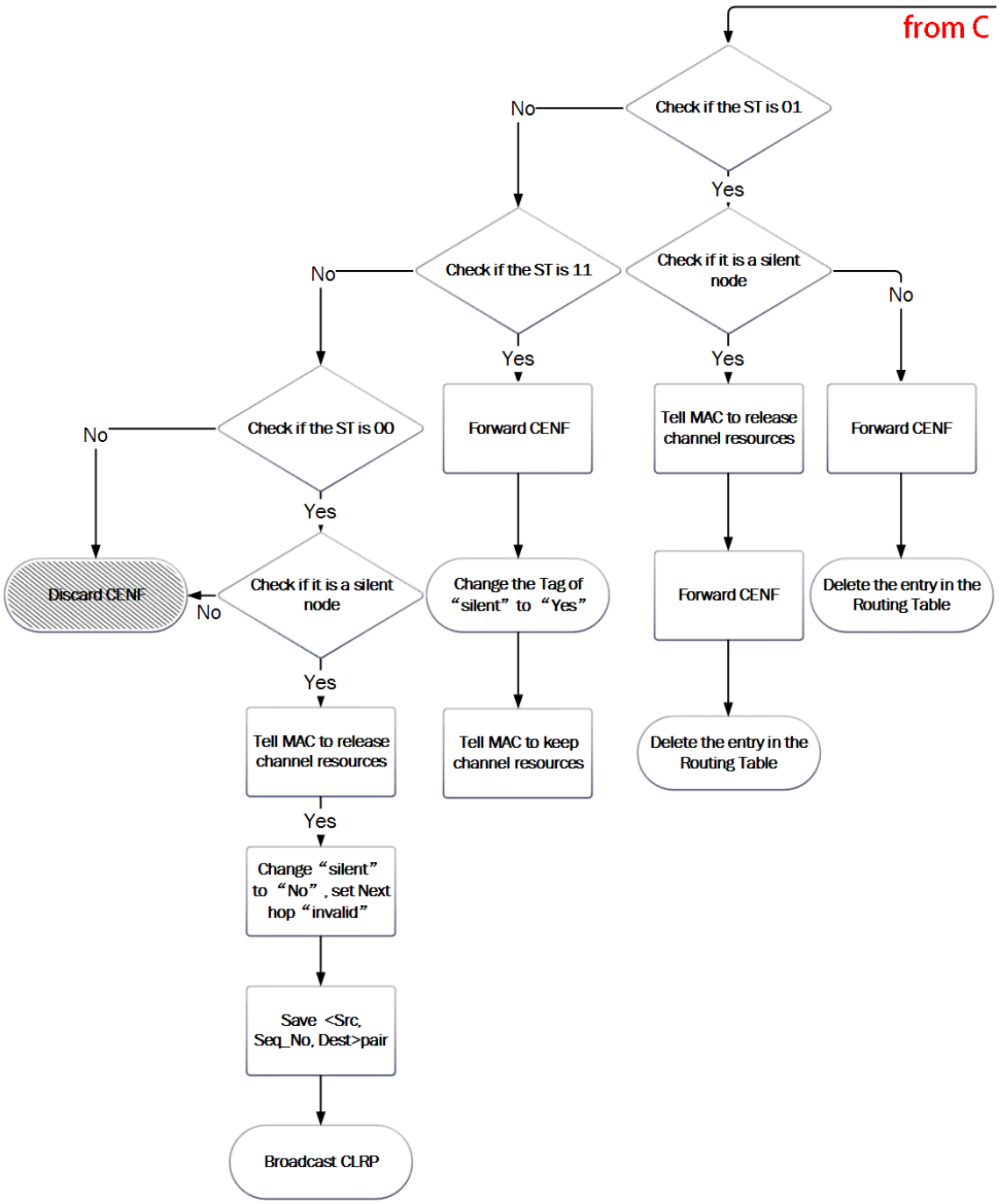


Figure A.12: CENF Packet Processing Procedure (3/4)

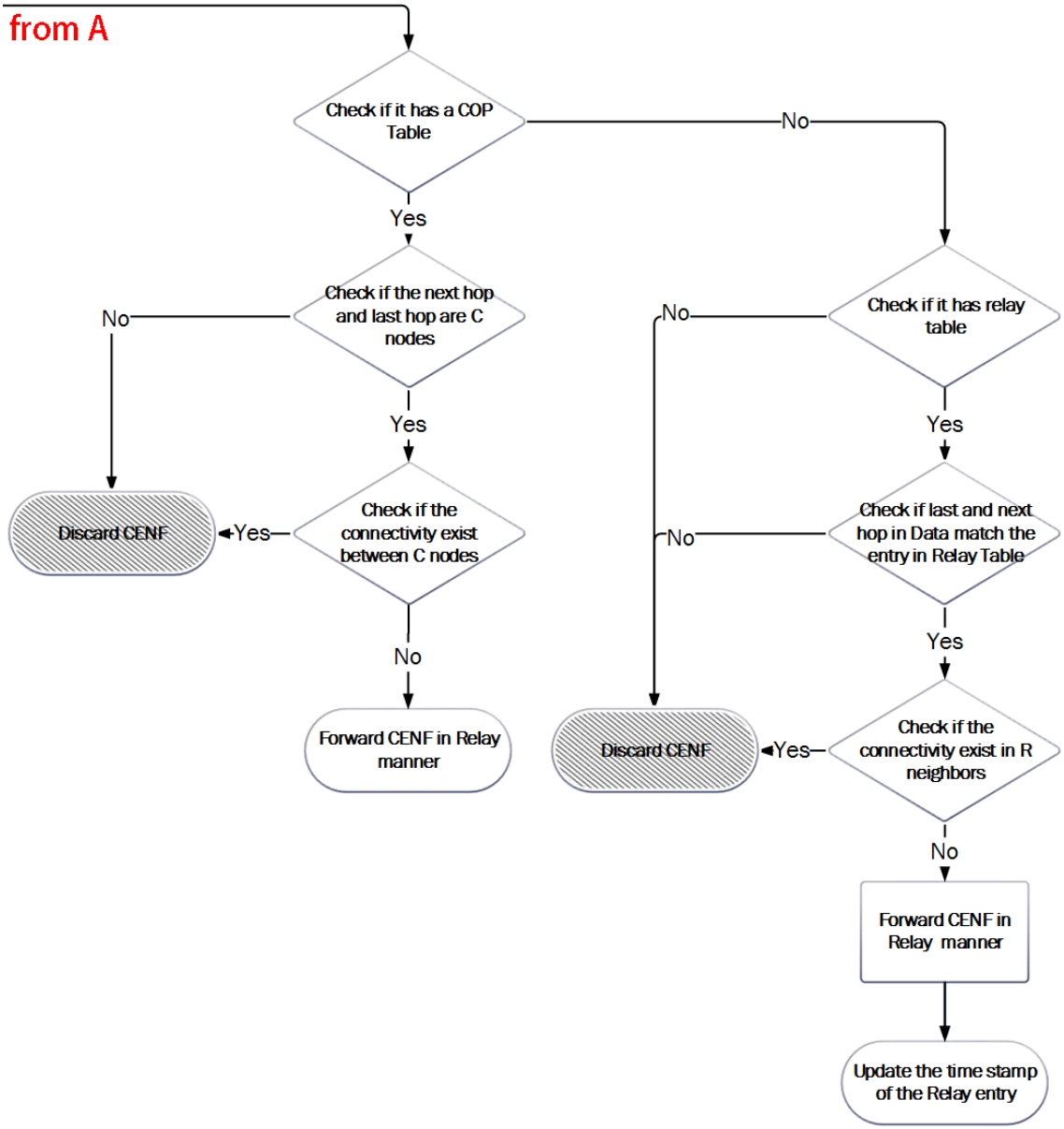


Figure A.13: CENF Packet Processing Procedure (4/4)

Bibliography

[1] *Innovation: The Government Was Crucial After All.* New York Review of Books, 2014.

[2] U. G. Swarnapriyaa, V. Vinodhini, S. Anthoniraj, and R. Anand. Notice of violation of ieee publication principles auto configuration in mobile ad hoc networks. In *Innovations in Emerging Technology (NCOIET), 2011 National Conference on*, pages 61–66, Feb 2011.

[3] J. N. Laneman, D. N. C. Tse, and G. W. Wornell. Cooperative diversity in wireless networks: Efficient protocols and outage behavior. *IEEE Transactions on Information Theory*, 50(12):3062–3080, Dec 2004.

[4] A. Sendonaris, E. Erkip, and B. Aazhang. User cooperation diversity. part i. system description. *IEEE Transactions on Communications*, 51(11):1927–1938, Nov 2003.

[5] A. Sendonaris, E. Erkip, and B. Aazhang. User cooperation diversity. part ii. implementation aspects and performance analysis. *IEEE Transactions on Communications*, 51(11):1939–1948, Nov 2003.

[6] J. N. Laneman and G. W. Wornell. Distributed space-time coded protocols for exploiting cooperative diversity in wireless networks. In *Global Telecommunications Conference, 2002. GLOBECOM '02. IEEE*, volume 1, pages 77–81 vol.1, Nov 2002.

[7] P. Liu, Z. Tao, S. Narayanan, T. Korakis, and S. S. Panwar. Coopmac: A cooperative mac for wireless lans. *IEEE Journal on Selected Areas in Communications*, 25(2):340–354, February 2007.

[8] Hao Zhu and Guohong Cao. rdcf: A relay-enabled medium access control protocol for wireless ad hoc networks. *IEEE Transactions on Mobile Computing*, 5(9):1201–1214, Sept 2006.

[9] J. Alonso-Zarate, C. Verikoukis, E. Kartsakli, and L. Alonso. A novel near-optimum medium access control protocol for a distributed cooperative arq scheme in wireless networks. In *Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on*, pages 1–5, Sept 2008.

[10] H. Adam, W. Elmenreich, C. Bettstetter, and S. M. Senouci. Core-mac: A mac-protocol for cooperative relaying in wireless networks. In *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pages 1–6, Nov 2009.

[11] Ad hoc. Merriam-webster.com. In *Merriam-Webster*, Aug 2016.

[12] C.K. Toh. *Ad Hoc Mobile Wireless Networks: Protocols and Systems*. Pearson Education, 2001.

[13] *Introduction*, pages 1–50. CRC Press, 2016/08/03 2013.

[14] William C. Y. Lee. *Mobile Cellular Telecommunications Systems*. McGraw-Hill, Inc., New York, NY, USA, 1990.

[15] Theodore Rappaport. *Wireless Communications: Principles and Practice*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2nd edition, 2001.

[16] R. E. Kahn, S. A. Gronemeyer, J. Burchfiel, and R. C. Kunzelman. Advances in packet radio technology. *Proceedings of the IEEE*, 66(11):1468–1496, Nov 1978.

[17] L. Kleinrock and J. Silvester. Optimum transmission radii for packet radio networks or why six is a magic number. In *Conference Record, National Telecommunications Conference*, pages 4.3.2–4.3.5, Birmingham, Alabama, December 1978.

[18] A. Ephremides, J. E. Wieselthier, and D. J. Baker. A design concept for reliable mobile radio networks with frequency hopping signaling. *Proceedings of the IEEE*, 75(1):56–73, Jan 1987.

[19] J. Jubin and J. D. Tornow. The darpa packet radio network protocols. *Proceedings of the IEEE*, 75(1):21–32, Jan 1987.

[20] M. B. Pursley. The role of spread spectrum in packet radio networks. *Proceedings of the IEEE*, 75(1):116–1E34, Jan 1987.

[21] F. A. Tobagi. Modeling and performance analysis of multihop packet radio networks. *Proceedings of the IEEE*, 75(1):135–155, Jan 1987.

[22] (2014) Ofcom. The communications market 2014. [online] ofcom. *Available at: http://stakeholders.ofcom.org.uk/market-data-research/market-data/communications-market-reports/cmr14/?a=0*.

[23] *ITU Radio Regulations, CHAPTER II Frequencies, ARTICLE 5 Frequency allocations, Section IV*. Table of Frequency Allocations, 2012.

[24] Sunil Kumar, Vineet S. Raghavan, and Jing Deng. Medium access control protocols for ad hoc wireless networks: A survey. *Ad Hoc Netw.*, 4(3):326–358, May 2006.

[25] K-W Sung G. Miao, J. Zander and B. Slimane. *Fundamentals of Mobile Data Networks*. Cambridge University Press, Redwood City, CA, USA, 2016.

[26] M. Ciampa and J. Olenewa. *Wireless# Guide to Wireless Communications*. Cengage Learning, 2006.

[27] Andrew J. Viterbi. *CDMA: Principles of Spread Spectrum Communication*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA, 1995.

[28] Federico Calì, Marco Conti, and Enrico Gregori. Dynamic ieee 802.11: Design, modeling and performance evaluation. In *Proceedings of the IFIP-TC6 / European Commission International Conference on Broadband Communications, High Performance Networking, and Performance of Communication Networks*, NETWORKING '00, pages 786–798, London, UK, UK, 2000. Springer-Verlag.

[29] P. Karn. Maca - a new channel access method for packet radio. In *Amateur Radio 9th Computer Networking Conference*, pages 134–140, September 2004.

[30] Vaduvur Bharghavan, Alan Demers, Scott Shenker, and Lixia Zhang. Macaw: A media access protocol for wireless lan's. In *Proceedings of the Conference on Communications Architectures, Protocols and Applications*, SIGCOMM '94, pages 212–225, New York, NY, USA, 1994. ACM.

[31] Ansi/ieee std 802.11 - 1999 wireless lan medium access control (mac) and physical (phy) specifications. 1999.

[32] A. Dadej S. Gordon A. Jayasuriya, S. Perreau. Hidden vs. exposed terminal problem in ad hoc networks. In *Proceedings of the Australian Telecommunication Networks and Applications Conference*, Sydney, Australia, 2004.

[33] Charles E. Perkins and Pravin Bhagwat. Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers. In *Proceedings of the Conference on Communications Architectures, Protocols and Applications*, SIGCOMM '94, pages 234–244, New York, NY, USA, 1994. ACM.

[34] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot. Optimized link state routing protocol for ad hoc networks. In *Multi Topic Conference, 2001. IEEE INMIC 2001. Technology for the 21st Century. Proceedings. IEEE International*, pages 62–68, 2001.

[35] R. Ogier, F. Templin, and M. Lewis. Topology dissemination based on reverse-path forwarding (tbrpf), 2004.

[36] Shree Murthy and J. J. Garcia-Luna-Aceves. A routing protocol for packet radio networks. In *Proceedings of the 1st Annual International Conference on Mobile Computing and Networking*, MobiCom '95, pages 86–95, New York, NY, USA, 1995. ACM.

[37] W. Liu C.-C. Chiang, H.-K. Wu and M. Gerla. Routing in clustered multihop, mobile wireless networks with fading chaneel. In *Proceedings of IEEE Singapore International Conference on Networks*, 1997.

[38] J. J. Garcia-Luna-Aceves and M. Spohn. Source adaptive routing in wireless networks. In *IETF Internet Draft*, Oct 1999.

[39] J. J. Garcia-Luna-Aceves and M. Spohn. Source-tree routing in wireless networks. In *Network Protocols, 1999. (ICNP '99) Proceedings. Seventh International Conference on*, pages 273–282, Oct 1999.

[40] Charles E. Perkins and Elizabeth M. Royer. Ad-hoc on-demand distance vector routing. In *Proceedings of the Second IEEE Workshop on Mobile Computer Systems and Applications*, WMCSA '99, pages 90–, Washington, DC, USA, 1999. IEEE Computer Society.

[41] David B. Johnson, David A. Maltz, and Josh Broch. Dsr: The dynamic source routing protocol for multi-hop wireless ad hoc networks. In *In Ad Hoc Networking, edited by Charles E. Perkins, Chapter 5*, pages 139–172. Addison-Wesley, 2001.

[42] V. D. Park, J. P. Macker, and M. S. Corson. draft-ietf-manet-tora-spec-04.txt. In *IETF Internet Draft*, July 2004.

[43] V. D. Park, J. P. Macker, and M. S. Corson. Applicability of the temporally-ordered routing algorithm for use in mobile tactical networks. In *Military Communications Conference, 1998. MILCOM 98. Proceedings., IEEE*, volume 2, pages 426–430 vol.2, Oct 1998.

[44] Chai-Keong Toh. Associativity-based routing for ad hoc mobile networks. *Wirel. Pers. Commun.*, 4(2):103–139, March 1997.

[45] R. Dube, C. D. Rais, Kuang-Yeh Wang, and S. K. Tripathi. Signal stability-based adaptive routing (ssa) for ad hoc mobile networks. *IEEE Personal Communications*, 4(1):36–45, Feb 1997.

[46] M Sanchez and P Manzoni. A java-based ad hoc networks simulator. In *Proceedings of the SCS Western Multiconference Web-based Simulation Track*, pages 573–583, 1999.

[47] J. Boleng T. Camp and V. Davies. A survey of mobility models for ad hoc network research. In *Wireless Communication and Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, volume 2, pages 483–502, 2002.

[48] Muhammad Zaheer Aslam and Abdur Rashid Khan. Comparison of random waypoint & random walk mobility model under dsr, AODV & DSDV MANET routing protocols. *CoRR*, abs/1104.2368, 2011.

[49] David B. Johnson and David A. Maltz. *Dynamic Source Routing in Ad Hoc Wireless Networks*, pages 153–181. Springer US, Boston, MA, 1996.

[50] J. M. Rabaey, M. J. Ammer, J. L. da Silva, D. Patel, and S. Roundy. Picoradio supports ad hoc ultra-low power wireless networking. *Computer*, 33(7):42–48, Jul 2000.

[51] A. Chandrakasan, R. Amirtharajah, Seonghwan Cho, J. Goodman, G. Konduri, J. Kulik, W. Rabiner, and A. Wang. Design considerations for distributed microsensor systems. In *Custom Integrated Circuits, 1999. Proceedings of the IEEE 1999*, pages 279–286, 1999.

[52] J. Rabaey, J. Ammer, J. L. da Silva, and D. Patel. Picoradio: Ad-hoc wireless networking of ubiquitous low-energy sensor/monitor nodes. In *VLSI, 2000. Proceedings. IEEE Computer Society Workshop on*, pages 9–12, 2000.

[53] K. Sohrabi, J. Gao, V. Ailawadhi, and G. J. Pottie. Protocols for self-organization of a wireless sensor network. *IEEE Personal Communications*, 7(5):16–27, Oct 2000.

[54] J. M. Kahn, R. H. Katz, and K. S. J. Pister. Emerging challenges: Mobile networking for smart dust. *Journal of Communications and Networks*, 2(3):188–196, Sept 2000.

[55] Prathima Agrawal. Energy efficient protocols for wireless systems. In *Personal, Indoor and Mobile Radio Communications, 1998. The Ninth IEEE International Symposium on*, volume 2, pages 564–569 vol.2, Sep 1998.

[56] W. R. Heinzelman, A. Sinha, A. Wang, and A. P. Chandrakasan. Energy-scalable algorithms and protocols for wireless microsensor networks. In *Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE International Conference on*, volume 6, pages 3722–3725 vol.6, 2000.

[57] N. Bambos. Toward power-sensitive network architectures in wireless communications: concepts, issues, and design aspects. *IEEE Personal Communications*, 5(3):50–59, Jun 1998.

[58] A. Ephremides. Energy concerns in wireless networks. *IEEE Wireless Communications*, 9(4):48–59, Aug 2002.

[59] A. J. Goldsmith and S. B. Wicker. Design challenges for energy-constrained ad hoc wireless networks. *IEEE Wireless Communications*, 9(4):8–27, Aug 2002.

[60] W. Stark, Hua Wang, A. Worthen, S. Lafortune, and D. Teneketzis. Low-energy wireless communication network design. *IEEE Wireless Communications*, 9(4):60–72, Aug 2002.

[61] Richard Bellman. *On a Routing Problem*. Quarterly of Applied Mathematics, 1958.

[62] University of Santa Cruz and DARPA/ITO. Sparrow - secure protocols for adaptive, robust, reliable, and opportunistic wings. In *Available from: http://www.cse.ucsc.edu/research/ccrg/projects/sparrow.html*, 2000.

[63] M. Abolhasan, T. Wysocki, and E. Dutkiewicz. A review of routing protocols for mobile ad hoc networks. In *Ad hoc Networks*, pages 1–22 vol.2, Jan 2004.

[64] C. Siva Ram Murthy and B. S. Manoj. Routing protocols for ad hoc wireless networks. In *Ad Hoc Wireless Networks: Architectures and Protocols, Chapter 7, Prentice Hall*, June 2004.

[65] V. D. Park and M. S. Corson. A highly adaptive distributed routing algorithm for mobile wireless networks. In *INFOCOM '97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Driving the Information Revolution., Proceedings IEEE*, volume 3, pages 1405–1413 vol.3, Apr 1997.

[66] Douglas S. J. De Couto, Daniel Aguayo, John Bicket, and Robert Morris. A high-throughput path metric for multi-hop wireless routing. In *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking*, MobiCom '03, pages 134–146, New York, NY, USA, 2003. ACM.

[67] Richard Draves, Jitendra Padhye, and Brian Zill. Comparison of routing metrics for static multi-hop wireless networks. In *Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, SIGCOMM '04, pages 133–144, New York, NY, USA, 2004. ACM.

[68] D. B. Johnson. Routing in ad hoc networks of mobile hosts. In *Mobile Computing Systems and Applications, 1994. Proceedings., Workshop on*, pages 158–163, Dec 1994.

[69] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides. Algorithms for energy-efficient multicasting in ad hoc wireless networks. In *Military Communications Conference Proceedings, 1999. MILCOM 1999. IEEE*, volume 2, pages 1414–1418 vol.2, 1999.

[70] E. van der Meulen. A survey of multi-way channels in information theory: 1961-1976. *IEEE Transactions on Information Theory*, 23(1):1–37, Jan 1977.

[71] R. U. Nabar, H. Bolcskei, and F. W. Kneubuhler. Fading relay channels: performance limits and space-time signal design. *IEEE Journal on Selected Areas in Communications*, 22(6):1099–1109, Aug 2004.

[72] J. N. Laneman and G. W. Wornell. Distributed space-time-coded protocols for exploiting cooperative diversity in wireless networks. *IEEE Transactions on Information Theory*, 49(10):2415–2425, Oct 2003.

[73] D. Gesbert, M. Shafi, Da shan Shiu, P. J. Smith, and A. Naguib. From theory to practice: an overview of mimo space-time coded wireless systems. *IEEE Journal on Selected Areas in Communications*, 21(3):281–302, Apr 2003.

[74] S. M. Alamouti. A simple transmit diversity technique for wireless communications. *IEEE J.Sel. A. Commun.*, 16(8):1451–1458, September 2006.

[75] G. Scutari and S. Barbarossa. Distributed space-time coding for regenerative relay networks. *IEEE Transactions on Wireless Communications*, 4(5):2387–2399, Sept 2005.

[76] J. N. Laneman, D. N. C. Tse, and G. W. Wornell. Cooperative diversity in wireless networks: Efficient protocols and outage behavior. *IEEE Transactions on Information Theory*, 50(12):3062–3080, Dec 2004.

[77] Douglas S. J. De Couto, Daniel Aguayo, John Bicket, and Robert Morris. A high-throughput path metric for multi-hop wireless routing. In *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking*, MobiCom '03, pages 134–146, New York, NY, USA, 2003. ACM.

[78] Richard Draves, Jitendra Padhye, and Brian Zill. Comparison of routing metrics for static multi-hop wireless networks. *SIGCOMM Comput. Commun. Rev.*, 34(4):133–144, August 2004.

[79] A. Scaglione and Yao-Win Hong. Opportunistic large arrays: cooperative transmission in wireless multihop ad hoc networks to reach far distances. *IEEE Transactions on Signal Processing*, 51(8):2082–2092, Aug 2003.

[80] J. N. Laneman, D. N. C. Tse, and G. W. Wornell. Cooperative diversity in wireless networks: Efficient protocols and outage behavior. *IEEE Transactions on Information Theory*, 50(12):3062–3080, Dec 2004.

[81] Sanjit Biswas and Robert Morris. Exor: Opportunistic multi-hop routing for wireless networks. *SIGCOMM Comput. Commun. Rev.*, 35(4):133–144, August 2005.

[82] Szymon Chachulski, Michael Jennings, Sachin Katti, and Dina Katabi. Trading structure for randomness in wireless opportunistic routing. In *Proceedings of the 2007 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, SIGCOMM '07, pages 169–180, New York, NY, USA, 2007. ACM.

[83] Christina Fragouli, Jean-Yves Le Boudec, and Jörg Widmer. Network coding: An instant primer. *SIGCOMM Comput. Commun. Rev.*, 36(1):63–68, January 2006.

[84] I. Leontiadis and C. Mascolo. Geopps: Geographical opportunistic routing for vehicular networks. In *2007 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*, pages 1–6, June 2007.

[85] S. Yang, F. Zhong, C. K. Yeo, B. S. Lee, and J. Boleng. Position based opportunistic routing for robust data delivery in manets. In *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pages 1–6, Nov 2009.

[86] S. Murthy. Routing in packet-switched networks using path-finding algorithms. *Ph.D. dissertation*, 1996.

[87] Jochen Behrens and J. J. Garcia-Luna-Aceves. Distributed, scalable routing based on link-state vectors. In *Proceedings of the Conference on Communications Architectures, Protocols and Applications*, SIGCOMM '94, pages 136–147, New York, NY, USA, 1994. ACM.

[88] Z. Wang, Y. Chen, and C. Li. Corman: A novel cooperative opportunistic routing scheme in mobile ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 30(2):289–296, February 2012.

[89] Shuguang Cui, A. J. Goldsmith, and A. Bahai. Energy-efficiency of mimo and cooperative mimo techniques in sensor networks. *IEEE Journal on Selected Areas in Communications*, 22(6):1089–1098, Aug 2004.

[90] Xiaohua Li. Energy efficient wireless sensor networks with transmission diversity. *Electronics Letters*, 39(24):1753–1755, Nov 2003.

[91] Xiaohua Li, Mo Chen, and Wenyu Liu. Application of stbc-encoded cooperative transmissions in wireless sensor networks. *IEEE Signal Processing Letters*, 12(2):134–137, Feb 2005.

[92] S. K. Jayaweera. Energy analysis of mimo techniques in wireless sensor networks. In *in 38th Annual Conference on Information Sciences and Systems (CISS 04)*, Princeton, NJ, USA, March 1999.

[93] G. Farhadi and N. C. Beaulieu. Ergodic capacity of multi-hop wireless relaying systems in rayleigh fading. In *IEEE GLOBECOM 2008 - 2008 IEEE Global Telecommunications Conference*, pages 1–6, Nov 2008.

[94] S. Gherekhloo B. Zafar, D. Schulz and M. Haardt. Ad-hoc networking solutions - cooperative mimo multi-hop networks. In *IEEE Vehicular Technology Magazine*, March 2011.

[95] D. Schulz M. Haardt B. Zafar, S. Gherekhloo and J. Seitz. A novel multi-hop routing scheme for ad-hoc cooperative mimo systems. In *in Proceedings of the 26th Wireless World Research Forum (WWRF)*, Doha, April 2011.

[96] B. Zafar, S. Gherekhloo, and M. Haardt. A joint clustering and routing scheme to maximize link performance in cooperative mimo ad-hoc networks. In *Personal Indoor and Mobile Radio Communications (PIMRC), 2011 IEEE 22nd International Symposium on*, pages 51–55, Sept 2011.

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

[98] W. Ge, J. Zhang, and G. Xue. Joint clustering and optimal cooperative routing in wireless sensor networks. In *2008 IEEE International Conference on Communications*, pages 2216–2220, May 2008.

[99] Z. Sheng, Z. Ding, and K. K. Leung. On the design of a quality-of-service driven routing protocol for wireless cooperative networks. In *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, pages 624–628, May 2008.

[100] Chao Chen, Baoyu Zheng, and Xianjing Zhao. An ad hoc cooperative routing algorithm based on optimal channel selection. In *Wireless, Mobile and Multimedia Networks, 2006 IET International Conference on*, pages 1–4, Nov 2006.

[101] Yang Liu, Jianfeng Zhang, Jingmei Song, Renhui Xu, and Zefeng Chen. An ad hoc cooperative routing algorithm based on symmetric link selection. In *Communication Technology (ICCT), 2010 12th IEEE International Conference on*, pages 1156–1159, Nov 2010.

[102] A. Qayyum, L. Viennot, and A. Laouiti. Multipoint relaying for flooding broadcast messages in mobile wireless networks. In *System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on*, pages 3866–3875, Jan 2002.

[103] A. Nosratinia, T. E. Hunter, and A. Hedayat. Cooperative communication in wireless networks. *IEEE Communications Magazine*, 42(10):74–80, Oct 2004.

[104] G. Kramer, M. Gastpar, and P. Gupta. Cooperative strategies and capacity theorems for relay networks. *IEEE Transactions on Information Theory*, 51(9):3037–3063, Sept 2005.

[105] R. Madan, N. B. Mehta, A. F. Molisch, and J. Zhang. Energy-efficient cooperative relaying over fading channels with simple relay selection. *IEEE Transactions on Wireless Communications*, 7(8):3013–3025, August 2008.

[106] Y. Nebat and R. L. Cruz. Routing, cooperative transmission and the relaying bound: the effect of multi-user diversity. *Proc. 2005 CISS, Princeton, NJ*, August 2005.

[107] J. Luo, R. S. Blum, L. J. Cimini, L. J. Greenstein, and A. M. Haimovich. Link-failure probabilities for practical cooperative relay networks. In *2005 IEEE 61st Vehicular Technology Conference*, volume 3, pages 1489–1493 Vol. 3, May 2005.

[108] A. Stefanov and E. Erkip. Cooperative space-time coding for wireless networks. *IEEE Transactions on Communications*, 53(11):1804–1809, Nov 2005.

[109] A. Bletsas, A. Khisti, D. P. Reed, and A. Lippman. A simple cooperative diversity method based on network path selection. *IEEE Journal on Selected Areas in Communications*, 24(3):659–672, March 2006.

[110] R. Madan, N. B. Mehta, A. F. Molisch, and J. Zhang. Energy-efficient decentralized cooperative routing in wireless networks. *IEEE Transactions on Automatic Control*, 54(3):512–527, March 2009.

[111] F. Xianyang and W. Feng. Design and implementation of interference-aware cooperative qos routing for multi-hop wireless network. In *2014 Sixth International Conference on Measuring Technology and Mechatronics Automation*, pages 211–217, Jan 2014.

[112] J. N. Laneman, G. W. Wornell, and D. N. C. Tse. An efficient protocol for realizing cooperative diversity in wireless networks. In *Information Theory, 2001. Proceedings. 2001 IEEE International Symposium on*, pages 294–, 2001.

[113] T. E. Hunter and A. Nosratinia. Cooperation diversity through coding. In *Information Theory, 2002. Proceedings. 2002 IEEE International Symposium on*, pages 220–, 2002.

[114] T. E. Hunter and A. Nosratinia. Diversity through coded cooperation. *IEEE Transactions on Wireless Communications*, 5(2):283–289, Feb 2006.

[115] Wing-Hin Wong, J. M. Shea, and T. F. Wong. Cooperative-diversity slotted aloha. In *Second International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks (QSHINE'05)*, pages 8 pp.–35, Aug 2005.

[116] S. Moh, C. Yu, S. M. Park, H. N. Kim, and J. Park. Cd-mac: Cooperative diversity mac for robust communication in wireless ad hoc networks. In *2007 IEEE International Conference on Communications*, pages 3636–3641, June 2007.

[117] A. Azgin, Y. Altunbasak, and G. AlRegib. Cooperative mac and routing protocols for wireless ad hoc networks. In *GLOBECOM '05. IEEE Global Telecommunications Conference, 2005.*, volume 5, pages 6 pp.–2859, Dec 2005.

[118] A. Sharma, V. Gelara, S. R. Singh, T. Korakis, Pei Liu, and S. Panwar. Implementation of a cooperative mac protocol using a software defined radio platform. In *2008 16th IEEE Workshop on Local and Metropolitan Area Networks*, pages 96–101, Sept 2008.

[119] Q. Chen, Q. Zhang, and Z. Niu. Qos-aware cooperative and opportunistic scheduling exploiting multiuser diversity for rate-adaptive ad hoc networks. *IEEE Transactions on Vehicular Technology*, 57(2):1113–1125, March 2008.

[120] T. Korakis, Z. Tao, S. Singh, P. Liu, and S. Panwar. Implementation of a cooperative mac protocol: Performance and challenges in a real environment. *EURASIP Journal on Wireless Communications and Networking*, May 2009.

[121] Thanasis Korakis, Zhifeng Tao, Salik Makda, Boris Gitelman, and Shivendra Panwar. *It Is Better to Give Than to Receive – Implications of Cooperation in a Real Environment*, pages 427–438. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[122] A. E. Khandani, J. Abounadi, E. Modiano, and L. Zheng. Cooperative routing in static wireless networks. *IEEE Transactions on Communications*, 55(11):2185–2192, Nov 2007.

[123] B. Maham, M. Debbah, and A. Hjorungnes. Energy-efficient cooperative routing in ber constrained multihop networks. In *Communications and Networking in China, 2008. ChinaCom 2008. Third International Conference on*, pages 168–172, Aug 2008.

[124] Yuxia Lin, Joo-Han Song, and Vincent W. S. Wong. Cooperative protocols design for wireless ad-hoc networks with multi-hop routing. In *Proceedings of the 5th International ICST Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*, QShine '08, pages 8:1–8:7, ICST, Brussels, Belgium, Belgium, 2008. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[125] S. C. Draper, L. Liu, A. F. Molisch, and J. S. Yedidia. Routing in cooperative wireless networks with mutual-information accumulation. In *2008 IEEE International Conference on Communications*, pages 4272–4277, May 2008.

[126] F. Li, K. Wu, and A. Lippman. Energy-efficient cooperative routing in multihop wireless ad hoc networks. In *2006 IEEE International Performance Computing and Communications Conference*, pages 8 pp.–222, April 2006.

[127] A. S. Ibrahim, Z. Han, and K. J. R. Liu. Distributed energy-efficient cooperative routing in wireless networks. In *IEEE GLOBECOM 2007 - IEEE Global Telecommunications Conference*, pages 4413–4418, Nov 2007.

[128] B. Maham, R. Narasimhan, and A. Hjorungnes. Energy-efficient space-time coded cooperative routing in multihop wireless networks. In *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pages 1–7, Nov 2009.

[129] S. Chen, M. Huang, Y. Li, Y. Zhu, and Y. Wang. Energy-balanced cooperative routing in multihop wireless ad hoc networks. In *Communications (ICC), 2012 IEEE International Conference on*, pages 307–311, June 2012.

[130] A. Stefanov and E. Erkip. Cooperative information transmission in wireless networks. In *in Proc. AsianEur. Workshop Inf. Theory*, Breisach, Germany, June 2002.

[131] M. Elhawary and Z. J. Haas. Energy-efficient protocol for cooperative networks. *IEEE/ACM Transactions on Networking*, 19(2):561–574, April 2011.

[132] M. Z. Siam, M. Krunz, and O. Younis. Energy-efficient clustering/routing for cooperative mimo operation in sensor networks. In *INFOCOM 2009, IEEE*, pages 621–629, April 2009.

[133] Z. Sheng, Z. Ding, and K. K. Leung. Distributed and power efficient routing in wireless cooperative networks. In *2009 IEEE International Conference on Communications*, pages 1–5, June 2009.

[134] Yu Wang, Xinyu Yang, Shusen Yang, W. Yu, S. Bhattarai, Dan Shen, and Genshe Chen. Towards energy-efficient cooperative routing algorithms in wireless networks. In *Consumer Communications and Networking Conference (CCNC), 2013 IEEE*, pages 79–84, Jan 2013.

[135] Jabed Faruque and Ahmed Helmy. Gradient-based routing in sensor networks. *SIGMOBILE Mob. Comput. Commun. Rev.*, 7(4):50–52, October 2003.

[136] Dimitrios J. Vergados, Nikolaos A. Pantazis, and Dimitrios D. Vergados. Energy-efficient route selection strategies for wireless sensor networks. *Mob. Netw. Appl.*, 13(3-4):285–296, August 2008.

[137] L. A. Bush, C. D. Carothers, and B. K. Szymanski. Algorithm for optimizing energy use and path resilience in sensor networks. In *Proceeedings of the Second European Workshop on Wireless Sensor Networks, 2005.*, pages 391–396, Jan 2005.

[138] Baoxian Zhang and H. T. Mouftah. Adaptive energy-aware routing protocols for wireless ad hoc networks. In *Quality of Service in Heterogeneous*

Wired/Wireless Networks, 2004. QSHINE 2004. First International Conference on, pages 252–259, Oct 2004.

[139] H. Hassanein and Jing Luo. Reliable energy aware routing in wireless sensor networks. In *Second IEEE Workshop on Dependability and Security in Sensor Networks and Systems*, pages 54–64, April 2006.

[140] R. C. Shah and J. M. Rabaey. Energy aware routing for low energy ad hoc sensor networks. In *Wireless Communications and Networking Conference, 2002. WCNC2002. 2002 IEEE*, volume 1, pages 350–355 vol.1, Mar 2002.

[141] Raminder P. Mann, Kamesh R. Namuduri, and Ravi Pendse. Energy-aware routing protocol for ad hoc wireless sensor networks. *EURASIP J. Wirel. Commun. Netw.*, 2005(5):635–644, October 2005.

[142] S. D. Muruganathan, D. C. F. Ma, R. I. Bhasin, and A. O. Fapojuwo. A centralized energy-efficient routing protocol for wireless sensor networks. *IEEE Communications Magazine*, 43(3):S8–13, March 2005.

[143] C. Ma and Y. Yang. Battery-aware routing for streaming data transmissions in wireless sensor networks. In *2nd International Conference on Broadband Networks, 2005.*, pages 464–473 Vol. 1, Oct 2005.

[144] Jinghao Xu, B. Peric, and B. Vojcic. Energy-aware and link-adaptive routing metrics for ultra wideband sensor networks. In *2nd International Workshop Networking with Ultra Wide Band and Workshop on Ultra Wide Band for Sensor Networks, 2005. Networking with UWB 2005.*, pages 1–8, July 2005.

[145] Thiemo Voigt, Hartmut Ritter, and Jochen Schiller. *Solar-Aware Routing in Wireless Sensor Networks*, pages 847–852. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003.

[146] T. Voigt, A. Dunkels, J. Alonso, H. Ritter, and J. Schiller. Solar-aware clustering in wireless sensor networks. In *Computers and Communications, 2004. Proceedings. ISCC 2004. Ninth International Symposium on*, volume 1, pages 238–243 Vol.1, June 2004.

[147] J. Islam, M. Islam, and N. Islam. A-sleach: An advanced solar aware leach protocol for energy efficient routing in wireless sensor networks. In *Network-*

ing, 2007. ICN '07. Sixth International Conference on, pages 4–4, April 2007.

[148] Emanuele Lattanzi, Edoardo Regini, Andrea Acquaviva, and Alessandro Bogliolo. Energetic sustainability of routing algorithms for energy-harvesting wireless sensor networks. *Computer Communications*, 30(1415):2976 – 2986, 2007. Network Coverage and Routing Schemes for Wireless Sensor Networks.

[149] L. Lin, N. B. Shroff, and R. Srikant. Asymptotically optimal energy-aware routing for multihop wireless networks with renewable energy sources. *IEEE/ACM Transactions on Networking*, 15(5):1021–1034, Oct 2007.

[150] Kai Zeng, Kui Ren, Wenjing Lou, and Patrick J. Moran. Energy-aware geographic routing in lossy wireless sensor networks with environmental energy supply. In *Proceedings of the 3rd International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks*, QShine '06, New York, NY, USA, 2006. ACM.

[151] M. K. Jakobsen, J. Madsen, and M. R. Hansen. Dehar: A distributed energy harvesting aware routing algorithm for ad-hoc multi-hop wireless sensor networks. In *World of Wireless Mobile and Multimedia Networks (WoWMoM), 2010 IEEE International Symposium on a*, pages 1–9, June 2010.

[152] P. Gong, Q. Xu, and T. M. Chen. Energy harvesting aware routing protocol for wireless sensor networks. In *Communication Systems, Networks Digital Signal Processing (CSNDSP), 2014 9th International Symposium on*, pages 171–176, July 2014.

[153] S. Sudevalayam and P. Kulkarni. Energy harvesting sensor nodes: Survey and implications. *IEEE Communications Surveys Tutorials*, 13(3):443–461, Third 2011.

[154] Sravanthi Chalasani and J. M. Conrad. A survey of energy harvesting sources for embedded systems. In *IEEE SoutheastCon 2008*, pages 442–447, April 2008.

[155] Shad Roundy, Dan Steingart, Luc Frechette, Paul Wright, and Jan Rabaey. *Power Sources for Wireless Sensor Networks*, pages 1–17. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[156] C. Park and P. H. Chou. Ambimax: Autonomous energy harvesting platform for multi-supply wireless sensor nodes. In *2006 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, volume 1, pages 168–177, Sept 2006.

[157] Sheetalkumar Doshi, Shweta Bhandare, and Timothy X Brown. An on-demand minimum energy routing protocol for a wireless ad hoc network. *SIGMOBILE Mob. Comput. Commun. Rev.*, 6(3):50–66, June 2002.

[158] K. Kar, Murali Kodialam, T. V. Lakshman, and L. Tassiulas. Routing for network capacity maximization in energy-constrained ad-hoc networks. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 1, pages 673–681 vol.1, March 2003.

[159] Jae-Hwan Chang and L. Tassiulas. Energy conserving routing in wireless ad-hoc networks. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 1, pages 22–31 vol.1, 2000.

[160] C. Pandana, W. P. Siriwongpairat, T. Himsoon, and K. J. R. Liu. Distributed cooperative routing algorithms for maximizing network lifetime. In *IEEE Wireless Communications and Networking Conference, 2006. WCNC 2006.*, volume 1, pages 451–456, April 2006.

[161] C. Mikeka and H. Arai. Design of a cellular energy-harvesting radio. In *Wireless Technology Conference, 2009. EuWIT 2009. European*, pages 73–76, Sept 2009.

[162] K. K. Win, X. Wu, S. Dasgupta, W. J. Wen, R. Kumar, and S. K. Panda. Efficient solar energy harvester for wireless sensor nodes. In *Communication Systems (ICCS), 2010 IEEE International Conference on*, pages 289–294, Nov 2010.

[163] M. Maleki, K. Dantu, and M. Pedram. Lifetime prediction routing in mobile ad hoc networks. In *Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE*, volume 2, pages 1185–1190 vol.2, March 2003.

[164] L. Gong, Y. Bai, M. Chen, and D. Qian. Link availability prediction in ad hoc networks. In *Parallel and Distributed Systems, 2008. ICPADS '08. 14th IEEE International Conference on*, pages 423–428, Dec 2008.

[165] Xinjie Chang. Network simulations with opnet. In *Proceedings of the 31st Conference on Winter Simulation: Simulation—a Bridge to the Future - Volume 1*, WSC '99, pages 307–314, New York, NY, USA, 1999. ACM.

[166] Y. Yemini. The osi network management model. *Comm. Mag.*, 31(5):20–29, May 1993.

[167] Doo-Hyun Sung, Joo-Sang Youn, Ji-Hoon Lee, and Chul-Hee Kang. A local repair scheme with adaptive promiscuous mode in mobile ad hoc networks. In *Proceedings of the First International Conference on Mobile Ad-hoc and Sensor Networks*, MSN'05, pages 351–361, Berlin, Heidelberg, 2005. Springer-Verlag.

[168] Andreas Maeder and Nader Zein. Ofdma in the field: Current and future challenges. *SIGCOMM Comput. Commun. Rev.*, 40(5):71–76, October 2010.

[169] P. Tomer and M. Chandra. An application of routing protocols for vehicular ad-hoc networks. In *Networking and Information Technology (ICNIT), 2010 International Conference on*, pages 157–160, June 2010.

[170] B.H. Sathyaraj and R.C. Doss. Route maintenance using mobility prediction for mobile ad hoc networks. In *Mobile Adhoc and Sensor Systems Conference, 2005. IEEE International Conference on*, pages 6 pp.–101, Nov 2005.

[171] Bo Xue, Pinyi Ren, and Shuangcheng Yan. Link optimization ad-hoc on-demand multipath distance vector routing for mobile ad-hoc networks. In *Wireless Communications, Networking and Mobile Computing, 2009. WiCom '09. 5th International Conference on*, pages 1–6, Sept 2009.

[172] S. Kumar, R.K. Rathy, and D. Pandey. Traffic pattern based performance comparison of two reactive routing protocols for ad hoc networks using ns2.

In *Computer Science and Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on*, pages 369–373, Aug 2009.

[173] C.E. Perkins and E.M. Royer. Ad-hoc on-demand distance vector routing. In *Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA '99. Second IEEE Workshop on*, pages 90–100, Feb 1999.

[174] Stefano Basagni, Imrich Chlamtac, Violet R. Syrotiuk, and Barry A. Woodward. A distance routing effect algorithm for mobility (dream). In *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, MobiCom '98, pages 76–84, New York, NY, USA, 1998. ACM.

[175] Brad Karp and H. T. Kung. Gpsr: Greedy perimeter stateless routing for wireless networks. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, MobiCom '00, pages 243–254, New York, NY, USA, 2000. ACM.

[176] C.K. Toh. *Ad Hoc Mobile Wireless Networks: Protocols and Systems*. Pearson Education, 2001.

[177] V.C. Giruka and M. Singhal. Hello protocols for ad-hoc networks: overhead and accuracy tradeoffs. In *World of Wireless Mobile and Multimedia Networks, 2005. WoWMoM 2005. Sixth IEEE International Symposium on a*, pages 354–361, June 2005.

[178] D JOHNSON. The dynamic source routing protocol (dsr) for mobile ad hoc networks for ipv4. *IETF-Request-for-Comments, rfc4728. txt*, 2007.

[179] John T. Moy. *OSPF: Anatomy of an Internet Routing Protocol*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1998.

[180] K. Tsukamoto, S. Koba, M. Tsuru, and Y. Oie. Cognitive radio-aware transport protocol for mobile ad hoc networks. *IEEE Transactions on Mobile Computing*, 14(2):288–301, Feb 2015.

[181] Y. Song and J. Xie. Bracer: A distributed broadcast protocol in multi-hop cognitive radio ad hoc networks with collision avoidance. *IEEE Transactions on Mobile Computing*, 14(3):509–524, March 2015.

[182] Y. Liu and W. Trappe. Topology adaptation for robust ad hoc cyberphysical networks under puncture-style attacks. *Tsinghua Science and Technology*, 20(4):364–375, August 2015.

[183] S. Yang, C. K. Yeo, and B. S. Lee. Toward reliable data delivery for highly dynamic mobile ad hoc networks. *IEEE Transactions on Mobile Computing*, 11(1):111–124, Jan 2012.

[184] A. Fridman, S. Weber, C. Graff, D. E. Breen, K. R. Dandekar, and M. Kam. Oman: A mobile ad hoc network design system. *IEEE Transactions on Mobile Computing*, 11(7):1179–1191, July 2012.

[185] Jingwen B., Yan S., and C. Ph. Hello message scheme enhancement in crmanet. In *Wireless Telecommunications Symposium (WTS)*, April 2016.