A Cross Layer Routing Protocol for OFDMA Based Mobile Ad Hoc Networks.

Xiong, Hong Yi

A Cross Layer Routing Protocol for

OFDMA Based Mobile Ad Hoc

Networks

Submitted for the degree of Doctor of Philosophy

HONG YI XIONG

School of Electronic Engineering and Computer Science

Queen Mary University of London

Supervisor: Dr Eliane Bodnese

30th May 2013

Abstract

Mobile ad hoc networks are of growing interest because of their unique characteristics and advantages in many practical applications. QoS provision acts as a major challenge in the routing protocol design in the real-world mobile ad hoc networks, especially for the real-time services. OFDM is a new technology which has many advantages over the other modulation schemes. Because of its prominent features, many popular wireless standards have adopted it as physical layer modulation, such as IEEE 802.11 series, WiMAX, 3GPP LTE etc, and it is extended to multiuser environment known as OFDMA. So far none of the existing ad hoc routing protocols fully account for the OFDMA based mobile ad hoc networks. In this thesis, a QoS routing protocol is proposed for OFDMA based mobile ad hoc networks. A signal strength-based sub-channel allocation scheme is proposed in the routing protocol aiming to reduce the signalling overhead and co-channel interference. The performance of the proposed routing protocol is compared with other alternative proposals through simulations using OPNET simulator. Moreover, a partial time synchronization and a null subcarrier based frequency synchronization algorithms are also proposed for OFDMA based ad hoc network to further support and facilitate the proposed sub-channel allocation scheme and routing protocol.

Acknowledgement

I would like to give my grateful thanks to my supervisor Dr. Eliane Bodanese for her insightful help and support from the very beginning of my PhD study in Queen Mary, University of London. She has given me a lot of valuable advice on how to become a professional research student. She is also a very patient supervisor who sorts out any problems I encountered during the research. I also want to thanks my second supervisor Professor Xiaodong Chen and Independent Assessor Dr Raul who gave me a lot of valuable suggestions in my research and make me much more clear in my research field. They also teach me how to be a good PhD student.

Table of Contents

List of Figures

List of Tables

List of abbreviations

ACK   Acknowledgement message

AODV   On-demand Distance Vector

AP   Access Point

AWGN   Additive White Gaussian Noise

BER   Bit Error Rate

BS   Base Station

BPSK   Binary Phase Shift Keying

CBR   Constant Bit Rate

CDMA   Code Division Multiple Access

CEDAR   Core -Extraction Distributed Ad-hoc Routing

CFO   Carrier Frequency Offset

CoS   Cluster of Sub-carriers

CP   Cyclic Prefix

CPA   Channel Priority Assignment

CSMA/CA   Carrier Sense Multiple Access with Collision Avoidance

CTRMA   Concurrent Transmission or Reception Multiple Access

CTS   Clear-to-Send

DA   Data-Aided

DCF   Distributed Coordination Function

DFIR   Diffused Infrared

DIFS   Distributed Interframe Space

DS   Differentiated Service

DS	Dominating Set
DSDV	Destination Sequenced Distance Vector
DSR	Dynamic Source Routing
DSSS	Direct Sequence Spread Spectrum
FDMA	Frequency Division Multiple Access
FEC	Forward Error Correction
FFT	Fast Fourier Transfer
FHSS	Frequency Hopping Spread Spectrum
FSM	Finite State Machine
IARP	Intrazone Routing Protocol
ICI	Inter Carrier Interface
ID	Identification
IEEE	Institute of Electrical and Electronics Engineers
IERP	Interzone Routing protocol
IETF	Internet Engineering Task Force
IFFT	Inverse Fast Fourier Transfer
IP	Internet Protocol
ISI	Inter Symbol Interference
MAC	Media Access Control (layer)
MAI	Multiple Access Interference
MANET	Mobile Ad hoc Network
MM	Multiple-radio Multiple-channel
MPLS	Multi-protocol Label Switching
MT	Mobile Terminal

MUI	Multi-User Interference
NAV	Network Allocation Vector
NB	Neighbouring (node)
NCO	Number-Controlled Oscillator
NDA	Non-Data-Aided
NDM	network discovery message
NS	Network Simulator
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
OLSR	Optimized Link State Routing Protocol
PCF	Point Coordination Function
PDA	Personal Digital Assistant
PHY	Physical (layer)
PN	Pseudo-Noise sequence
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
ORSS	Overall Received Signal Strength
OSPF	Open Shortest Path First
RDIS	Route Discovery Message
REQ	Request (state)
RERR	Route Error
RF	Radio Frequency
RREP	Route Reply
RREQ	Route Request

RRES	Route Response Message
RSS	Received Signal Strength
RSSI	Received Signal Strength Percentage
RTS	Request-to-Send
SAS	Subcarrier Allocation Strategy
SIFS	Short Interframe Space
SIR	Signal to Interference Ratio
SM	Single-radio Multiple Channel
SNR	Signal to Noise Ratio
SREA	Sub-channel Re-allocation Message
SRRP	Sub-channel Re-allocation Reply Message
SSMAP	Signal Strength based Medium Access protocol
TBA	Ticket-based Probing
TCP	Transfer Control Protocol
TORA	Temporary Ordered Routing Algorithm
TTL	Time to Live
TDMA	Time Division Multiple Access
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunication System
W-CDMA	Wideband Code Division Multiple Access
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
WRP	Wireless Routing Protocol
ZRP	Zone Routing Protocol

Chapter 1

Introduction

1.1   Introduction to mobile ad hoc networks

With the development of the Internet, wireless communications have attracted more and more attention. In the last few years, many wireless communication technologies have emerged. These technologies provide users access to a network and communication anytime and anywhere. Among them, mobile ad hoc networks (MANETs) are one of the most interesting and challenging wireless networks because of their unique properties. A mobile ad hoc network is an autonomous system which only comprises of mobile wireless nodes. In contrast to other wireless networks, there is no fixed infrastructure in a MANET. All mobile nodes are self-configured and self-controlled. This type of network can be rapidly and easily set up, and deployed at any place without central administration. A detailed comparison table between a cellular and an ad hoc network is shown in table 1 below.

The distinct characteristics of ad hoc network from others have many advantages. First of all, it can avoid the cost, installation and maintenance of a network infrastructure.

Because there is no need for base stations, consequently this makes the network deployment much easier and more cost-effective. Secondly, ad hoc networks can be rapidly deployed and reconfigured. They can be formed from whatever wireless network nodes become available. This is especially important to specific applications such as in battlefield communications. Moreover, ad hoc networks are more robust due to their distributed nature and node redundancy. More than one route can be established simultaneously between two mobile nodes which prevail over single points-of-failure.

Table 1 Comparison of cellular and ad hoc networks

Cellular network	Ad Hoc wireless network
Infrastructure network	No infrastructure networks
Static network topology	Highly dynamic network topology
Fixed cell sites and pre-located base stations	No existence of a base station and rapid deployment
Mostly single-hop communication with base station	Single-hop or multi-hops communications between mobile nodes
Relatively normal environment and more stable connectivity	Hostile environment and irregular connectivity
Detailed planning	Automatically established and adapts to changes
High setup cost	Cost-effective
Large setup time	Short setup time

A commercial application, such as Bluetooth, is one of the recent developments utilizing the concept of ad-hoc networking. The IEEE 802.11x series based ad hoc networks are another alternative to Bluetooth but they are used in different markets. The 802.11x series can provide much higher data rate and larger coverage than Bluetooth. And bluetooth will be used to connect devices in a small area [1]. Another important commercial application for mobile ad hoc networks is WiFi direct [2] which is the technique that can enable two or more wireless devices directly connect to each other over longer distance and with high transmission speed.

1.2　Design and technical challenge for mobile ad hoc networks

Although the mobile ad hoc architecture has many benefits, such as self-reconfiguration and adaptability to highly variable mobile characteristics, it poses several technical and research challenges that need to be addressed [3]. The main challenges faced in an ad hoc network are described below:

- Security in an ad hoc network is one of the primary concerns in providing a protected communication between mobile users in a hostile environment [4]. The unique characteristics of the ad hoc networks pose many new nontrivial challenges in security design, such as open peer-to-peer network architecture, a shared wireless medium and rapidly changing network topology. These challenges raise the requirement of new security solutions to provide more powerful protection and maintain high system performance.

- Limited spectrum availability is another challenge in ad hoc networks. The external environment and interference from other mobile nodes can further influence the

bandwidth availability. How to efficiently distribute the limited network resources among all the mobile users to avoid collision is still an open issue.

- In most ad hoc networks, packets are forwarded from the source node to the destination node through multi-hop instead of single-hop routing. However, it is much more difficult to support high data rate and desirable end to end delay over multi-hop wireless channels than over single-hop wireless. This is a main challenge for applications with high data rate requirement and stringent end to end delay such as video, and voice conference.

- Routing is the determination and maintenance of a path between a pair of nodes to exchange information in a network. In an ad hoc network, the topology is constantly changing. It is much more challenging to provide a smooth transmission across the network. A path might be broken in an ongoing session because the movement of one or more participant nodes. This makes the route maintenance mechanism more important and difficult.

- Energy constrain is another big challenge in ad hoc networks [5]. All the mobile nodes can move freely, hence most of the mobile nodes are only powered by batteries which cannot be recharged. Therefore the power conservation is a key requirement which need to be taken into account in the design of an ad hoc network. Every mobile node in a MANET acts as a router in charge of forwarding the incoming packets to the next node towards the destination. So, most of the time the mobile nodes should be at least in the reception mode waiting for the incoming packets. This standby operation can consume significant energy. The network design should take this aspect into account to optimize the energy consumption.

- The scalability to larger networks is the main drawback because of the requirement of multi-hop traffic relay. In addition, the nodes in the ad hoc network need the help of other nodes to reach the correspondence over one hop away from them and it is unlikely that these intermediate nodes are willing to sacrifice their precious resource (such as power) for others without any common interest [6]. This could be the main restriction to the commercial success for ad hoc networks.

The physical time and clock synchronization are crucial in wireless communication so the nodes can successfully communicate with each other especially for real time services such as conference calls etc. However, there is no central controller in the ad hoc networks and they can be quite sparse in nature. Hence the traditional clock synchronization algorithms cannot be applicable in this setting. If OFDM modulation scheme is used in the physical layer, such as IEEE 801.11n [7], the synchronization becomes even harder because of the frequency offset in the physical layer. The mobile nodes by themselves must resolve both the time and frequency synchronization problems.

1.3   Motivation of this thesis

With the increasing interest in multi-media and other high bandwidth demanding applications, the provision of quality of service (QoS) support becomes one of the most important issues for MANETs system design. QoS is the performance level of a service provided by the network to the users [8]. The goal of the QoS provisioning is to guarantee the information transmission performance and optimise network resources utilization. A lot of efforts have been done in providing QoS support in both wired and

wireless networks. The QoS considered in this thesis is to find a route for a request session which satisfies the bandwidth and end-to-end delay requirements.

Compared to a wired network, QoS provisioning in wireless networks is more difficult because of the vulnerable and unstable features of the radio link and the mobility of mobile nodes, especially in mobile ad hoc networks. For mobile ad hoc networks, the unique features, such as no central coordination, limited network resource availability and hidden terminal problem [8], further complicate the QoS provisioning. Especially for MANETs, more consideration must be given to the MAC and routing problem when design to support real-time services which have high demands on end-to-end delay, jitter and bandwidth. In the past few years many QoS routing protocols have been proposed for MANET [9]. But most of them are based on the CDMA/TDMA contention free MAC protocol or on the contention based MAC protocol like IEEE 802.11 DCF.

Orthogonal Frequency Division Multiplexing (OFDM) is a digital modulation technique which has been widely adopted as primary physical layer technique in some of the major wireless standards (e.g. IEEE 802.11 [10], DVB-RCA [11], IEEE 802.16 [12], LTE-3GPP [13]). It divides the spectrum into a number of orthogonal parallel narrowband sub-carriers so that the symbol duration becomes longer than the channel delay spread. Consequently multipath effects are mitigated and wireless channel's multipath effects are alleviated. Due to its distinct advantages, an OFDM based multiple access scheme called OFDMA has been proposed as a medium access control scheme in many infrastructure based wireless networks, such as IEEE 802.16 [12], LTE-3GPP [13]. In OFDMA systems, sub-carriers can be further divided into many groups and each group is

called a sub-channel. These sub-channels can be assigned to different users and used at the same time without causing interference.

Recently, OFDMA has been considered to be used in ad hoc network and many proposals have already been published on implementing OFDMA in mobile ad hoc networks [14]-[17]. The sub-channel allocation to different applications in a distributed manner becomes an important research topic [14] [15]. Therefore developing an QoS routing protocol for OFDMA based ad hoc network becomes a very interesting and challenging topic. Interference avoidance and highly efficient frequency reuse of subcarriers are important research aims in the design of QoS routing protocols for OFDM/OFDMA based MANETs. A good routing protocol needs to consider both network layer and MAC layer, also the support from the physical layer in order to provide effective QoS support and bandwidth reservation especially for real-time services. Moreover, OFDMA requires restrict time and frequency synchronization to realize concurrent transmissions, especially in ad hoc networks. Therefore, developing an efficient time and frequency synchronization scheme is necessary.

1.4 Major contributions of this thesis

There are three main contributions in this thesis. First of all a novel signal strength based medium access protocol (SSMAP) is proposed to orchestrate the channel access in OFDMA based mobile ad hoc networks. SSMAP introduces a sub-channel selection criterion which is based on the interference power of the sub-channels in the receiver side aiming to reduce the co-channel interference and the signalling overhead. The proposed sub-channel allocation scheme is fully distributed and each node does not need to exchange periodic update information with its neighbours.

Secondly, a cross layer SSMAP based QoS routing protocol is proposed. The detailed route discovery algorithm as well as route maintenance scheme are described. With SSMAP, the collision-avoidance handshake (RTS/CTS) is no longer needed and the signalling overhead can be largely reduced. Moreover, with proposed sub-channel allocation scheme, each node can have multiple concurrent transmissions and receptions which can largely increase the system throughput.

Furthermore, to prove the feasibility of the proposed QoS routing protocol, a partial time synchronization scheme and a physical layer frequency synchronization scheme are proposed. With the partial time synchronization scheme, each node can successfully support concurrent transmissions. The proposed frequency synchronization scheme utilizes the null subcarriers inserted between adjacent sub-channels to estimate and compensate the frequency offsets. A system architecture of the receiver is proposed with reasonable synchronization complexity compared to other synchronization schemes in infrastructure based wireless networks.

1.5   Organization of this thesis

The rest of the thesis is organized as the following. Chapter 2 firstly describes the concept of OFDM and OFDMA, including the principles and implementation. Secondly some recent publications on implementing OFDMA in ad hoc networks have been described and analysed.

In chapter 3, the most popular routing protocols for ad hoc networks are described. Then QoS is discussed and the challenges for QoS support in ad hoc networks are presented. Finally some of the most important QoS routing protocols found in the research literature are classified and described in detail.

Chapter 4 presents the proposed signal strength based medium access protocol. A SSMAP based QoS routing protocol is also described in detail including both the route discovery process and the route maintenance scheme.

Chapter 5 describes the detailed proposed timing and frequency synchronization schemes including the system architecture and signal model. The synchronization scheme is to further support the proposed SSMAP based routing protocol from the physical layer point of view. A substantial literature review of OFDM and OFDMA based timing and frequency synchronization proposals is presented as well.

Chapter 6 implements the simulation model of the SSMAP based routing protocol developed using OPNET Modeller. The modelling design is presented in details from the network model to process model. The modelling of the routing protocol is validated step by step using OPNET tracing file. The developed SSMAP based routing protocol is fully evaluated through simulations in different scenarios in chapter 7. Conclusion and future work is presented in chapter 8.

Chapter 2

Implementing OFDMA in Ad Hoc networks

2.1 OFDM

The orthogonal frequency division multiplexing (OFDM) technique is a method in which an original data stream is split into multiple low frequency data streams and transmitted simultaneously. The technology was first conceived in the 1960s and 1970s during research into minimizing interference among channels near each others in frequency [99]. The key idea behind this is the orthoganality of the carrier frequency. In normal frequency division multiplexing (FDM) system, the concept of single carrier modulation is extended by using multiple subcarriers. These subcarriers in the frequency domain should not overlap with each other to avoid interference (Figure 1).

If the FDM system above uses a set of subcarriers which are orthogonal to each other, then the subcarriers' spectrums can overlap without causing interference. This is the

basic principle of OFDM modulation. It can be realised by splitting the original data stream into several low data rate sub streams and each sub stream is mapped to an orthogonal frequency basis. Finally these modulated sub streams are added together to form an OFDM signal. The Figure 2 (a) shows a simple representation of an OFDM system. The high data rate baseline stream is divided into many sub streams and each of them is multiplied by an orthogonal frequency basis *Fn*. Finally, all the sub streams are added together to form a low data rate OFDM data stream.
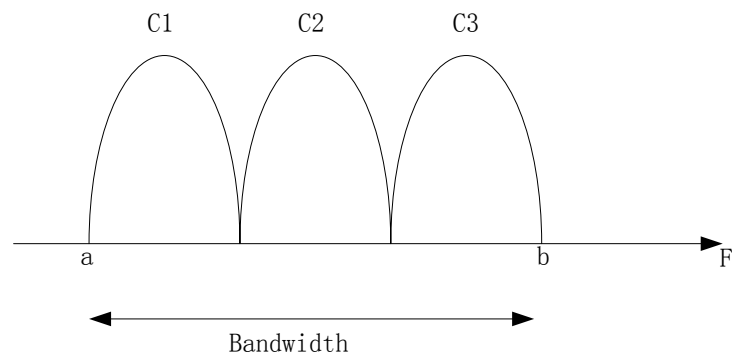
Figure 1 FDM frequency domain subcarrier division
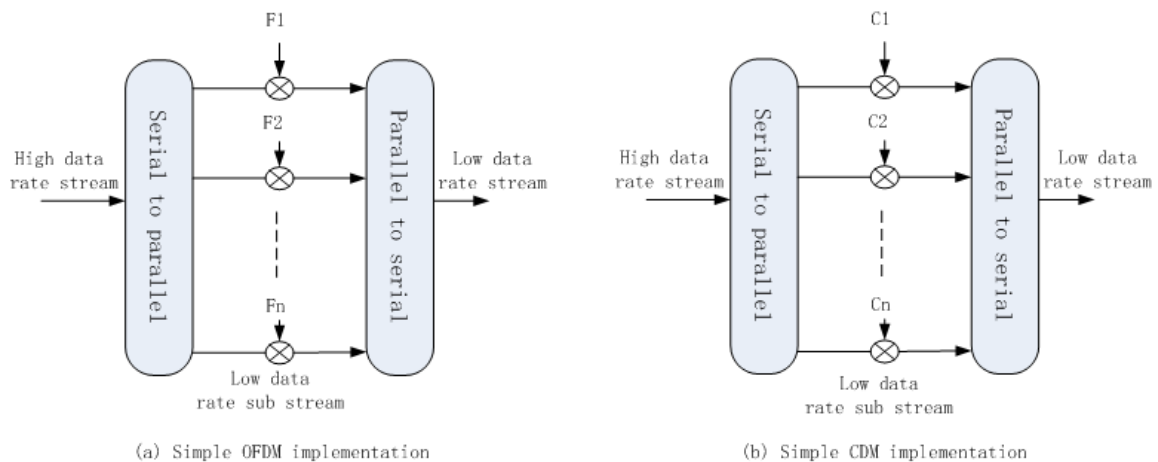
Figure 2 Similarities of OFDM and CDM implementations

The orthogonality of the subcarriers allows simultaneous transmission on multiple subcarriers in a tight frequency band without interfering with each other. This is very similar to the code division multiplexing (CDM) modulation (Figure 2 (b)). CDM is a networking technique in which multiple data signals are combined for simultaneous transmission over a common frequency band. When CDM is used to allow multiple users to share a single communications channel, the technology is called code division multiple access (CDMA). Figure 3 (a) shows a resultant frequency spectrum after OFDM implementation with four subcarriers. $X(n)$ is channel bandwidth, and $z1(n), z2(n),$ $z3(n), z4(n)$ represent the modulated subcarriers. In the frequency domain as shown in Figure 3(a), the subcarriers are overlapping with each other. However, in the peak point of each subcarrier, the signals from other sub-carriers are zero. Therefore this overlapping will not affect the system recovery of the individual original signals. The subcarrier overlapping in the frequency domain can largely increase the overall system spectrum efficiency. The Figure 3 (b) illustrates the time domain OFDM signal. Because the OFDM signal is the combination of many subcarriers, it looks like noise in the time domain.
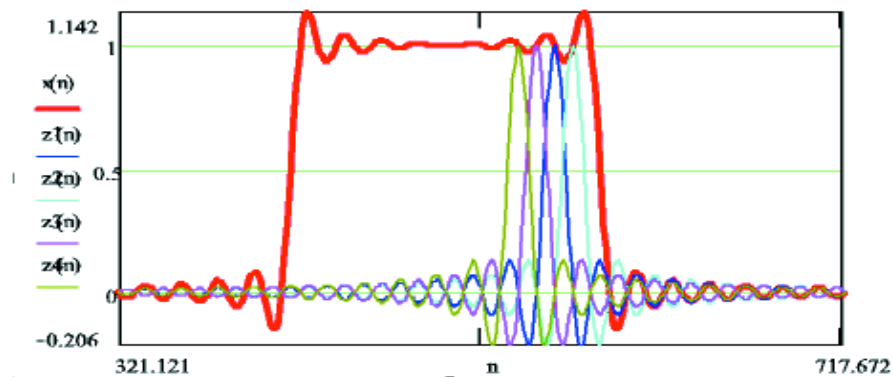
(a) Frequency domain OFDM signal [18]
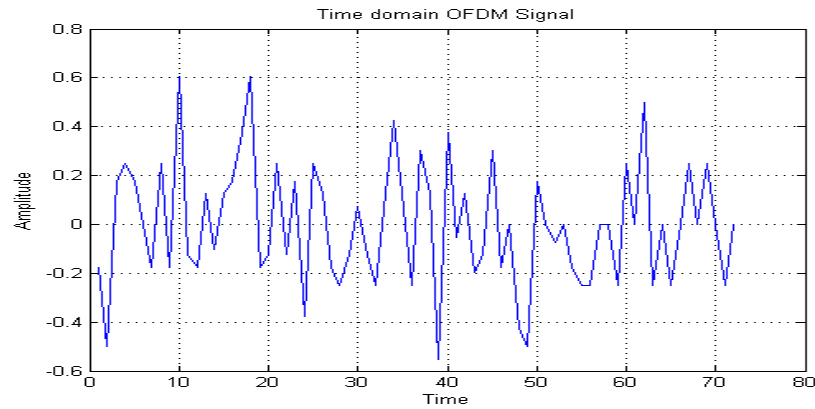
(b) Time domain OFDM signal

Figure 3 Frequency and time domain OFDM signal

In practice, IFFT is used to realise fast frequency mapping in OFDM implementation. The sinusoids of the IFFT form an orthogonal basis set which can be used as OFDM subcarrier frequency basis. The transform of IFFT is used to map the incoming signals to a set of orthogonal frequency subsets which is represented by the OFDM subcarriers. Similarly, in the receiver side, FFT is implemented to transfer the OFDM signal back to the original data stream.
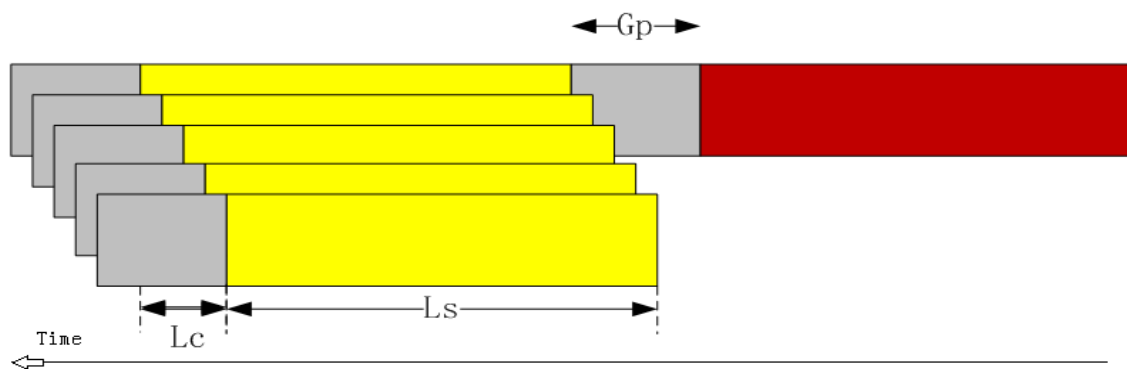
Figure 4 Inter symbol interference in OFDM

2.1.1 Advantages and disadvantages of OFDM

There are three main benefits from an OFDM operating system.

(1) A major problem in most wireless communication systems is the presence of a multi-path channel which results in multiple delayed versions of the transmitted signal at the receiver side. In a single-carrier system, the received symbol probably would be distorted by the delayed copies from the previous symbol which is called inter-symbol interference (ISI). However, in the OFDM modulation system, a high speed serial data is split into N low data rate streams which mean the OFDM symbol period is increased by a factor of N compared to the original serial data. In this case, the OFDM symbol duration Ls becomes much longer than the channel delay spread Lc which makes the effect of ISI neglectable. Figure 4 illustrates the ISI problem in an OFDM system, where Lc represents the channel delay spread and Gp is the guard time interval. Ls is the length of one OFDM symbol. Note that only the first Lc sample of the received OFDM symbol is affected. By introducing a guard interval Gp between contiguous OFDM symbols (as shown in Figure 4), the ISI can be easily removed.

(2) An OFDM signal can perfectly resist frequency selective fading channel. Because an OFDM symbol consists of many sub-carriers, in a frequency selective fading environment, instead of the whole symbol being corrupted, only a few sub-carriers will be affected. With proper coding, this can be recovered.

(3) Because of the orthogonality of the sub-carriers in a OFDM symbol, they can overlap without causing problems for the correct reception of each subcarrier in the receiver.

So eventually the bandwidth usage efficiency can be largely increased as can be seen from Figure 3.

Although OFDM have many distinct advantages comparing to the other technologies, it has some weaknesses too. First of all the OFDM signal is the combination of many low frequency subcarriers. It has a noise like amplitude with a very large dynamic range in the time domain. As it can be seen from Figure 3 (b), at some point, many subcarriers may all contribute to the OFDM signal which results in very high amplitude at that time compared to other instants of time. Therefore, it requires RF power amplifiers with a very high peak to average power ratio. This can largely decrease the RF power amplifiers efficiency [19]. Because when the signal power is larger than a threshold, the output power from the amplifier is not linearly proportional to the input power. To amplify the OFDM signal without distortion, it has to make sure that the output power of the OFDM signal peak point is not lager than the threshold. In this case, the degree of the amplification cannot be high.
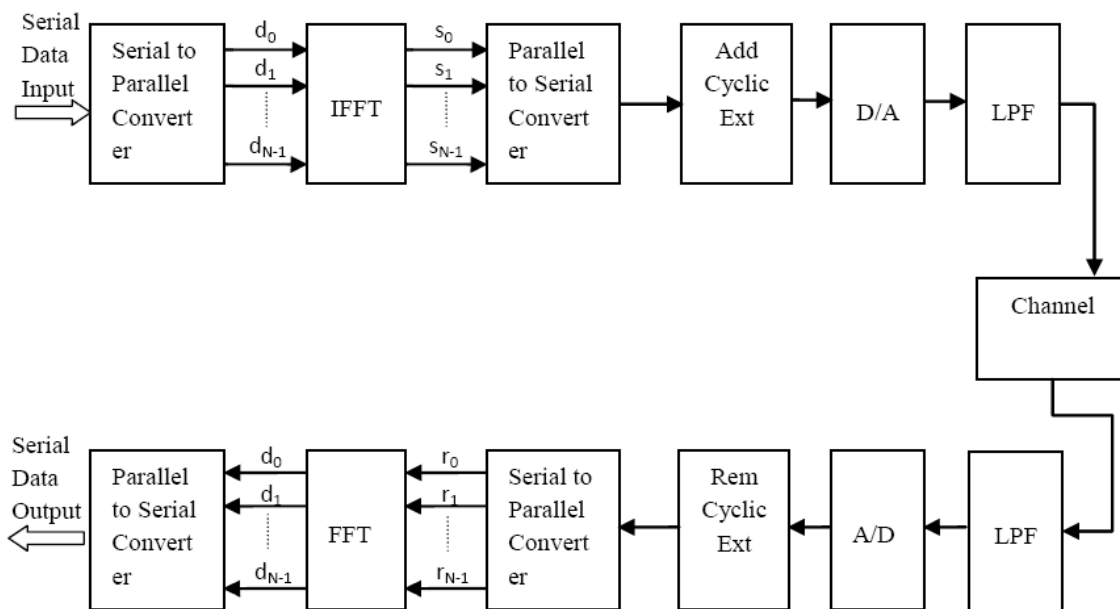
Figure 5 Flow chart of the OFDM system diagram   31

Another disadvantage for the OFDM technique is that, the power consumption is higher than other techniques such as CDMA or TDMA. So, for the mobile users that have limited power supply, it is a greater challenge to extend the battery life. Furthermore, the OFDM signal is more sensitive to the carrier frequency offset than single carrier systems [19].

2.1.2   The implementation of an OFDM system

The basic functional block diagram for the implementation of an OFDM system is shown in Figure 5. This block diagram depicts how the signal is modulated, transmitted and demodulated. First of all, the original serial data is partitioned into K parallel data streams from d_0 to d_{N-1}. An Inverse Fast Fourier Transform (IFFT) is applied to the K parallel data segments and finally combined together to become one OFDM symbol. The IFFT is one of the key components of OFDM, it was first proposed by Weinstein and Ebert in 1971. It correlates the input parallel data with its orthogonal basis functions. This correlation can be seen as mapping the input data on to relative orthogonal sub-carriers. An OFDM symbol contains K subcarriers modulated by N parallel data streams. Each subcarrier can be written as:

$$C_k(n) = e^{j2\pi n f_k} \tag{2.1}$$

Where f_k is the frequency of kth subcarrier. The subcarrier frequency f_k is equally spaced so that all subcarriers are orthogonal to each other.

$$f_k = \frac{k}{S_T} \tag{2.2}$$

Where S_T is the duration of the OFDM symbol. The resultant OFDM signal after IFFT can be written as [20]:

$$S(n) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} d_k C_k(n) \qquad 0 \leq n \leq N-1 \qquad (2.3)$$

Where d_k is the complex data symbol which will be transmitted in subcarrier k. N is the number of subcarriers. Replacing equations (2.1) and (2.2) into (2.3), the final output OFDM symbol can be written as [20]:

$$S(n) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} d_k \, e^{j 2\pi \frac{nk}{N}} \qquad 0 \leq n \leq N-1 \qquad (2.4)$$

After the IFFT, a cyclic extension is added to the created OFDM symbol. In practice, the front of one received OFDM symbol probably will be distorted by the delayed copy of the previous OFDM symbol. To mitigate this noise at the front of the symbol, a guard interval is inserted which is longer than the time span of the channel for each OFDM symbol $G_p > L_c$ as described in Figure 4. However, if a blank space is inserted as the guard interval for each OFDM, the problem of inter-carrier interference (ICI) will arise. This is because in order to successfully recover the OFDM signal in the receiver side using FFT, there must be integer number of cycles difference between sub-carriers within the FFT interval in the receiver part. So in this case the receiver cannot recover each sub-carrier from an OFDM symbol. To avoid this problem, the guard interval is replaced by a *cyclic prefix* in front of each OFDM symbol. This so called *cyclic prefix* is a replica of the last samples of the OFDM symbol which makes the OFDM symbol appear periodic. The cyclic extension actually is redundant information and will be removed at

the receiver. So like the case of the guard interval mentioned above, the ISI can be avoided as well.

After that, the OFDM signal passes through a digital-to-analogue transformer which maps the OFDM signal to higher carrier frequency as shown in Figure 6, where F_c is the carrier frequency and F_o is the OFDM signal frequency. Then this signal will passes through a low pass band filter that removes the higher frequency component $(F_c + F_o)$.

On the receiver side, the reverse functional blocks will process the received signal so that the original serial data can be recovered at the end.
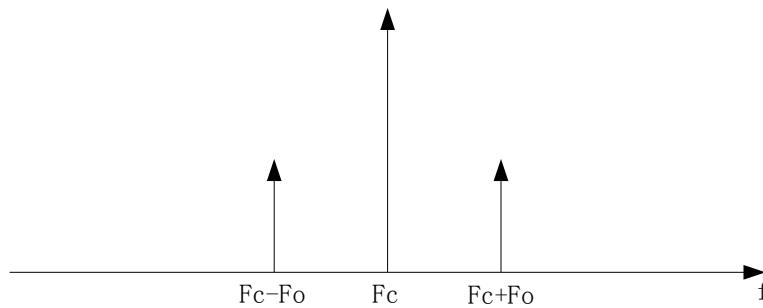
Figure 6 Frequency hopping for OFDM signal

2.2   OFDMA

Orthogonal Frequency-Division Multiple Access (OFDMA) is a multi-user version of OFDM which can support multiple user transmissions. In OFDM, a single user can occupy the entire bandwidth by using all subcarriers, while in OFDMA the subcarriers are divided into several subsets which can be assigned to individual users. Moreover, in many cases, the OFDMA is used along with the other multiple access techniques, such as TDMA. The OFDMA symbol is scheduled by TDMA and the subcarriers in OFDMA

symbol are allocated by OFDMA. An example of OFDMA/TDMA combination is showed in Figure 7. OFDMA/TDMA can make the subcarrier allocation scheme more flexible.
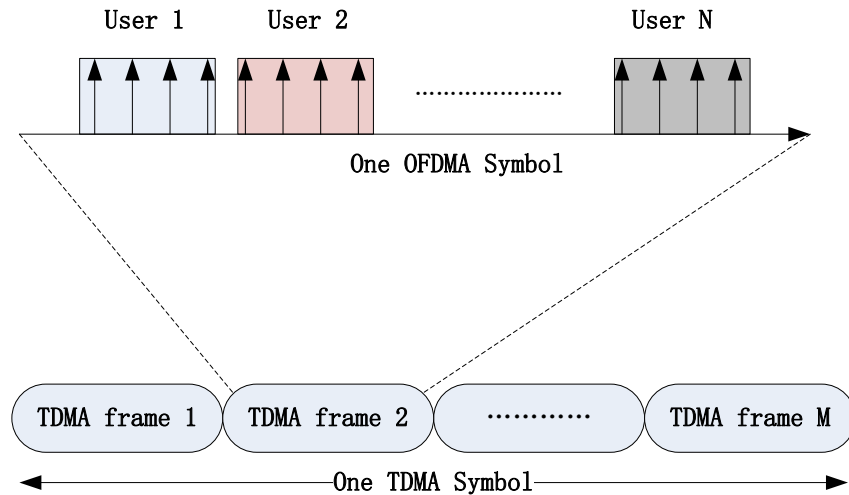
Figure 7 Combination of OFDMA and TDMA

Currently distributing the subcarriers over the sub-channels is a very open problem and many permutation modes in OFDMA have already been proposed. There are three possible subcarrier allocation schemes (SAS). They are illustrated in Figure 8. Figure 8 (a) is called s*ubband SAS* where each sub-channel consists of adjacent subcarriers. This type of SAS makes the sub-channel estimation much easier and the users can choose the part of the bandwidth presenting the best conditions at the moment of transmission. Accordingly the overall system throughput can be increased. Another SAS is the *interleave SAS* where the subcarriers of each user are uniformly spaced over the signal bandwidth at a distance R from each other. Although this method can fully exploit the channel frequency diversity, the current trend in OFDMA favours a more flexible allocation strategy where users can select the best subcarriers. An example is shown in Figure 8(b). The last SAS is called *generalized SAS* as shown in Figure 8(c). In this scheme,

35

the sub-carriers are divided into several sub-channels which consist of sub-carriers randomly distributed over the entire bandwidth which is more flexible than the others. Each sub-channel can select the sub-carriers based on their current channel conditions.

On the other standards, the subcarrier permutation is defined in a more specific way. For example in IEEE 802.16e, the subcarrier distribution modes can be classified into two main categories, *diversity* or *distributed permutations* and *contiguous* or *adjacent permutations* [21]. The diversity mode includes FUSC (Full Usage of the Sub-channels), PUSC (Partial usage of the sub-channels) modes and an example in contiguous mode is AMC (Adaptive Modulation and Coding) [21].
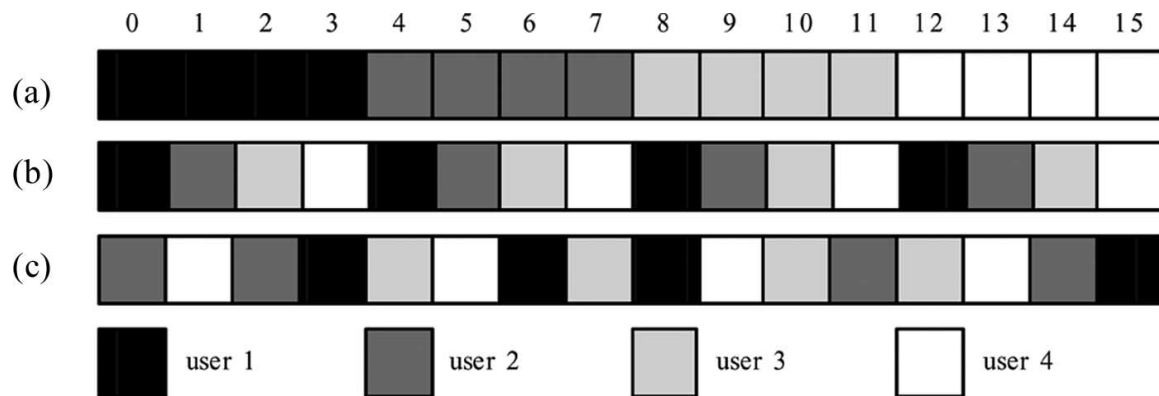
Figure 8 Three possible subcarrier allocation schemes

2.3 Implementation of OFDMA in ad hoc networks

OFDMA has been widely used in various wireless mobile network systems, such as 3GPP LTE and WiMAX. However, there is less published literature on implementing OFDMA in ad hoc networks. This is because in ad hoc networks, without a central coordinator, the resource allocation becomes much more difficult than in infrastructure based wireless

networks. In this section, some recent proposals about implementing OFDMA in ad hoc networks are explained and compared.

In [22], the authors propose an enhanced distributed coordination function (DCF) based on the IEEE 802.11 DCF to implement OFDMA for ad hoc networks. The aim of this proposal is by using OFDMA to realize concurrency in the physical layer and increase the capacity of the system when the number of contending devices is high. The main idea behind the proposed DCF is that the nodes are divided into several groups and one sub-channel is assigned for each group. The contention avoidance mechanism (RTS/CTS) will take place in each group respectively through the assigned sub-channel. The contention cycle ends after each group has transmitted an RTS or a specified timeout occurs. The recipients respond with CTS using the sub-channel at which they have received their RTS message. In each group, there is only one CTS message at a time. Then the nodes divide the channel into equal sub-channels based on the number of CTS messages of the whole network and start to transmit data in parallel using the assigned sub-channel.

The algorithm can be described as follows:

1. Every node in an ad hoc network periodically broadcasts a called *network discovery message* (NDM) that contains information about the sub-channel it is using, along with other parameters.

2. When a new node joins the network, it can transmit a *Hello* message. Other devices respond with an NDM. The new device can then select the least used sub-channel. The whole procedure will be repeated once the topology changes in the network.

3. Nodes in the same group share the same frequency sub-channel and contend among themselves to capture the channel using the same mechanism in IEEE 802.11 DCF.

4. After transmitting a RTS, the sender waits for a CTS. Other nodes in its group halt their backoff until the next contention cycle starts. So, there can be no more than one successful RTS sent from each group in a contention cycle.

5. When in backoff, nodes can listen for the channel. If a node receives an RTS message and detects that its intended receiver has transmitted that message, it halts its backoff and waits for the next contention cycle. Alternatively, if a device receives an RTS with its own address in the destination address field, it halts its backoff counter and replies with a CTS message when the contention cycle is over.

6. The contention cycle ends as soon as either each group has sent one RTS, or a timeout occurred. No device will initiate a new RTS after the contention cycle is over, only the ongoing RTS transmissions will continue. When the contention cycle is over, nodes that have received an RTS message wait for the channel to remain idle for a *Short Inter Frame Spacing (SIFS)* time which is defined in IEEE 802.11 [99] and then transmit a CTS using the same sub-channel at which they have received that RTS. *SIFS* normally corresponds to the time that is required to switch the radio of a device between the reception and transmission.

7. Devices that have received a CTS message, calculate a sub-carrier assignment in a distributed way. They start transmitting data using the calculated sub-channels after waiting for a SIFS time. Devices can detect a collision by the absence of a CTS in response to its RTS. Then they wait for the beginning of another contention cycle using

the same binary exponential backoff rules used by IEEE 802.11 DCF. Finally, after waiting for a SIFS time, the receiving devices transmit an ACK using the same sub-channels at which they have received data.

Although the author in [22] tried to propose a DCF to avoid collision and realize concurrency in an ad hoc network, the explanation of the sub-channel allocation algorithm is missing in the paper. Moreover, by using RTS/CTS mechanism in each group will significantly increase the system signalling overhead and reduce the overall throughput.

In [23], the authors propose a link-oriented resource allocation algorithm for OFDMA based ad hoc networks. The main aim of the proposed scheme is to guarantee fairness in the resource allocation by maximizing the spatial reuse rate, which means trying to use the same network resources as many times as possible. The basic idea for this algorithm to realize fairness is to allocate a given network resource every three-hop. One of the main drawbacks for this proposal is that by reusing the network resources as many times as possible will significantly increase the overall network interference and reduce the system throughput. Moreover, in order to realize the optimal spatial reuse rate, each node in the network has to periodically exchange local information with other nodes to make the decisions on the best resource allocation. This will largely increase the system signalling overhead.

Veyseh et. al. in [24] propose a Concurrent Transmission or Reception Multiple Access (CTRMA) protocol to allocate sub-channels in ad hoc networks. In their system model, each node is endowed with a single half-duplex radio and a single antenna. The aim of the proposed scheme is to develop a sub-channel allocation algorithm that takes

advantage of the ability of OFDMA to support concurrent transmissions. The advantage of the proposed scheme is that it can support dynamic bandwidth selection and enhance the channel reuse.

CTRMA consists of three main components: (a) using OFDMA at the physical layer, (b) a channel priority assignment (CPA) algorithm, and (c) a joint negotiation algorithm. OFDMA enables the concurrent transmission or reception of multiple packets by a given node. CPA is responsible for allocating high-quality channels to each transmitter-receiver pair so as to avoid multiple access interference, and it is executed when topology changes occur in the neighbourhood. The joint negotiation algorithm exploits the priorities assigned with the CPA to negotiate best channels immediately prior to data transmission. It improves throughput by reducing overhead and establishing multiple transmissions via a single round of control-message exchange.

The CPA algorithm assigns a channel priority to each channel for each link when a link is defined for a pair of nodes that are immediate neighbours. A channel priority has two values, *high (h)* and *low (l)*. A priority value of *high* denotes a low collision probability and a priority value of *low* means a high collision probability for the channel being utilized on the link. A channel with *high* priority on a link *u* must be assigned a priority *low* on all links within *k* hops away from the link *u*. *k* should be at least equal to 2 to avoid co-channel interference. Any channel assigned *high* priority can be utilized by the link. In CTRMA, a node can utilize multiple sub-channels concurrently which are assigned with a priority of *high* on a link to provide adaptive bandwidth selection for each transmission. By assigning priorities of all sub-channels for each link, the sub-channels

can be allocated among networks in a fully distributed way.

The joint negotiation process is encapsulated in the RTS/CTS mechanism. The handshake is initiated by the transmitter. A transmitter would need to send a Request-To-Send (RTS) message on the dedicated control channel and if successful, all neighbouring idle nodes would be able to receive the message. The RTS contains a list of the targeted receivers and the selected corresponding channels. If neighbours confirm the selected channels, they reply on the control channel with a Clear-to-Send (CTS) message at the scheduled delayed time assigned by the transmitter to make sure no collision occurs. In here, perfect time and frequency synchronization is assumed. For the sake of synchronization each RTS includes a time reference clock indicating the beginning of the next time slot set by the transmitter. If a receiver confirms the selected channels, it includes the same reference clock in the CTS message. Any potential neighbour that receives the CTS, would be able to find the beginning of the next time slot adjusted according to the reference clock of the two-hop neighbouring transmitter.

Although CTRMA can improve the system throughput by assigning a priority to each link to avoid collisions, it has many drawbacks. Firstly, to set the channel priorities for each link, each node must periodically exchange updated information with its one-hop neighbours, which will largely increase the network signalling overhead. Moreover, time slots are used in CTRMA to schedule the concurrent transmissions. This is based on the assumption that the network is perfectly synchronized by time which is one of the most difficult tasks in ad hoc networks. However, the time synchronization issue has not been addressed in this paper. Based on these drawbacks, this thesis proposes a more

effective MAC layer protocol which is aiming to improve the system throughput and reduce the overall signalling overhead. The OFDMA based medium access scheme proposed in this thesis will be later compared with the scheme described in [24] in terms of signalling overhead, system throughput and session success rate.

2.4 Concluding remarks

This chapter presents the fundamental concepts of OFDM and OFDMA. Section 2.1 presents the OFDM signal in both time and frequency domain. The advantages and disadvantages of OFDM are also described in detail. Finally, the implementation of an OFDM system is presented and explained in detail.

In section 2.2, the concept of OFDMA is described. Three common subcarrier allocation schemes (SAS) are illustrated which are called subband SAS, interleave SAS and generalized SAS respectively.

Section 2.3 presents the recent research publications on implementing OFDMA in ad hoc networks. The benefits and disadvantages of each presented scheme are described and analysed. The major published related work described in this chapter, the CTRMA protocol, is used as basis of comparison to the OFDMA based MAC layer protocol proposed in this thesis. The comparison will be shown in chapter 7.

The next chapter presents the classic routing protocols proposed for mobile ad hoc networks.

Chapter 3

Routing Protocols in Mobile Ad Hoc Network

3.1 Introduction

With the advances of wireless communication technologies, mobile networks have attracted significant attention in recent years. Compared to wired networks, node mobility in mobile networks may cause frequent topology changes which is rare in wired networks. Moreover, in contrast to the stable link capacity of wired networks, wireless link capacity is vulnerable to many factors such as transmission power, receiver sensitivity, fading, interference, and noise.

Mobile nodes in MANETs are autonomously self-organized without any infrastructure support. The arbitrary movement of nodes results in rapid and unpredictable topology changes. Additionally, since the wireless link normally has limited coverage, some nodes cannot communicate directly with each other. Therefore, an ad hoc network should be considered as a multi-hop network and each node in a mobile ad hoc network acts as

both host and router. All the characteristics lead to the requirement to build a new ad hoc suited routing protocol which is different from the traditional ones implemented in the current internet environment [25] [26].

As a promising wireless network for the future, MANETs have attracted more and more attention of researchers. Researches on MANETs have proposed many new solutions for better resource management, multiple access methods, and routing. In the last few years, many routing protocols have been proposed for dynamic multi-hop networks. In this chapter, some of the most classic routing protocols for mobile ad hoc networks are reviewed.

3.1.1   Classification of Routing Protocols

Routing is a fundamental issue for networks. Prior to the increased interests in wireless networks, a lot of routing algorithms have been proposed for wired networks. Among them, the Distance Vector routing [27] and the Link State routing [27] are two of the most popular routing algorithms in wired networks.

In the distance vector routing protocol, every router stores the distance information to all reachable destinations in a local routing table. A router periodically exchanges the distance information with its neighbours to update its routing table. The metrics used to calculate the distance can be hop number, queue size and delay. The routing Information Protocol (RIP) [28] is based on the distance vector routing.

In the link state routing, each node maintains an up-to-date view of the network by periodically broadcasting its current status of links to all routers in the network. If a link state change occurs, the related notifications will be flooded through the whole network

[27]. When each node receives the update information, it will recalculate the routes and choose the next-hop node for each destination by applying the shortest-path algorithm such as the Dijkstra's algorithm. Open Shortest Path First (OSPF) [29] is an example that utilizes the link state routing protocol.

The traditional distance vector and link state routing algorithms are not suitable for mobile ad hoc networks. This is because in mobile ad hoc networks, the network topology changes frequently due to node mobility. Frequent route updates which will consume a significant part of network resources and increase the channel contention. To overcome these potential problems, a number of routing protocols have been proposed for MANETs. These routing protocols can be classified into several types based on different criteria, such as network topology, and route discovery protocol. Normally, these routing protocols can be categorized into three major groups: global or proactive, on demand or reactive, and hybrid (table 2).

Routing protocols for mobile ad hoc networks		
Proactive	**Reactive**	**Hybrid**
DSDV, WRP, LOSR etc.	DSR, AODV etc.	ZRP etc.

Table 2 Classification of the routing protocols in mobile ad hoc networks

- In *proactive routing protocols*, every node in the network maintains up-to-date routes to all possible destinations by periodically broadcasting update information. When a network topology or link state change occurs, related notifications will be flooded throughout the whole network. Most proactive routing protocols for MANETs are deriving from the traditional routing algorithms [9] used in wired networks with necessary modifications to adapt to

the dynamic features of MANETs. Because every mobile node proactively maintains routes to all destinations and periodically broadcasts update information, the overhead to maintain the up-to-date network state is high. However, the proactive routing protocols save the time for route discovery, and consequently a mobile node can start to forward the packets to the destination as soon as a session is triggered based on its up-to-date routing table. The typical proactive routing protocols are the Destination Sequence Distance Vector (DSDV) [30] and the Wireless Routing Protocol (WRP) [31].

- *Reactive routing protocols* for MANETs are also called on-demand routing protocols. In a reactive routing protocol, routes are discovered only when needed. When a session is triggered in a mobile node, it invokes a route discovery process. Then a route request message is flooded from the source node to the destination. Once the destination node receives the route request message, it initiates a reply message and sends it back to the source node along the reverse path of the route request. The discovery process terminates either when a route is found or there is no route available. Once a route is established, a route maintenance procedure will be invoked until the end of the session. Compared to proactive routing protocols, the reactive routing protocols have less control overhead and better scalability. However, using reactive routing protocols, source nodes may suffer longer delay than proactive routing protocols for route discovery before they can start to forward the packets. Examples of on-demand routing protocols include Dynamic Source Routing (DSR) [32] and Ad hoc On-demand Distance Vector routing (AODV) [33].

- *Hybrid routing protocols* are proposed to overcome the shortcomings of both proactive and reactive routing protocols by combining their merits. Normally, hybrid routing protocols for MANETs exploit hierarchical network architecture and proactive and reactive routing protocols are implemented in different hierarchical levels. The Zone Routing Protocol (ZRP) [34] is typical example of hybrid routing protocol. It divides the network into several routing zones. Proactive routing protocols are implemented within each zone while reactive routing protocols are used for route discovery across zones.

3.1.2 Proactive routing protocols

3.1.2.1 Destination Sequenced Distance Vector (DSDV) routing protocol

DSDV [30] is one of the earliest proactive uni-cast routing protocols developed for mobile ad hoc networks. DSDV uses Bellman-Ford Distance Vector routing algorithm with some necessary modifications to adapt to the dynamic characteristics of MANETs, increase the reliability of update information exchange and avoid formation of route loops.

In DSDV, every node has a routing table which stores the next hops toward to each possible destination, the cost metric for the routing path to the destination which is represented by the number of hops for that destination and a destination sequence number for every destination. For example, as shown in Figure 9, the network contains 8 mobile nodes. Originally, N_1 was connecting with N_2. Table 3 shows a possible routing table maintained in N_4 before N_1 is moving away.

The route update in DSDV can be either time-driven or event-driven. DSDV requires each node periodically broadcasts updates about its routing information to its neighbours so that every node in the network can keep an up-to-date view of the whole network. In this way, routes to all destinations are available at any time in each node, and every node can start to forward packets as soon as a session is requested. A destination sequence number is created by each destination to show the freshness of a route. The route with the higher sequence number is considered the fresher one and if two routes have the same sequence number, the route which has the lower distance is favoured.
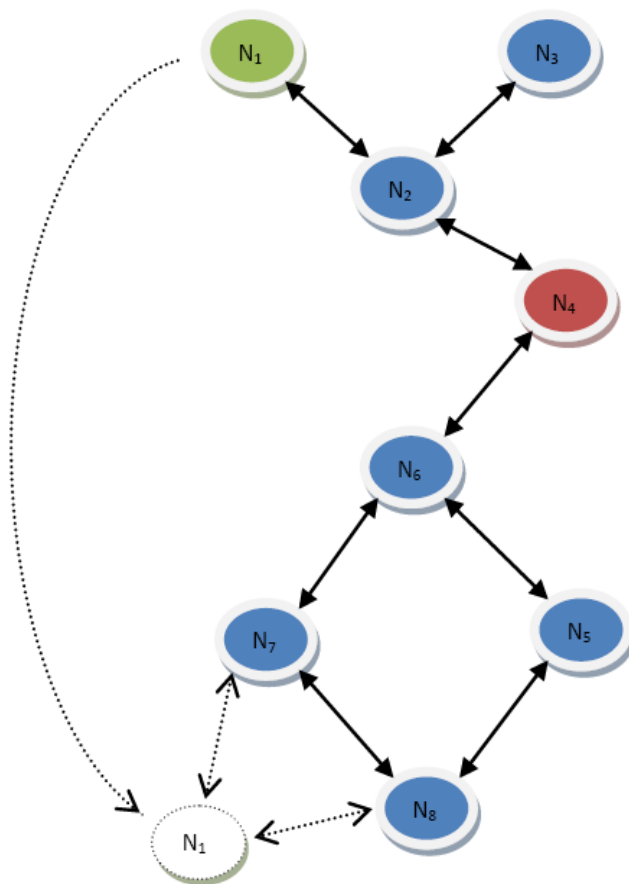
Figure 9 Example of DSDV in mobile ad hoc network

When a breakage is detected by a node, for example when a node moves out of the range of another, the node will set the distance to the affected destination to infinite (∞) and broadcast the related notification to all its neighbours. Any node who receives this notification will update its routing table and forward it to its neighbours again so that this information can propagate throughout the whole network. For example, assuming that N_1, in Figure 9, is moving away from N_2 and entering into the vicinity of N_7 and N_8. An immediate routing information update is triggered and the new routing information is broadcasted throughout the whole network. N_4, upon receiving this new update information, would then update the local routing table correspondently. All the information kept in the routing table will be recalculated. The new routing table at N_4 will appear as shown in table 4. Note that, the metric to the destination 1 now is changing from 2 to 3. The sequence number is increased to 36.

Table 3 Routing table kept in N_4 before N_1 moves

Destination	Next Hop	Metric	Sequence number
N_1	N_2	2	32
N_2	N_2	1	45
N_3	N_2	2	67
N_4	N_4	0	84
N_5	N_6	2	125
N_6	N_6	1	214
N_7	N_6	2	353
N_8	N_6	3	421

Table 4 Routing table kept in N_4 after N_1 moves

Destination	Next Hop	Metric	Sequence Number
N_1	N_6	3	36
N_2	N_2	1	45
N_3	N_2	2	67
N_4	N_4	0	84
N_5	N_6	2	125
N_6	N_6	1	214
N_7	N_6	2	353
N_8	N_6	3	421

DSDV is a modified distance vector with triggered updates to adapt to the dynamic topology in MANETs. The route updates significantly increase the network overhead and consume the limited bandwidth which is precious network resources in MANETs.

3.1.2.2 The Wireless Routing Protocol (WRP)

The wireless Routing Protocol (WRP) [31] is also a proactive unicast routing protocol for mobile ad hoc networks. Just like DSDV, WRP is also based on Bellman-Ford Distance Vector routing algorithm. However, different mechanisms are used in WRP to improve routing performance in MANETs.

In WRP, each node maintains a routing table, a distance table, a link-cost table, a Message Retransmission List (MRL) and an ack-status table. The routing table contains a distance to a destination, the predecessor and the successor nodes along the path to the destination. Storing predecessor and successor in the routing table can effectively

avoid loop and count-to-infinite problem which is the main shortcoming for the original distance vector routing protocol [31]. The link-cost table consists of an entry for each neighbour which contains the cost of the link to the related neighbour and the number of timeouts since an error-free message was received from those neighbours.

In WRP, each node exchanges its routing table with its neighbours by using update messages. The routing update information exchange can be triggered either periodically or whenever a link state change occurs. MRL records every neighbour node which does not reply to the update message. If necessary the update messages will be retransmitted to the related neighbours. Upon receiving the update information, a node will recalculate the route and check if there are better route paths, then it will record to the distance table. The biggest shortcoming of WRP is that it consumes large memory storage and compute resources to maintain several tables in each node. As a proactive protocol, it has poor scalability and large control overhead which make it not suitable for large MANETs [35].

3.1.2.3 Optimized Link State Routing Protocol (OLSR)

The Optimized Link State Routing Protocol (OLSR) [105] is developed for mobile ad hoc networks. It operates as a table driven and proactive protocol, thus exchanges topology information with other nodes of the network regularly. Other than in a pure link state protocol where all the links with neighbour nodes are declared and are flooded in the entire network, OLSR protocol is an optimization of a pure link state protocol for mobile ad hoc networks.

The basic idea of OLSR is using so called *multipoint relays* to forward the control messages. Each node in the network selects a set of nodes in its neighbourbood, which retransmits its packets. This set of selected neighbour nodes is called the multipoint relays (*MPRs*) of that node. The neighbours of any node *N* which are not in its *MPR* set, read and process the packet but do not retransmit the broadcast packet received from node *N*. For this purpose, each node maintains a set of its neighbours which are called the *MPR* Selectors of the node. Every broadcast message coming from these *MPR* Selectors of a node is assumed to be retransmitted by that node.

Each node selects its multipoint relay set among its one hop neighbours in such a manner that the set covers (in terms of radio range) all the nodes that are two hops away. The multipoint relay set of node *N*, called *MPR(N)*, is an arbitrary subset of the neighbourhood of *N* which satisfies the following condition: every node in the two hop neighbourhood of *N* must have a bi-directional link toward *MPR(N)*. The smaller is the multipoint relay set, the more optimal is the routing protocol.

There are two advantages for the OLSR protocol. Firstly, it reduces the size of control packets: instead of all links, it declares only a subset of links with its neighbours who are its multipoint relay selectors. Secondly, it minimizes flooding of this control traffic by using only the selected nodes, called multipoint relays, to diffuse its messages in the network. Only the multipoint relays of a node retransmit its broadcast messages. This technique significantly reduces the number of retransmissions in a flooding or broadcast procedure. However, with its nature of proactive routing protocols, each node in the network has to periodically exchange update information with each other which will largely increase the system overall signalling overhead. Moreover, in order to choose

the multipoint relays for each node, the information kept in each node will also increase compared to other solutions.

3.1.3   Reactive routing protocols

3.1.3.1 Dynamic Source Routing (DSR) Protocol

DSR [32] is a reactive routing protocol based on source routing algorithm. In DSR, each packet contains complete routing information to reach its destination in its header. Additionally, each node uses caching technology to maintain all known routes that it has learnt.

There are two phases in DSR: the route discovery phase and route maintenance phase. When a node has a packet to send, it firstly checks its route cache and see if there is already a route to the destination. If the required route exists, it will specify the complete path to the destination into the packet's header with the address of each node along the path and forward the packet to the next hop. Otherwise, the source node invokes the route discovery process by broadcasting a route request message to all its neighbours. The route request message contains addresses of both the source node and the destination. Upon receiving the request message, a node will first check its route cache to see if it has the routing information to the destination. If the route to the required destination exists or the route request message reaches its destination, a route reply packet is generated and returned back to the source along the reversed path. Otherwise, the node appends its own address to the route request packet and forwards it to its neighbours.

One example is shown in Figure 10. Source node 1 triggered the route discovery process and broadcasted route request packet to its neighbouring nodes. Every node, upon receiving this routing request message will attach itself to the request packet and broadcast it to its neighbours. After the route discovery packet reaches the destination, a best reverse path will be selected and a route reply packet will be generated and sent back to the source with the whole path information as shown in Figure 10 (b).
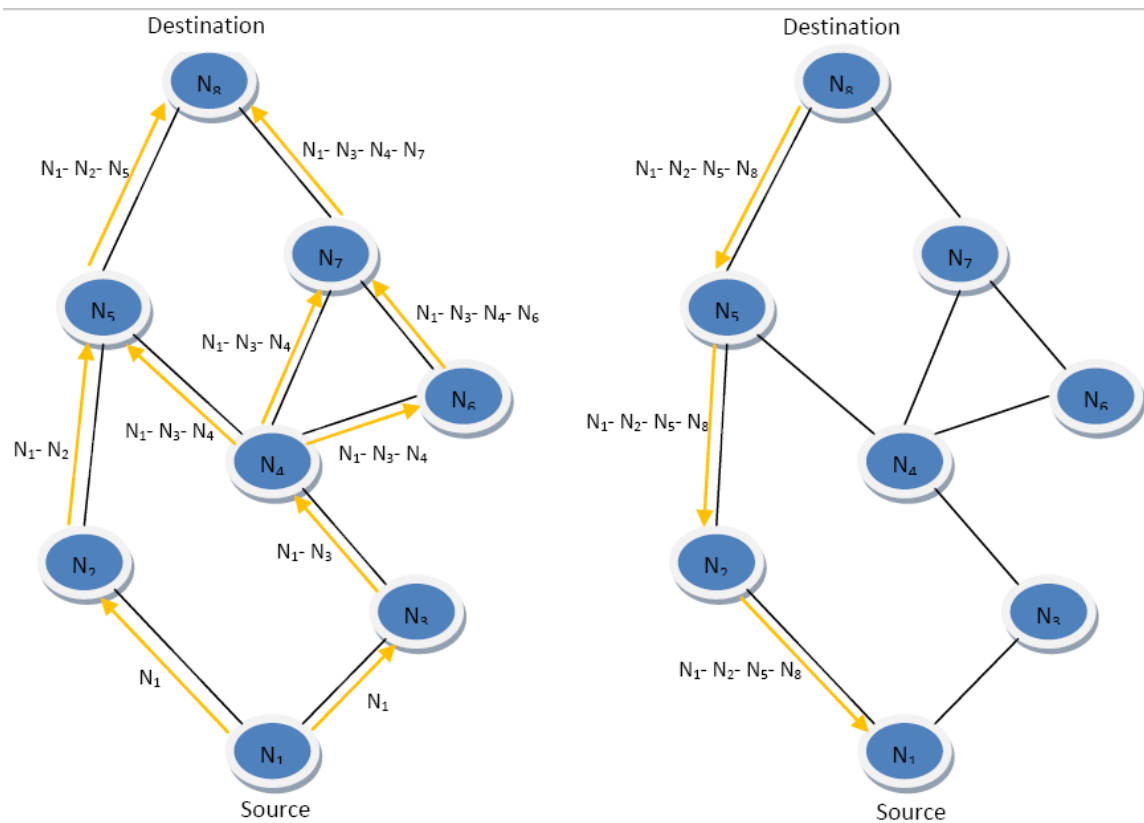
Figure 10 DSR route discovery process

Once the source node receives the reply packet, a route is established and stored in the route cache with a time stamp. The node will start to forward the packets and the route maintenance process begins. When a failed link is detected in the link layer, an error message will be created and sent back to the source node. After receiving the error message, the source node initiates another route discovery process to find a new route.

Additionally, every immediate node along the path who receives the error message will remove all routes which contain the broken link from its route cache.

Since every packet in DSR contains complete addresses of the immediate nodes along the path to the destination, the traffic overhead will increase significantly with the increase of the network size. This is the main disadvantage of DSR for mobile ad hoc networks [36].

3.1.3.2 Ad hoc On-demand Distance Vector (AODV) Routing Protocol

AODV [33] is a reactive uni-cast routing protocol based on the distance vector algorithm. As a reactive routing protocol like DSR, AODV only requests a route when necessary and only maintain the routing information about the active paths. The major difference between AOVD and DSR is that each node in AODV only stores the next-hop information while in DSR every packet contains complete routing information for each flow. The next-hop information in AODV is stored in a next-hop routing table and updated periodically. The routing table expires if it is no longer to be used. Moreover, AODV adopts the destination sequence number technique used by DSDV to determine the freshness of a path.

If a session is triggered and a node needs to send packets to another node, it firstly checks if there is an active route to the required destination. If so, it just forwards the packets to the next hop toward to that destination according to the routing table. Otherwise, it creates a new Route Request (RREQ) message and broadcasts to its neighbours. A RREQ packet contains the addresses of both source and destination, the current sequence number, the last seen sequence number of the destination and the

broadcast ID that is used as its identifier and increased each time the source node

initiates a new RREQ. Also each RREQ includes a time to live (TTL) identifier that is used

to control the propagation hops of a RREQ message. The RREQ starts with a small TTL

value and increased each time a new RREQ message created if the destination is not

found. Once a node receives a RREQ packet, it will discard it if it has seen it before. If the

RREQ is fresh, it records the information and sets up a reverse route entry for the source

node in its routing table. One example is shown in Figure 11. As illustrated in Figure 11

(a), a discovery process is triggered in the resource node 1 and a RREQ message is

broadcasted to the destination node 7. For each node receiving the RREQ, it reserves a

reverse path and forwards a new RREQ to the next hop.
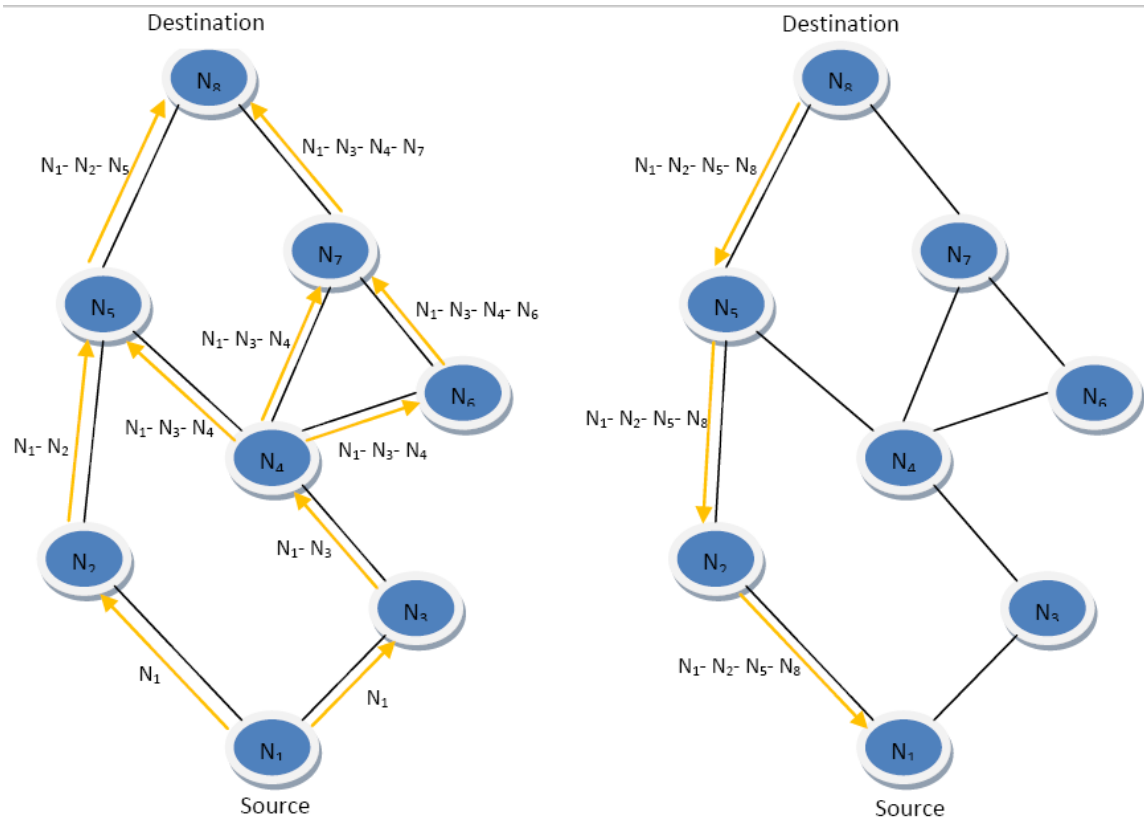
Figure 11 Route discovery process in AODV

Only one reverse route is set up by each node. If the destination node receives a RREQ, it then creates RREP packet and sends it back to the resource along the reverse path. Since each node can only initiate one reverse route, there is only one path the RREP can follow. So in the Figure 11 (b) the reverse path is 1-2-8-7.
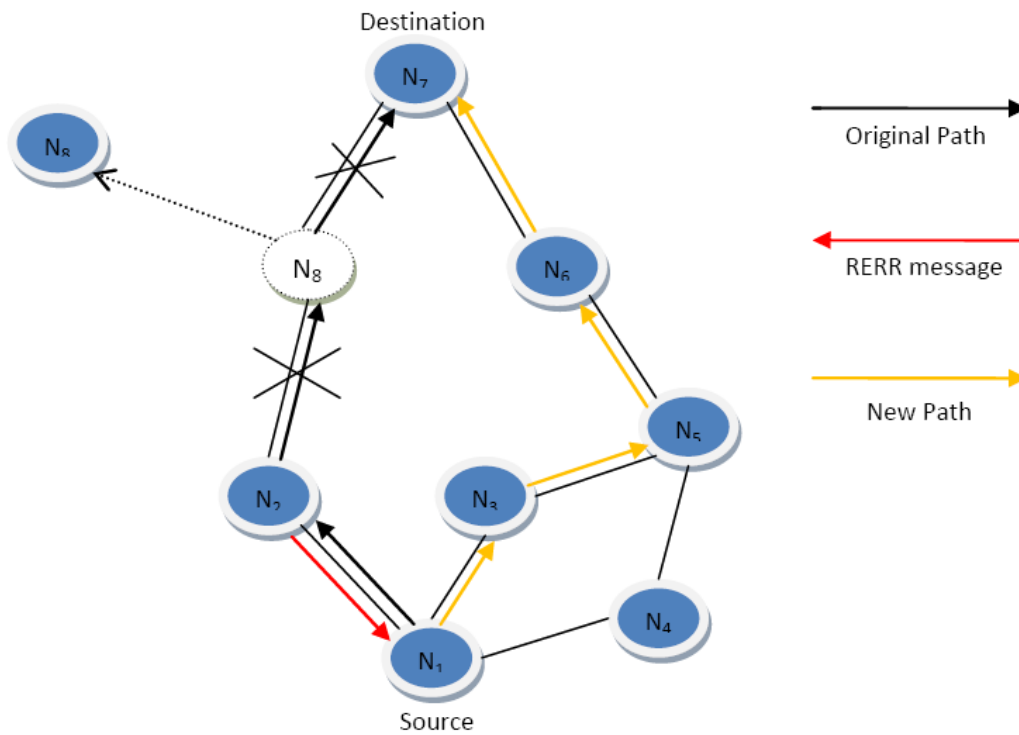
Figure 12 Route maintenance scheme in AODV

Unlike DSR, AODV uses Hello messages to maintain connectivity of a node and its immediate neighbours, and it is also used to detect a failed link. A node periodically broadcasts and receives the Hello messages to and from its neighbours. If a node is not receiving the Hello messages from its neighbour for a period of time, it considers there is a link breakage between them and an error message is created and sent to the source and destination nodes. All nodes along the route that receive the error message will disable the related entries in their routing tables for that route. For example, in Figure

12, the original path is broken because node 8 is moving away. So the affected node 2 will create a RRER message and send back to the source node 1. After that, a new route discovery process will be triggered and a new path will be found. As it can be seen from Figure 12, the new path is 1-3-5-6-7.

3.1.4 Hybrid routing protocols

3.1.4.1 Zone Routing Protocol (ZRP)

ZRP [34] is a zone based hierarchical routing protocol which is proposed to overcome the shortcomings of proactive routing protocols and reactive routing protocols. In ZRP, the network is divided into several independent zones according to the distance between the mobile nodes. Different routing approaches are used for inter-zone and intra-zone packets. A proactive routing protocol called Intra-zone Routing Protocol (IARP) is used within the routing zone and a reactive routing approach named Inter-zone Routing Protocol (IERP) is exploited between routing zones. Each node only needs to keep the topology of the network within its routing zone and if the transmission between source and destination node are in the same zone, a route can be available immediately. Most of the existing proactive routing protocols can be used as the IARP, such as DSDV described above. If the source and destination are residing in different zones, IERP will invoke a route discovery to broadcast the route request. The route discovery in IERP is similar to DSR with the exception that route request messages are broadcasted through the peripheral nodes.

One example is shown in Figure 13. Source node D needs to send a packet to node Z. First it will see if the node Z is in its local zone by checking its routing table provided by

IARP. Since it is not found, then a route request message will be broadcasted to its peripheral nodes (the gray ones highlighted in Figure 13) by IERP. Upon receiving this route request message, each of them will search their routing tables for destination Z. Since the node Z is in the zone of peripheral node L (as shown in Figure 14), node L will attach the path from itself to node Z in the route request. And then a reply message which includes the reversed path will be created in node L and sent back to the source node D.
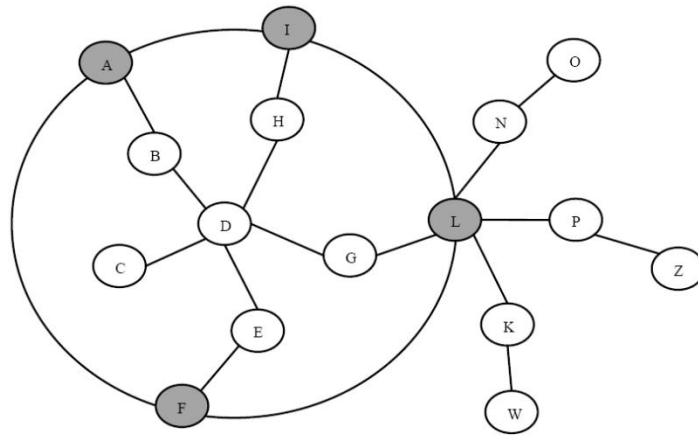
Figure 13 Route discovery for ZRP in phase 1
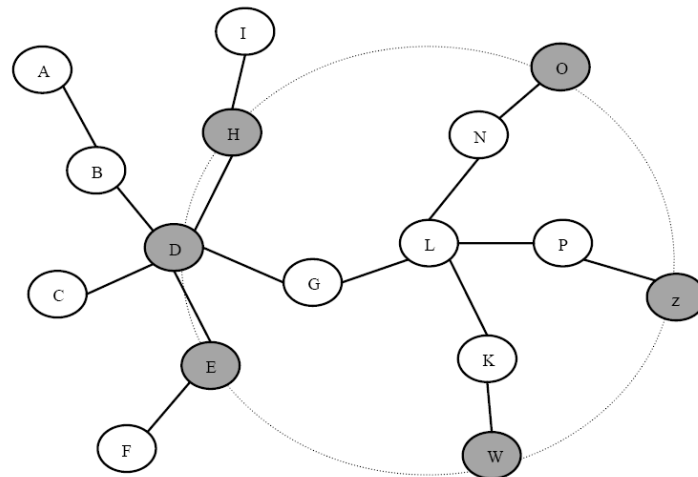
Figure 14 Route discovery for ZRP in phase 2

The prominent advantage of ZRP is that it reduces the control overhead of proactive routing protocols and at the same time decreases the overall latency for route discovery compared to reactive routing protocols. Moreover, the route discovery in ZRP is much faster than the one that takes place in reactive routing protocols, because only the peripheral nodes are queried in the route discovery process. However, the performance of a zone based routing protocol is highly related to the dynamics and size of the network. A breakage of a link may affect several routing zones and the control overhead for update information may be very high.

3.2   QoS Routing Solutions for Mobile Ad Hoc Networks

Recently there are more and more entertainment and other multimedia applications with high stringent delay and reliability requirements, resulting in a shift from best-effort services to the support of higher and better-defined quality of services in MANETs research. However, provision of QoS guarantees in MANETs is much more challenging than in wired networks because of the high node mobility, lack of central coordination, dynamic network topology and limited available network resources. The QoS routing protocol is a crucial part of a QoS solution since it selects the proper paths that satisfy the application's requirements. For the past few years, a lot of work has been done in providing QoS routing solutions for mobile ad hoc networks. Until now, most of the proposed QoS routing protocols in the literature have focused on providing QoS guarantees based on two metrics: bandwidth or throughput and end-to-end delay. This is because in most applications, throughput is considered as a minimum requirement. Some other applications such as video conference require certain level of throughput

guarantees in addition to other constraints. In the next section, the QoS routing protocols proposed in recent years are classified and described in detail.

3.2.1 Issues and challenges in providing QoS in MANETs

MANETs have several unique characteristics which pose great difficulties to provide QoS guarantees. The following is a summary of the major challenges in provisioning QoS in MANETs.

- Unreliable shared radio channel: The wireless channel is prone to bit errors and suffers from several impairments such as interference from neighbouring transmissions, thermal noise, multi-path fading effects, attenuation and shadowing. These effects make it impossible to provide hard throughput and reliability guarantees.

- Dynamic network topology: Mobile nodes in MANETs are moving arbitrarily and independently. This means that topology information has limited lifetime and should be updated frequently for an active route especially for multi-hop paths. Frequent link breakage may significantly degrade the network performance and introduce higher delay for re-establishing a new QoS path. An important general assumption should be mentioned here: the rate of network state information propagation should be greater than the rate of topology change. Otherwise, the routing information will always be stale which results in failure of the routing mechanism. Any network which satisfies this condition is called "*combinatorially stable*" [37].

- Lack of central coordination: In MANETs there is no base station or central controller to coordinate the activity of nodes and efficiently manage the network

resources. All mobile nodes are highly autonomous and self-configured. So any routing protocol operates in a completely distributed way by utilizing only local state information. This will further increase the overhead and complexity to provide QoS guarantees in MANETs.

- Limited mobile device resources: Although many advances in technology have taken place and mobile devices are becoming more and more powerful and capable, generally such devices still have less memory stack, less computational capability and limited power supply compared to the fixed devices such as workstations and desktops. These limitations significantly affect the QoS assurance in MANETs. For example, high mobility and frequent link breakage lead to more frequent state information update and greater signalling overhead. These factors will quickly consume the limited mobile node's battery. Moreover, limited memory capacity limits the amount of QoS information can be maintained in mobile nodes which results in more frequent signalling message exchange.

- Limited network resources: Unlike wired network, network resources such as bandwidth are limited in wireless ad hoc networks. Therefore, a more efficient resource management is required to optimize the utilization of these scarce resources.

In the next section, some recent publications will be presented and described in details.

3.2.3 QoS routing protocols for CDMA based Ad Hoc networks integrating a contention-free TDMA mechanism

In earlier QoS routing protocols, the first problem concerned in QoS provision is to deliver packets over a path which satisfy the throughput requirement. This is because the fact that the assured throughput seems to be the minimum common requirement in multimedia applications. To provide throughput based QoS guarantee, the first part of the solution is to estimate the channel capacity availability to the nodes along a path.

An earlier channel capacity estimation scheme for MANETs is proposed in [39]. A cluster-based network architecture is presented in which nodes are grouped into several clusters. Each cluster is employed a different spreading code based on a CDMA scheme and mobile nodes in a cluster are time-slotted under a TDMA scheme. So the channel capacity for a node can be measured in terms of time slots and the achievable throughput on a link is determined by the set of common free time slots between the transmitter and receiver. In the proposed scheme, the classical DSDV [30] routing protocol is used with the necessary modifications to provide QoS mechanism. The achievable throughput information is stored in the local routing table and time slots are reserved at nodes at the beginning of a session and released when no data packets are received for a period of time.

However, the proposed routing protocol in [39] does not consider the CDMA spreading code allocation which is critical task in mobile ad hoc network. Because there is no central coordinator, every mobile node needs to allocate the spreading code in a totally distributed way. Based on this consideration, a cross layer routing protocol called CDMA

Bus Lane was proposed by Lin Xiao et al. [40]. His routing protocol is based on AODV and the code channel allocation is implemented in a distributed way. TDMA is used along with CDMA to schedule multiple sessions in a node. AODV Hello messages are used and modified to exchange code information between nodes as well as maintain the active routes. An example of the code allocation algorithm along a path is illustrated in Figure 15.
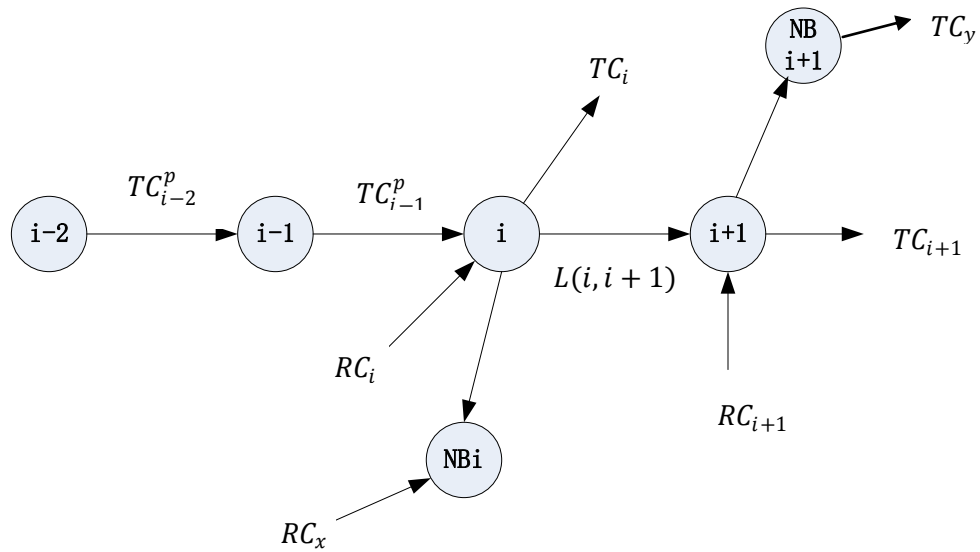
Figure 15 Example of code allocation for link $L(i, i+1)$ on path p [40]

Where:

- RC_i is the reception code set of node i.

- TC_i is the transmission code set of node i.

- RC_{i+1} is the reception code set of node i+1.

- TC_{i+1} is the transmission code set of node i+1.

- NBi is the neighbour nodes of node i.

- NB_{i+1} is the neighbour node of node i+1.

- TC_y is the transmission code set of NBi+1

- RC_x is the reception code set of NB_{i+1}.

- TC_i^p is the transmission code set of node i on path p.

To decide the transmission code set $AC_{i\sim i+1}^p$ for link $i\sim i+1$ on path p, the transmission and reception code set of the neighbour node should be considered. From the node i point of view, to avoid interference and collision, the codes in $RC_i, TC_{i+1}, TC_{i-1}^p, TC_{i-2}^p$ cannot be used for the link $i\sim i+1$. However, because multiple sessions are scheduled in time by TDMA, the codes in TC_i or RC_{i+1} can be reused. For the neighbour nodes, the interference code set comes from the reception code set of the neighbour nodes. However, the reception codes of RC_x originated from transmissions of node i can be ignored because they belong to TC_i. The reception codes of the node i+1 can also be ignored because the reception of multiple sessions in node i+1 will be scheduled by TDMA. So for the NBi, the interference code set can be expressed as [40]:

$$IC_i = RC_x - TC_i - RC_{i+1} \qquad (3.1)$$

Similarly, the interference code set from the neighbour node NB_{i+1} can also be represented by [40]:

$$IC_{i+1} = TC_y - TC_i - RC_{i+1} \qquad (3.2)$$

Therefore, the total available transmission codes set for node i on the link $i\sim i+1$ can be expressed as [40]:

$$AC_{i\sim i+1}^p = \overline{RC_i \cup TC_{i+1} \cup IC_i \cup IC_{i+1}} - TC_{i-1}^p - TC_{i-2}^p \qquad (3.3)$$

The code allocation is done in the route discovery stage with the modified RREQ messages. The code information about the surrounding nodes is periodically updated through Hello messages. However, this routing protocol increases the overall network overhead than the original AODV protocol because for the AODV protocol, the Hello messages are only broadcasted between the nodes which are in an active route. But in the CDMA Bus Lane, the Hello messages are used to exchange the code information. So every node should periodically broadcast their code information even though it is not in an active route.

3.2.4   A mobility based mobile ad hoc network routing protocol

A mobility based routing protocol in mobile ad hoc networks is proposed in [41]. The proposed routing protocol is based on traditional AODV routing protocol and designed for mobile environment where different nodes in an ad hoc network have different mobility characteristics. This paper presents a proposal of using the mobility characteristic of the nodes in the candidate routes as the criteria in setting up a route.

The motivation of this proposal is based on the fact that in a real ad-hoc network, different nodes probably have many mobile speeds. For mobile ad-hoc network, such mobility difference is a significant difference among nodes, as it will impact the network's packets delivery rate, control packets rate, packets delay time and the network's throughput. Therefore, the mobility difference among the nodes shall be taken into account by the routing protocol. The detail algorithm is described as follows:

The author introduces a node's mobility indication which is defined as [41]:

$$M_i = L_{i1} \times 70\% + L_{i2} \times 30\% \qquad (3.4)$$

where M_i is the mobility indication of node i. L_{i1} is the number of link breakage of node i in the last 10 seconds. L_{i2} is the number of link breakage of node i between the last 10 seconds and the last 20 seconds. If M_i is higher, link breakage happens more frequently in node i. If M_i is lower or even zero, node i has no link breakage recently. When setting up a route, the routing protocol shall try to select the nodes with lower M_i so to get a more "stable" route.

During setting up a route, each node except for the source node and the destination node will compare its own mobility indication with the one carried in the RREQ or RREP packet. If a node's mobility indication is greater than the one carried in the RREQ or RREP packet, it will replace the one in the RREQ or RREP packet.

The route selection is made by comparing the mobility indication value in the candidate routes. When receiving multiple RREQ packets, if the difference of the mobility indication values between two candidate routes is greater than a given threshold (this paper defines it 25%), the destination node will select the route with a smaller mobility indication value. Otherwise the destination node will select a route according to the shortest path principle. The source node selects a route following the same way too. The greatest novelty in this paper is to use the number of link breakages as route selecting criteria to increase the packets delivery rate. However, this proposed routing protocol can only have good improvement in high mobility ad hoc networks. Moreover, the routing protocol does not support concurrent transmissions which will limit the overall system throughput.

3.2.5   A cross layer routing protocol in ad hoc networks

Another QoS routing protocol is proposed in [44]. The goal of the proposed QoS routing protocol is to improve the performance of the existing AODV routing protocol by introducing a new routing metric. This routing metric is formed by the received signal strength (RSS) from the physical layer, the remaining energy from the MAC layer and the remaining queue length information from the network layer. This QoS routing protocol aims to find a route with the highest RSS, remaining battery power and remaining queue length. This strategy will lead to select the nodes that have the best connectivity with each other, and to decide about the most powerful nodes that can handle traffic and last longer before their energy resources are depleted. Therefore, the proposed routing strategy can lead to avoid core nodes remaining energy depletion quickly by excluding route selection that contains nodes having poor remaining energy. Moreover, the algorithm can also lead to select the less congested routes and thus lessening the packets dropped by the interface queue.

The cross layer design can be seen from Figure 16. The proposed cross layer routing protocol relies on information (remaining energy, RSS and remaining queue length) obtained from the lower layers. Based on this information, the author defines a routing metric weight which is expressed in equation 3.5.

$$RREQ_{weight} = (\alpha \times RSSI) + (\beta \times RE) + (\gamma \times RQL) \qquad (3.5)$$

Where *RSSI* represents the received signal strength percentage. *RE* represents the remaining energy percentage. *RQL* represents the remaining queue length percentage. α, β and γ represent the weight associated to each attribute and $\alpha + \beta + \gamma = 1$.

During the route discovery process, any intermediate node that receives the route discovery message will calculate the reverse route weight according to the formula in equation 3.5. In the destination node, the path with the weakest hop weight will be selected as the reverse route. Therefore, the route selection mechanism allows avoiding routes that contain weaker nodes which may break the selected path, thus extending the link's life between the source and the destination. However, the author did not specify how to choose the value of $\alpha, \beta\ and\ \gamma$ to get the best system performance. Moreover, it also does not support concurrent transmissions and the analysis of the extra signalling overhead introduced by the proposed routing protocol is missing.
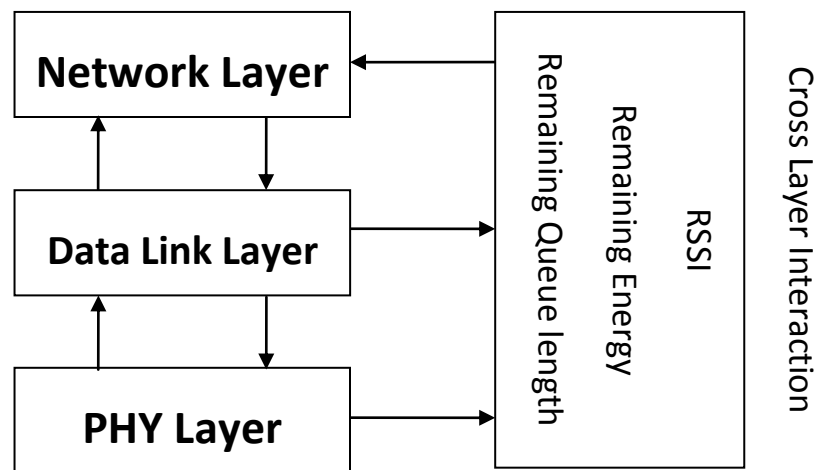
Figure 16 Cross layer design diagram [44]

3.3 Concluding remarks

This chapter presents different routing protocol designs for ad hoc networks. Section 3.1 describes the classic routing protocols proposed in the literature. These routing protocols can be categorized into three classes: proactive routing protocols, reactive routing protocols and hybrid routing protocols. Proactive routing protocol, such as DSDV, does not need route discovery process which can save a plenty of time. On the other

hand, it introduces a lot of signalling overhead to maintain the up-to-date network state which is very energy-consuming. By contrast, the reactive routing protocol, such as AODV, is an on-demand routing protocol which does not need to maintain the network state information. Route discovery process is only triggered when there is a new session requirement. Compared to proactive routing protocols, it can largely reduce the system overhead. However, it will suffer longer delay for the route discovery process and it is not suitable for highly time constraint applications. Hybrid routing protocol, such as ZRP, is a compromise which combines the merits of proactive and reactive routing protocols. However, it requires more complex and intelligent system which is hard to realise in mobile terminals. Moreover, cooperating proactive and reactive routing protocols still need large signalling messages for negotiation between the mobile nodes.

In Section 3.2, some recent publications for developing QoS routing protocols are presented and analysed. Most of the recent proposals are based on the classic routing protocols with different route selection criteria. [41] and [44] are two of the most recent publications on QoS routing protocol in ad hoc networks. However, these two proposals only focus on increasing the route reliability and reduce the link breakage rate. Concurrent transmission and signalling overhead issues are not taken into account. Lin Xiao in [40] proposed a cross layer routing protocol called CDMA Bus Lane which uses CDMA to realize concurrent transmission to improve the system throughput. However, the proposed routing protocol is based on TDMA technique which is very hard to realize in a mobile ad hoc network. Moreover, the CDMA Bus Lane needs each node in an ad hoc network periodically exchange update information with its neighbour which will largely increase the overall system overhead.

In this next chapter, a novel signal-strength based cross layer QoS routing protocol for

OFDMA based mobile ad hoc networks is presented.

Chapter 4

A cross layer QoS routing solution for OFDMA based ad hoc networks

4.1 Introduction

In the mobile ad hoc network, the most important issues in QoS routing are the signalling overhead and collision. Conventionally, in order to distribute the global information among the mobile nodes to allocate the network resources effectively, periodic information exchange between the mobile nodes is needed before the sessions start. This information includes the network resources which are being used in the other nodes, such as the code set in a CDMA-based network or subcarriers in an OFDM-based network. This apriori signalling between the mobile terminals will significantly increase the network overhead and reduce the overall system performance. So how to reduce the signalling overhead is one of the most important challenges. In this chapter, a signal

strength-based sub-channel allocation scheme, SSMAP, is proposed to move the signalling from the up layers to the physical layer as much as possible. A SSMAP based QoS routing protocol is also proposed in this chapter to present a cross layer solution for OFDMA based ad hoc networks. The proposed cross layer routing protocol is aiming to support real-time applications in ad hoc networks, such as teleconference, voice and video transfer. Firstly the system model and some assumptions will be discussed.

4.2 System architecture and assumptions

Consider a wireless OFDMA-based ad hoc network with N subcarriers in total. These subcarriers are further divided into S sub-channels. A sub-channel represents the minimum transmission unit in the network. Each mobile node can transmit and receive on all sub-channels and each user can be allocated one or more sub-channels at a time for transmission or reception depending on the data rate requirements of the user application. For each active sub-channel (i.e., a sub-channel being used in current communication), the power used in its transmission is p_n . If a sub-channel is not being used by the transmitter at the current transmission time, no power is allocated to this sub-channel. It is assumed that the network is homogeneous, which means all the radio parameters are the same in all mobile nodes.

The system model considers the following assumptions:

1. Every mobile terminal can sense the signal strength of each subcarrier and omni-directional transmitting antennas are used by every mobile node.

2. A mobile node can transmit and receive packets at the same time with different sub-channels. Simultaneous transmissions are supported in the nodes by using OFDMA.

3. One Two pairs of transceivers are equipped in each node, one for data packets and another for signaling messages. One OFDMA sub-channel is specifically selected at system level to be solely used for signaling messages such as route discovery and route repair through the signaling transceiver. This signaling sub-channel is known and shared among all users. The Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) mechanism is implemented in the signaling channel to avoid collisions for routing discovery in one area.
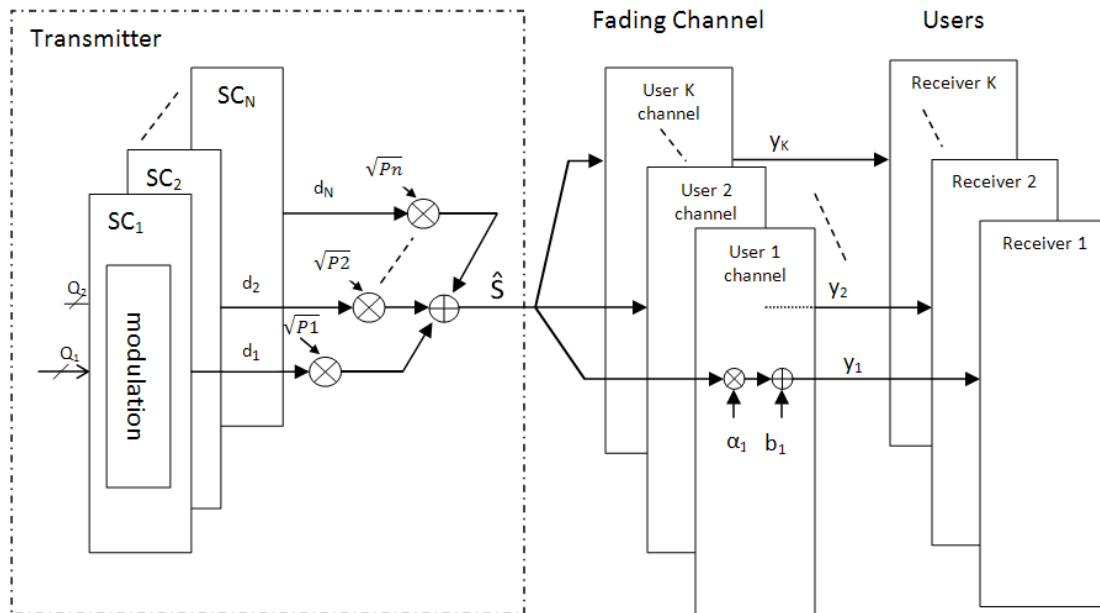
Figure 17 System architecture

The OFDMA transmission system architecture is shown in Figure 17. A mobile node has *N* subcarriers. The overall subcarriers are divided into *S* sub-channels. There are *K* neighbours. The modulated symbol transmitted in the *n*th sub-channel is denoted by d_n.

p_n is the power added to the nth sub-channel. If one sub-channel is not used by the transmitter at the current transmission session, there will be no power allocated to it.

In the receiver side, it is assumed that each subcarrier is propagated through a flat-fading channel with additive Gaussian noise. To simplify the problem, it is assumed that the system is in an open space environment and each sub-channel is experiencing open space path loss. In an open space environment, the path loss of a signal is usually modelled as the Two-Way Ground model [46]. Based on the Two-Way Ground model, when the transmitter is close to the receiver, the receiving signal power is inversely proportional to d^2. d is the distance between the transmitter and the receiver. When d is large (200 meters away or more), the receiving signal power is inversely proportional to d^4 [46]. It is assumed that the mobile nodes are sufficient apart so that the received signal power P_{rcv} for each sub-channel can be expressed as:

$$P_{rcv} = P_t G_t G_r \frac{h_t^2 h_r^2}{d^4} \tag{4.1}$$

Where P_t is the transmission power. G_t and G_r are the antenna gains of the transmitter and receiver respectively, h_t and h_r are the height of the transmitter and receiver antennas respectively. In this thesis, the following antenna characteristics are considered: an omni-directional antenna is used in each node, the antenna gain is 1 and the height of the antenna is set to 1.5 meters. It is assumed that the network is homogeneous which means all the radio parameters are the same in each mobile node. Equation (4.1) states that the received signal power decreases proportionally to the increase of the distance d to the power 4, which means that the farther the transceivers are from each other, the lower is the received signal strength. Note that for the other

fading channels like the Rayleigh fading channel, this phenomenon still exists, but there is a more rapid attenuation in signal strength.

4.3 Sub-channel allocation scheme

OFDMA is used to realize concurrent transmission or reception in ad hoc networks to increase the system throughput and avoid the inherent channel-switching delays of traditional multi-channel MAC protocols such as FDMA. One of the most important considerations for OFDMA based ad hoc networks is how to allocate the sub-channel in a distributed way. This section firstly describes the potential possible multi-session topologies. Secondly, it gives some considerations while implementing the subcarrier allocation. Finally the proposed sub-channel allocation scheme is presented at the end.

4.3.1 Multiple sessions in ad hoc networks

In OFDMA based ad hoc networks, sub-channels are shared among users to support initiation of multiple transmissions at the same time. Basically, application sessions in ad hoc network can be divided into four scenarios [40]: a) Parallel path, b) Flying path, c) Cross path, d) Common path which can be seen from Figure 18. The red nodes represent the session 1 and the blue nodes denote the session 2. The black nodes represent the common nodes which will be shared between these two sessions.

In a parallel path and a flying path, there are no common nodes. So the sub-channels used in each link should be different to avoid co-channel interference. Each node only participates in one session at a time. But in the cross path scenario, two or more sessions are crossing the same node. In this case, the common node will group the

multiple packets together to form an OFDM signal and forward to the related receiver at the same time. The detailed algorithm will be described in the next section. The same thing happens in the common path scenario where two or more sessions have several common nodes in their paths.

a) Parallel path

b) Flying path

c) Cross path

d) Common path
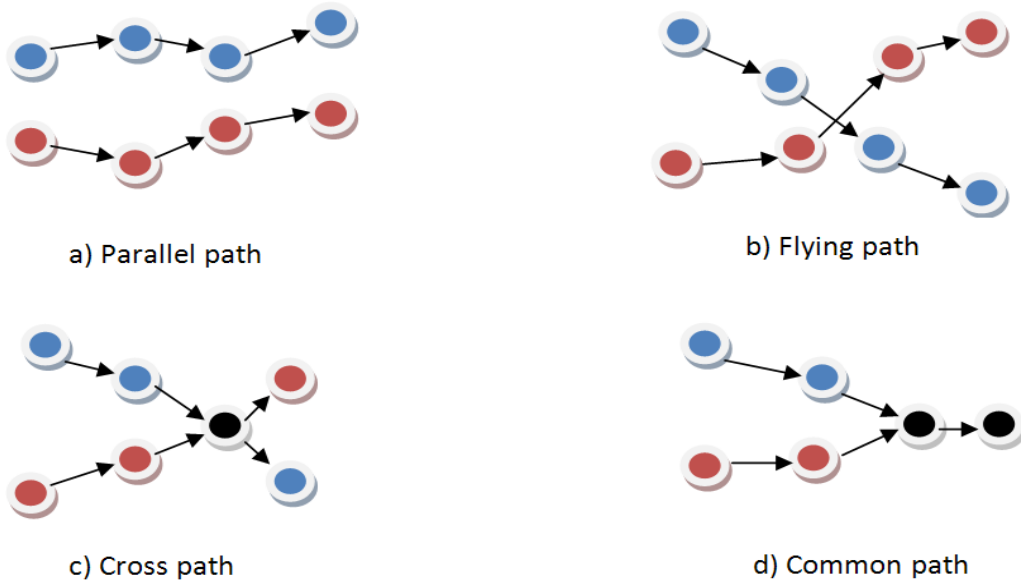
Figure18 Different transmission scenarios for multiple sessions

4.3.2 Considerations in OFDMA sub-channel allocation

The channel bandwidth is a limited resource in ad hoc networks. Therefore, the proposed work in this thesis uses OFDMA to divide the total bandwidth into several sub-channels to support multiple concurrent transmissions. However, it is impractical to allocate different sub-channels to all nodes in an ad hoc network, because the number of sub-channels is limited by the bandwidth, and the number of users is arbitrary. An effective way to support more active transmissions and avoid the co-channel interference is to use the channel reuse concept similar to the concept used in cellular

networks. Consequently, the MAC protocol should allow the allocation of sub-channels that minimizes co-channel interference. The sub-channels should be allocated among the mobile nodes in a fully distributed manner. In [47] and [48], Veyseh et al. proposed a distributed sub-channel allocation scheme for OFDMA based ad hoc networks, however each mobile node must exchange update information with its neighbours. Consequently, Veyseh's method introduces large signalling overhead which reduces the system throughput.

Another problem faced by traditional multiple access mechanisms used in local area ad hoc networks is the hidden terminal problem. The traditional collision avoidance mechanism *RTS/CTS* used to overcome this problem is not always effective and it also requires extra signalling messages. In this thesis, a novel Signal Strength based Medium Access Protocol (SSMAP) is proposed to allocate sub-channels in a fully distributed manner. The SSMAP can also eliminate the need of the RTS/CTS mechanism which can largely reduce the system signalling overhead. Before describing the proposed sub-channel allocation scheme, firstly the effectiveness of the traditional collision avoidance mechanism RTS/CTS is investigated.

4.3.2.1 How effective is the RTS/CTS in ad hoc networks to avoid hidden terminal and exposed terminal problems

In wireless mobile networks, hidden and exposed terminal problems may happen frequently [42][43], which are illustrated in Figure 19. In Figure 19 (a), node A is transmitting packets to node B since it senses the media to be idle. But the media in node B is busy because of the transmission from node C. Hence the packets from node A

are lost in B due to collision. In Figure 19 (b), node C wants to transmit packets to node D. It senses that the media is busy due to the transmission from node B to A. So it backs off to wait the media to become free again. However, the backoff is unnecessary since interference from node C is too weak to affect the reception in node A.

  (a) Hidden terminal problem   (b) Exposed terminal problem
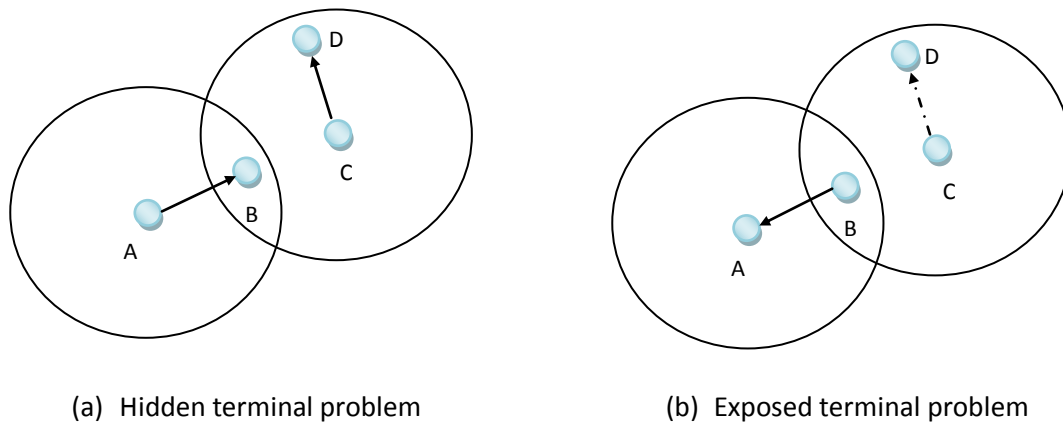
Figure 19 Hidden and exposed terminal problem in wireless networks

How to avoid these problems becomes one of the major considerations in MAC layer protocol. IEEE 802.11x family mainly relies on two techniques to avoid interference in the MAC layer: the physical carrier sensing and the RTS/CTS handshake. The RTS/CTS handshake is specifically designed to avoid the hidden terminal problem. It is based on the assumption that the nodes that are causing interference are in the transmission range of the receivers. However, this RTS/CTS handshake can largely increase the network overhead and the transmission delay. Moreover, in mobile ad hoc networks, due to the highly distributed network topology and the multi-hop operation, the RTS/CTS handshake is not always effective [49].

To illustrate the weakness of the RTS/CTS, an open space environment is assumed. Nodes within the interference range of a receiver node *R* are called hidden nodes of *R*. If

node *R* is receiving packets while its hidden nodes are transmitting with the same sub-channels, packet collisions occur. The Signal to Interference and Noise Ratio (*SINR*) is used to evaluate if a packet can be successfully received. However, compared to the level of the interference power, the noise can be neglected, therefore, as in [49] here only the Signal to Interference Ratio (*SIR*) is considered. Let's denote SIR_{min} as the minimum SIR for a receiver *R* to successfully demodulate packets. In practice, the received *SIR* should be larger than SIR_{min} at the receiver R for the demodulation of a packet to be considered valid.

$$SIR_{rcv} = \frac{P_r}{\sum_k P_i} \geq SIR_{min} \tag{4.2}$$

In equation (4.2), p_r is the received signal power (at receiver R) of the signal from a transmitter node *T*, and $\sum_k p_i$ is the total signal interference power from all the interfering nodes. The received signal power in an open space environment is described in equation (4.1). Therefore, the total signal interference in the receiver node *R* can be expressed as:

$$\sum_k P_i = \sum_k P_t G_t G_r \frac{h_t^2 h_r^2}{r_k^4} \tag{4.3}$$

Equation (4.2) can be rewritten as:

$$SIR_{rcv} = \frac{P_t G_t G_r \frac{h_t^2 h_r^2}{d^4}}{\sum_k P_t G_t G_r \frac{h_t^2 h_r^2}{r_k^4}} = \frac{P_t G_t G_r \frac{h_t^2 h_r^2}{d^4}}{P_t G_t G_r h_t^2 h_r^2 \sum_k \frac{1}{r_k^4}} = \frac{1}{d^4 \sum_k \frac{1}{r_k^4}} \geq SIR_{min} \tag{4.4}$$

Where r_k is the distance between the interfering node N_k to the receiver node R. When k is equal to 1, $r \geq \sqrt[4]{SIR_{min}}\, d$. This means that the receiver node R would be able to successfully receive a packet from a transmitter node T as long as an interfering node N is at a greater distance than $\sqrt[4]{SIR_{min}}\, d$ from node R, where d is the distance between the transmitter node T and the receiver node R. However, the worst case scenario is depicted in Figure 20, where node R is in the coverage boundary of node T, and T wants to transmit packets to node R. The distance d between node R and node T is equal to the transmission range R_{tx} of the transmitter node T. Any interfering node whose distance to the receiver node R is between R_{tx} and $\sqrt[4]{SIR_{min}}\, R_{tx}$ is hidden from the receiver node R and it will interfere with the packets being received by node R. The shaded area represents the interference area and any node in this shaded area is hidden from node R and cannot be detected by the *RTS/CTS* handshake. The transmission from any node (using the same sub-channels as node T to node R) in the shaded area will jeopardize the successful reception in node R.

It is easy to understand that, when the number of interfering nodes N_k increases, the interference range will also increase. As shown in Figure 20, nodes C and D are outside of the interference area. However, the combination of C and D interference signals in node R still may affect R's successful reception of packets from node T. Therefore, in highly distributed ad hoc networks, the *RTS/CTS* handshake mechanism is only effective when the transmitters are very close to the receivers. Moreover, the *RTS/CTS* handshake is not solving the exposed terminal problem.
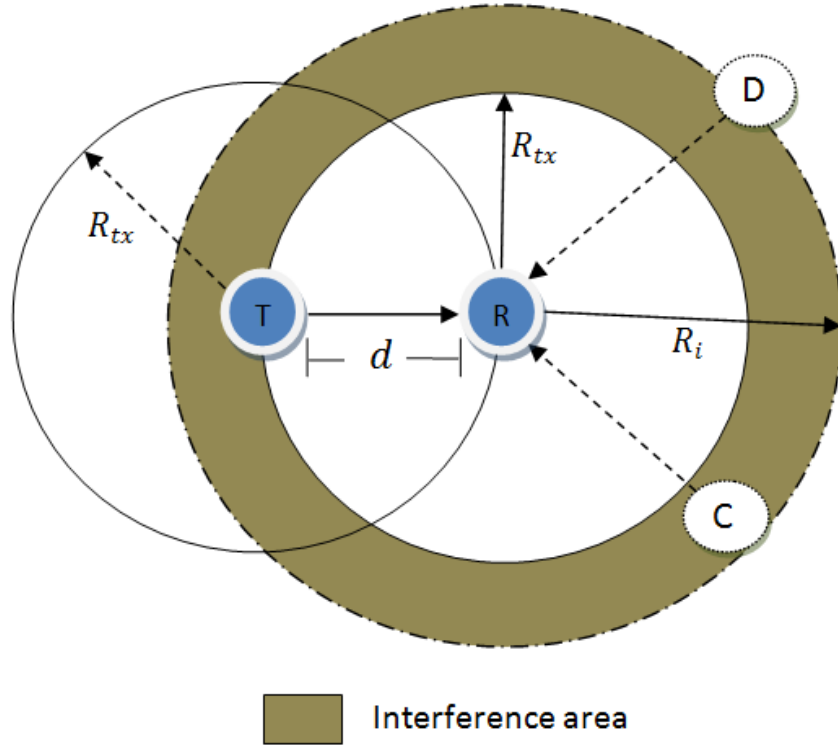
Figure 20 Interference range against transmission range

4.3.3   Signal strength based sub-channel allocation algorithm

Based on the considerations described above and the fact that in OFDMA one node can transmit and receive multiple sessions at the same time, a new sub-channel allocation scheme is proposed in this thesis aiming to reduce the network signalling overhead and system co-channel interference.

In OFDMA based ad hoc networks, the total bandwidth is divided into many sub-channels and each sub-channel is composed of several orthogonal sub-carriers. The sub-channel is the minimum transmission unit. These sub-channels can be shared by the mobile nodes to support concurrent transmissions in the same area. By using OFDMA, one node can transmit multiple sessions at the same time.

SSMAP consists of two main parts: sub-channel sensing and sub-channel selection. The sub-channel sensing is responsible for recording the interference power levels of each sub-channel. The sub-channel selection is responsible for allocating the best suitable sub-channels for each session across a node and avoiding multiple access interference. Below the detailed algorithm is described.

Table 5 The Free_Channel table

Subcarrier index	ORSS	Availability	Priority
SC_6	0	1	1
SC_7	0	1	2
SC_2	1.2386e-012	1	3
SC_1	4.4391e-011	1	4
SC_4	2.7936e-010	0	5
SC_5	7.3479e-010	0	6
SC_3	5.1696e-009	0	7

Each mobile node periodically senses all the sub-channels and records their instant receiving power into a local *Free_Channel* table (see an example in table 5). The *Free_Channel* table consists of four components for each sub-channel: *index*, *Overall Received Signal Strength* (*ORSS*), *availability* and *priority*. *ORSS* records the overall received signal strength for that sub-channel. When a sub-channel's *ORSS* value is lower than a pre-defined carrier sensing threshold TH_s, the *availability* flag for this sub-channel becomes 1, which means this sub-channel is available. Otherwise, the value of its *availability* is 0. The *priority* of each sub-channel relates to its *ORSS*. The sub-channel with the lowest *ORSS* has the highest priority. From the previous sub-section 4.3.2 it is

easy to see that if the packets can be successfully received, the value of TH_s should satisfy equation (4.5):

$$\frac{P_{rcv}}{TH_s} = \frac{P_t G_t G_r \frac{h_t^2 h_r^2}{d^4}}{TH_s} \geq SIR_{min} => TH_s \leq \frac{P_t G_t G_r \frac{h_t^2 h_r^2}{d^4}}{SIR_{min}} \tag{4.5}$$

Where SIR_{min} is the minimum *SIR* for successfully receiving packets. All the variables are constant except for *d*, which is the distance between the transmitter and the receiver. For the worst case scenario, where the transmitter is in the boundary of the coverage of the target receiver, *d* is equal to R_{tx}. Based on this consideration, the maximum value of TH_s should be:

$$TH_s(max) = \frac{P_t G_t G_r \frac{h_t^2 h_r^2}{R_{tx}^4}}{SIR_{min}} \tag{4.6}$$

The equation (4.5) can also be written as:

$$\frac{P_{rcv}}{TH_s} = SIR_{TH} \geq SIR_{min} \tag{4.7}$$

Where SIR_{TH} is the *SIR* threshold to decide the availability of the sub-channels.

To avoid the hidden and exposed terminal problems, the sub-channel selection procedure takes place at the receiver side of a link. The receiver of the link informs the correspondent transmitter what sub-channels should be used. This information is passed through a routing signalling message, because the SSMAP interfaces with the proposed QoS routing protocol in a cross layer integrated scheme. Once a new session request is triggered in one node, it will first check its local *Free_Channel* tables to see if it has enough available sub-channels to carry the new request. If it has available sub-

channels, it will choose the sub-channels with the highest priorities, which are the sub-channels with the lowest received interference signal strength. The sub-channels with the lowest signal strength are also the sub-channels which are being used at the farthest place from the receivers. Consequently, the interference in the local area is minimized. In this way, every mobile terminal can efficiently sense and select the available sub-channels at the local area, without any communication between this node and its neighbours. If every node chooses the sub-channels with the lowest received signal strength, it will indirectly distribute the interference among all sub-channels and the system's overall interference level will decrease. An example is illustrated in Figure 21.
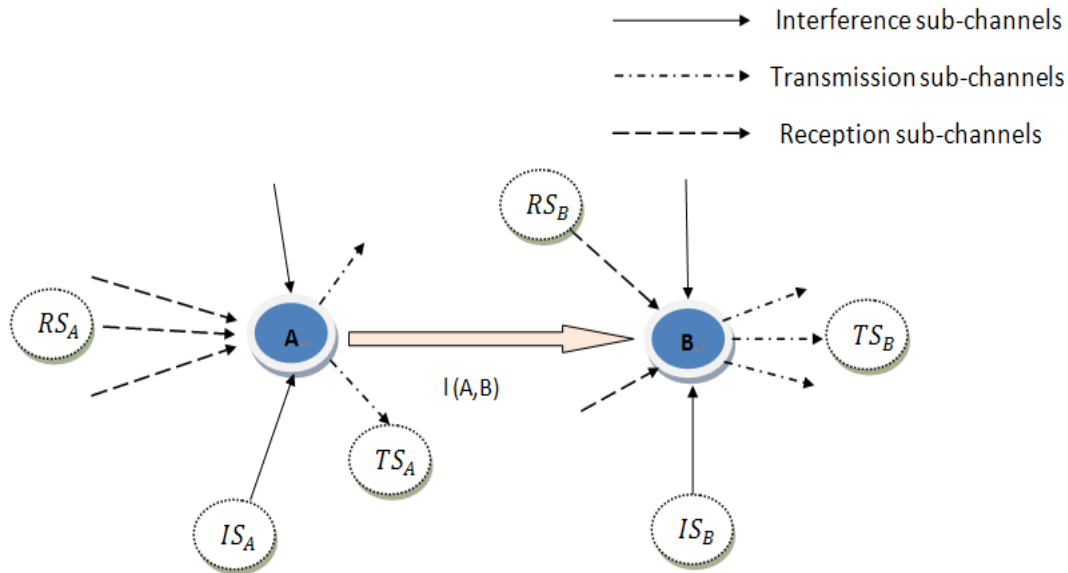
Figure 21 Sub-channel allocation scheme

Where:

RS_A: The set of sub-channels used for reception in node A;

IS_A: The set of sub-channels from interfering nodes received in node A;

TS_A: The set of sub-channels used for transmission in node A;

RS_B: The set of sub-channels used for reception in node B;

IS_B: The set of sub-channels from interfering nodes received in node B;

TS_B: The set of sub-channels used for transmission in node B;

The example assumes that node A wants to transmit packets to node B. The dashed arrows are the sub-channels being used for reception in both nodes and the dashed-point arrows represent the sub-channels used for transmission. The solid arrows are the interference from the surrounding nodes. Both nodes A and B will sense the signal strength of all sub-channels and record the values in their local *Free_Channel* tables. Node B will decide which sub-channels are going to be used by node A for its transmission through routing signalling messages. B will first check whether there are enough available sub-channels in its local *Free_Channel* table. If it does, then it will choose the sub-channels with the highest priority from the available sub-channel list. One thing should be noted that one node can only transmit and receive packets simultaneously in different sub-channels. Therefore, whichever sub-channels node A has in its RS_A, these sub-channels cannot be used as transmission sub-channels to node B (TS_A). This can be solved in the route discovery process. Node A will attach RS_A in the route discovery message to inform node B that these sub-channels should be avoided.

Through the proposed sub-channel allocation scheme, the hidden terminal problem can be completely avoided and the exposed terminal problem can also be relieved by

selecting an appropriate value of TH_s. Moreover, by increasing the sub-channel sensitivity and decreasing TH_s, *CTS/RTS* messages might be no longer needed for packet transmission. The implication is that the carrier sensing range for the sub-channels will increase compared to normal *CSMA/CA* mechanism.

4.4 The QoS routing protocol with the sub-channel allocation scheme

In chapter 3, many routing protocols have already been described as well as recent QoS routing protocols. In mobile ad hoc networks, on-demand routing protocols have better performance than table driven routing protocols in terms of signalling overhead and power consumption. To support QoS in mobile ad hoc networks, especially for real-time service, it is necessary to find a route from source to destination with guaranteed constant bandwidth and stringent end to end delay. In this thesis, a QoS routing protocol with bandwidth and end to end delay consideration is proposed incorporating the proposed sub-channel allocation scheme to form a cross layer QoS routing solution.

The proposed sub-channel allocation scheme is signal-strength based and when a new session is required the receiver will decide which sub-channel is going to use in the transmitter side. The signalling message can be encapsulated into the routing discovery message and transmitted from the transmitter to the receiver.

When a new session is triggered, a route discovery message (*RDIS*) is broadcasted from the source node. This message contains the source node current sub-channel usage information including the reception sub-channels and available sub-channels. The QoS information can also be encapsulated into the *RDIS* messages to find the route which

meets the QoS requirement. The nodes which are receiving the information will check their local routing table and choose the receiving sub-channels and forward its local information to the next hop.

The *RDIS* message is broadcasted from the source node. Each node receiving the route request message will modify the message according to the information attached to the *RDIS* message and its local data base, the node then will broadcast the modified *RDIS* to the next hop. This process will continue until it reaches the destination node. The destination node will wait for multiple routes to be received and then it will choose the one with the lowest interference level.

This chapter describes the details of the proposed cross layer QoS routing protocol. It combines the sub-channel allocation scheme with the on-demand routing protocol. The network layer and the MAC layer cooperate with each other to support concurrent sessions with bandwidth and end-to-end delay restrictions. Firstly the formats of the necessary signalling messages exchanged among nodes and the information required to be maintained in each node are introduced. After that, methods to implement the proposed sub-channel allocation scheme with the on demand routing request are described. Finally, a route maintenance scheme is presented.

4.4.1 The QoS routing signalling messages

Without central control, all the calculations for the route discovery and scheduling are through negotiations among nodes in a distributed way. These negotiations can be done by signalling messages exchange through a special signalling sub-channel. The formats and functions of these signalling messages are described as follow:

1) *Route Discovery Message (RDIS)*

The *RDIS* is a multi-hop message generated by the source node at the route discovery stage. Besides the attributes in the classic RREQ message of AODV, the *RDIS* also contains the bandwidth requirement, delay restriction, time delay information and sub-channel usage information from the previous hop. Figure 22 shows the details of *RDIS* packet format. The time stamp is the current time of sending this *RDIS* message according to the local clock and the *Cumulative Delay* is the overall end to end delay from the source node to the previous node. The *Reception Sub-channels* indicates the sub-channels used by the previous node for reception. The *Interference Level* is the cumulative interference power level from the source to the previous node. This information will be used at the destination node to choose the best path.

Type	Hop Count	Broadcast ID	Bandwidth Requirement	Cumulative Delay
Node IP Address		Source IP Address	Destination IP Address	Delay Restriction
Source Sequence Number		Destination Sequence Number		Time Stamp
Reception Sub-channels			TTL	Interference Level

Figure 22 *RDIS* packet format

The node which receives the *RDIS* message will check the attached information with its local routing tables to allocate valid sub-channels for the link from the node where the *RDIS* message was sent. The allocated resources are just pre-reserved in the node and will be fully reserved only if a confirmation message is received from the destination node.

2) *Route Response Message (RRES)*

Once the first *RDIS* is received in the destination node, it will wait for a short time to see if there are other alternative routes. These paths all satisfy the bandwidth requirement and end to end delay restriction with different interference level. Once it confirms a path, a *RRES* message will be generated and sent back to the source node through the backward path. The Figure 23 describes the detailed *RRES* message format.

Type	Hop Count	Next Hop	Node IP Address
Source IP Address		Destination IP Address	
Transmission Sub-channels		Destination Sequence Number	

Figure 23 *RRES* message format

The *Transmission Sub-channels* tell the reverse path receiver node (up-stream) which sub-channel it should use for transmission during the request session (down-stream). When an intermediate node receives the *RRES* message, it will confirm the reserved sub-channels used for the new session. After that, it generates a new *RRES* message with its IP address and forward to the next hop in the reverse path.

3) ***Sub-channel Reallocation Message (SREA)***

The *SREA* message is used when a down-stream node asks for a transmission sub-channel reallocation. It happens when the interference level of a reception sub-channel is higher than a certain level. This is because a node which is transmitting with the same sub-channel is moving closer or a new session is starting in a neighbour node which is using the same sub-channel as transmitting sub-channels. The *SREA* message will be sent to the related transmitter with reallocated transmission sub-channels. Figure 24 shows the detailed *SREA* packet. The *Node IP Address* is the affected node and the

Transmitting Node IP Address is the up-stream transmitting node which needs to change the transmission sub-channels.

Type	Transmitting Node IP Address	Node IP Address
Source IP Address		Destination IP Address
Reallocated Transmission Sub-channels		

Figure 24 *SREA* packet format

4) ***Sub-channel Reallocation Reply Message (SRRP)***

After the node receives the *SREA* message from the down-stream node, it will check with the local routing table to see if the new transmission sub-channels allocated by the down-stream receiver have already been used as receiving sub-channels. Then it will generate a *SRRP* message and sent back to the node where the *SREA* message was sent. The *SRRP* message format is described in Figure 25.

Type	Receiving Node IP Address	Node IP Address
Session Source IP Address		Session Destination IP Address
Receiving Sub-channels		Confirmation Flag

Figure 25 *SRRP* message format

If the reallocated sub-channels are not used as reception, the *confirmation flag* is 1. Otherwise the value of *confirmation flag* is 0. In this case, the node who sent a *SREA* message need to send a new *SREA* message back to the transmission node again with a new allocated transmission sub-channels according the returned *SRRP* message.

5) *Route Error message (RERR)*

A *RERR* message is created when a link breakage occurs in a route. It happened because of the node movement or interference node approaching and the failure of local repair. Figure 26 shows the packet format for *RERR*. The nodes on both sides of the failed link generate a *RERR* message and forward to other nodes on the same route on both directions.

Type	Next Hop	Node IP Address
Session Source IP Address		Session Destination IP Address

Figure 26 *RERR* packet format

The *Next Hop* gives the IP address of either up-stream or down-stream nodes which are still connected with the node on the route. Along the propagation of the *RERR* message, all the reserved network resources and route information for the related session are released.

4.4.2 The information maintained in each node

In the proposed routing protocol, some information needs to be kept in the mobile nodes for sub-channel allocation and QoS routing. The information kept in each node includes the *QoS routing table*, the *Free_Channel routing table*, *slot scheduling information*, *resource pre-reservation information*, *route discovery state information* and *neighbour information*. These tables are described in detail as follows:

1. QoS Routing table

It records the routing information for all the sessions the nodes involved. Each session is identified by a source and destination node ID address. For each session entry, there is an *ID* assigned by the node and a flag to describe if it is an active entry. There is a *destination sequence* number for each route which shows the freshness of the route. There are other classic attributes such as the hops to the destination from this node, the next hop node address, *a breakage flag* to show if the route is currently broken and a *repair flag* to show if the route is at a repair stage. For each active route, there are two attributes to indicate the bandwidth supported and end to end delay. The routing table also contains the information about the transmission and reception sub-channels used for each active session.

2. Free_Channel table

This table records the signal strength information for each node. Each time slot has a *Free_Channel table* for all sub-channels. The active *Free_channel table* contains the signal strength for all sub-channels and the priority for each of them. The sub-channel with the lowest signal strength has the highest priority to be used as a reception sub-channel. The availability attribute in the *Free_channel table* indicates the available sub-channels which can be used as reception sub-channels without causing serious interference in this node. There is a pre-defined threshold to distinguish these available sub-channels which is explained in section (4.3.3). Each sub-channel whose signal strength is lower than the threshold is marked as available one. There is a tradeoff between the value of the threshold and the number of available sub-channels. The

higher the threshold value is the more available sub-channels this node can use, but the higher interference level it will face. It is necessary to set the threshold value to balance the sub-channel availability and interference to get maximum performance.

3. Route discovery state table

It is used at the route discovery stage. This table records the route discovery messages (*RDIS*) which have been successfully received and forwarded by this node. In the classic AODV routing protocol, a node will forward all the route request messages it has not seen before. The route request message (RREQ) is identified by its source and destination IP address and broadcast ID. Each new RREQ message will be recorded in the table to avoid sending the same message again. In this way, a dead loop at the route discovery stage can be avoided. Similar to the table in AODV, the route discovery state table in the proposed routing protocol records every new *RDIS*. However, it only forwards the new *RDIS* messages to the next hop only if it can satisfy the bandwidth and end to end delay requirement appended in the *RDIS* messages. If the bandwidth and delay requirement is satisfied, an interference power level for the selected sub-channel will be attached in the new *RDIS* message and forward to the next hop.

4. Pre-reserved table

Once a route discovery message is received at a mobile node, if it can satisfy the bandwidth and end-to-end delay requirement, it will pre-reserve transmission sub-channels for the node where the *RDIS* came from. Unless a route reply message is received from the destination node or the route discovery process times out, the pre-

reserved sub-channels cannot be used by the other sessions. The pre-reserved information is recorded in the pre-reserved table.

5. *Neighbour information table*

Since the reception transmission sub-channels cannot be used for transmission in the same node, the reception information is attached into the route discovery messages as well as the active transmission sub-channels. Once a node received this information, it records it into the local neighbour information table. This information is distinguished by the neighbour node's ID and address. It only records the information for the active neighbour nodes which are involved in a route discovery process.

4.4.3 The sub-channel allocation at route discovery and establishment stage

The proposed sub-channel allocation scheme is performed in a completely distributed way. Each mobile node in an ad hoc network does not need to keep the transmission or reception sub-channels for all other nodes. Each node does not need to periodically exchange information with their neighbours for the sub-channel assignments. The sub-channel allocation is only started when the route discovery process is triggered. A crucial issue to be solved is how to determine which node should take the responsibility for the sub-channel selection and assignment. As described in the previous section 4.3, the sub-channel allocation algorithm is based on the received signal *SINR* to minimize the overall interference level. The signal strength measured in the transmitter is probably different from the one measured in the receiver for the same sub-channel. If the sub-channel selection is made by the transmitter, it has to know the sub-channel information in the receiver side to make the decision. Therefore, in the proposed

routing protocol the receiver performs the sub-channel allocation process according to its local *Free_Channel table*. However, the transmitter still needs to inform the receiver its active transmission and receiving sub-channels to avoid collisions. This information is encapsulated into the route request messages.

To provide QoS and support real time service, bandwidth and end to end delay are taken into account in the sub-channel and route selection. To guarantee the bandwidth requirement for each session, bandwidth calculation is taking place when a new session request is received. Assuming the total sub-carriers are divided into *N* sub-channels and the bandwidth for each sub-channel can be denoted as BW_s. The sub-channel is the minimum transmission unit. To simplify the scenario, it is also assumed that each sub-carrier is perfectly orthogonal with the others and sub-channels are independent with each other so that they can be used simultaneously in the same area without causing interference. One or more sub-channels can be used for a session depending on the bandwidth requirement. Therefore, the number of sub-channels n which should be used for a session should satisfy the following condition:

$$(n-1) \times BW_s \leq R_s \leq n \times BW_s \tag{4.8}$$

where R_s is the bandwidth requirement of this session which is inserted in the *RDIS* message. Sub-channels are shared among the mobile nodes to support concurrent transmission. Each mobile node will only use certain amount of sub-channels and only apply transmission power to the sub-carriers which will be used for the current session. The remaining subcarriers will not be powered.

For the delay sensitive networks, some sessions may require stringent end to end delay. In order to discover a route which satisfies the end to end delay requirements of a requested session, a field called *cumulative delay* with value $D_{cumulative}$ is created in the *RDIS* message to record the time elapsed. The calculation can be expressed as follow:

$$D_{total} = (T_{receive} - T_{transmit}) + D_{cumulative} \qquad (4.9)$$

where $D_{cumulative}$ is the cumulative delay from the source node to the previous hop recorded in the received *RDIS* packet. $T_{receive}$ and $T_{transmit}$ are the time stamps that the receiver node starts receiving a packet on the link and the previous node starts transmitting this packet respectively. The value of D_{total} should be less than the end-to-end delay requirement indicated in the *RDIS* message for the node to forward the request. Once a new *RDIS* message needs to be created from this node, the calculated value of D_{total} is assigned to the *cumulative delay* field in the new *RDIS* message.

4.4.3.1 The event driven QoS routing protocol

As each node is independent, the system can be modelled as an event driven system. Each action is triggered by an event from inside or outside of the node. These events can be the arrival of a new packet or a timer that expires. Moreover, the action triggered by an event depends on the state the node is in. After a node powers on, it can be in three different states:

- *Idle:* The mobile node enters into this state right after it powers on. In this state, there is no active route across this node.

- *Waiting:* When a new session request is received from the application layer or a *RDIS* message is received from a neighbouring node, the node moves from the idle

to the waiting state. If the session request is accepted in principle, a new *RDIS* message is generated and broadcasted to its neighbours. The node stays in the waiting state until a timer expires or a Route Response message *RRES* from the destination node is received.

- **Active:** After the node receives a RRSP message from the destination node and the pre-reserved resources are still available, the node goes to the active state straight away. In this state, the network resources are concretely reserved and a new route entry is created.
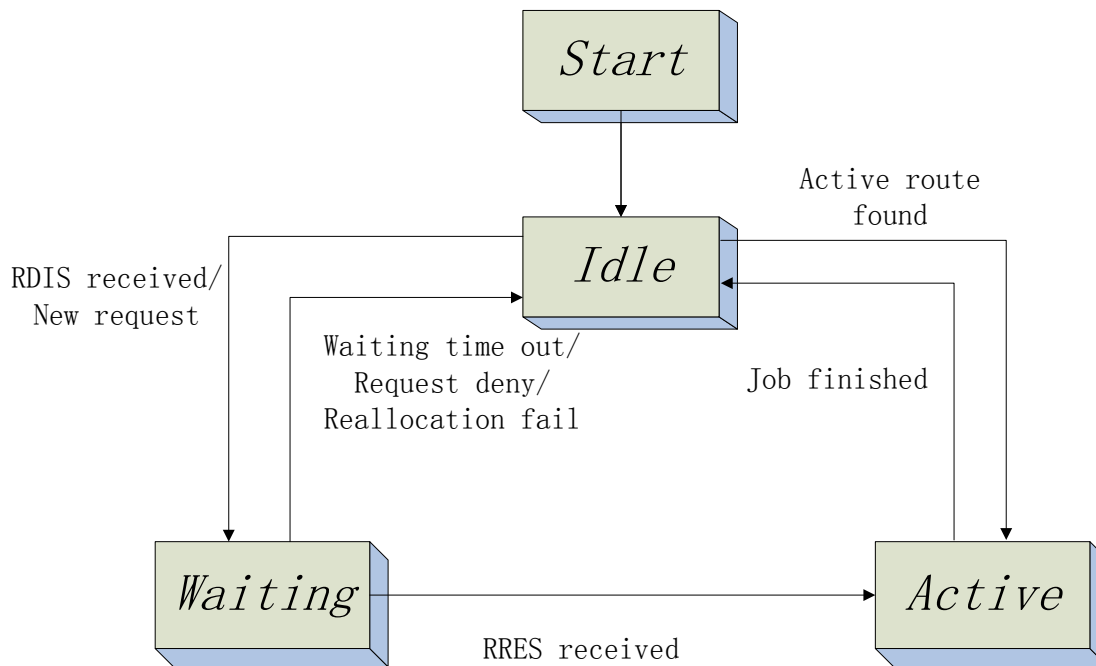
Figure 27 The transition diagram for the route establishment

The transition diagram for the route discovery and establishment stage is illustrated in Figure 27. The whole process of the route discovery and establishment can be explained by the event driven system. The detailed algorithm is explained as follows:

- **Event:** A node receives a data packet from the application layer directed to a given destination node with specific bandwidth and end to end delay requirements.

- **Actions:** The node checks its routing table to see if it has already an active route entry for this destination that satisfies the bandwidth and end to end delay requirements. If it does, it enters in the *Active* state and starts to forward the packets to the next hop according to its routing table. Otherwise, a new route discovery is needed. If a route discovery process has already been carried out for this destination, the node inserts the packet into the corresponding buffer, until it receives the reply message sent from the destination node. Otherwise, it will enter into the waiting state and a new route discovery process is triggered.

  Once the node initiates a new route discovery process, it first checks its local *Free_Channel* tables to see if there are enough available sub-channels to carry out this request. If it does not have enough sub-channels to satisfy the bandwidth requirement, it will deny this request. Otherwise, a *RDIS* message is created with an IP address and a sequence number for both the source and the destination nodes. A time stamp is created and inserted into the *RDIS* message to inform the time this packet was created to the next hop node. The bandwidth and end-to-end delay requirement are also recorded and encapsulated into the *RDIS* message. The *RDIS* message also contains an initialized *Time To Live* (*TTL*) field in order to limit the hops of the desired route, and a broadcast *ID*. The *TTL* is increased by one each time the source node re-sends a route discovery message. To facilitate the next hop to decide the sub-channels to be used in the link on the path, the transmission and reception sub-channels currently in use for other on-going sessions on the node (if

any) are recorded into the *RDIS* message. After creating the *RDIS*, the source node broadcasts it to its neighbouring nodes and set up a timer (*RRES_WAITING_TIME*) for the *RRES* message. The node enters into the waiting state. Furthermore, a value for *RDIS_RETRIES* is set up indicating the maximum number of retries possible if no response is received.

- **Event:** A node receives a *RDIS* message from its neighbour.

- **Action:** When a node receives a *RDIS* from its neighbour, its actions depend on the type of node.

1) The node is not the destination node: Firstly it checks its local route discovery state table to see if it has seen this request before. If so, it simply discards this message. If not, the node will then check if the node itself satisfies the bandwidth and end to end requirements for the requested session. If all QoS requirements are satisfied, the node selects the sub-channels with the highest priority from the *Free_Channel* table and reserves them for transmission node where the *RDIS* message came from on the reverse link and writes this information in its pre-reserved table. After that, the *cumulative delay* is recalculated and a new *RDIS* message is created. The node adds its *IP* address and current transmission time as well as its currently used transmission and reception sub-channels (if it has) into *RDIS*. The signal strength of the selected sub-channels is added to the *Total_Interference_Level* field in the *RDIS* message. After the node broadcasts the *RDIS* message to its neighbours, it sets a timer (*RRES_WAITING_TIME*) to wait for the reply message.

2) The node is the destination of the requested session: It firstly checks if it has received the request from the same source node before. If not, it will check if it

satisfies the bandwidth and end to end requirements as described above. If the QoS requirements are satisfied, it will set a timer for this source node and wait for other possible requests from other routes, otherwise, it creates a Route Error (*RERR*) message sends back to the source node to indicate the failure of the session request. If the node has already seen the request before, it just keeps this message and records the route information. Once the timer is expired, it will check the *Total_Interference_Level* in each request message and choose the one with the least value to send a *RRES* message back to the source node through the reverse path. The destination node assigns the transmission sub-channels to be used by the neighbouring node on the reverse link into the *RRES* message.

- **Event:** A node receives a *RRES* message.

- **Action:** Once a node receives *RRES* message, it checks if it has already set up a path for this session before. If the requested route entry has already been established in its routing table or there are no pre-reserved resources for this request, it simply ignores this message. If this reply message is what the node is waiting for, it transfers the pre-reserved resources to a route table entry it creates for this session. The actions are different for an intermediate node and the source node.

1) If the node receiving the *RRES* is not the source node, it checks if the pre-reserved resources are still available. If so, the node creates a routing table entry for this session with the source and destination nodes' *IP* addresses and the sub-channels it will use for reception on the session link path. Moreover, it reads the transmission information from the *RRES* to set up its sub-channels for transmission on the session link path. However, the pre-reserved sub-channels might be no longer

available. In this case, the node rechecks its *Free_Channel* tables to find other available sub-channels. If an alternative is found, the new assigned sub-channels are recorded into a new created *RRES* with updated information and forwarded to the next hop. If there are no enough resources available, an error message *RERR* is generated and broadcasted along the path in both directions.

2) If the source node receives a *RRES*, it creates a routing table for this session with the bandwidth and end to end delay specifications. It reads the transmission information from the received *RRES* message and reserves these sub-channels in the routing table. Once the route entry is created, the node starts to transmit the packets to the destination node.

- *Event: RRES_WAITING_TIME* has expired before a valid reply message arrives.

- *Action:* If this event happened in the source node and the number of retries is less than or equal to *RDIS_RETRIES*, the node regenerates a new *RDIS* message and rebroadcasts it to find a route again. Otherwise, the node drops the data packets in the buffer, terminates the route discovery process and goes back to the idle state. When the source node retries the route discovery process, it increases the *TTL* value, the broadcast *ID*, the *sequence number* of both itself and destination node to distinguish from the previous route discovery message. After creating the new *RDIS* message, the node resets the timer *RRES_WAITING_TIME* and increases the retry counter.

If the node is not a source node, it just releases all the network resources pre-reserved for this session and goes back to the idle state.

- **Event:** Interference level of the active reception sub-channels are greater than a pre-defined threshold

- **Action:** Because in the proposed subcarrier allocation scheme, the sub-channels selection is made on the receiver side. When a new session starts, it may affect the other on-going sessions. Moreover, in mobile ad hoc network, every node within the network can move arbitrarily. So if two nodes with the same sub-channels being used are approaching each other, the interference between then will increase. In here, a *high_interference_trigger* threshold is introduced to measure the interference level of specific sub-channels. If the interference level of the sub-channels which are being used for the current session is higher than the *high_interference_trigger*, then the node starts to check its local *Free_Channel* table to search for available sub-channels for the current session. Once it finds alternative sub-channels, a Sub-channel Re-allocation Message (*SREA*) message will be created and it will be sent back to the transmitter to indicate the new transmission sub-channels. If there is no available sub-channels in the *Free_Channel* table, a Route Error Message (*RERR*) message will be created and it will be sent back to the source node to launch a new route discover process.
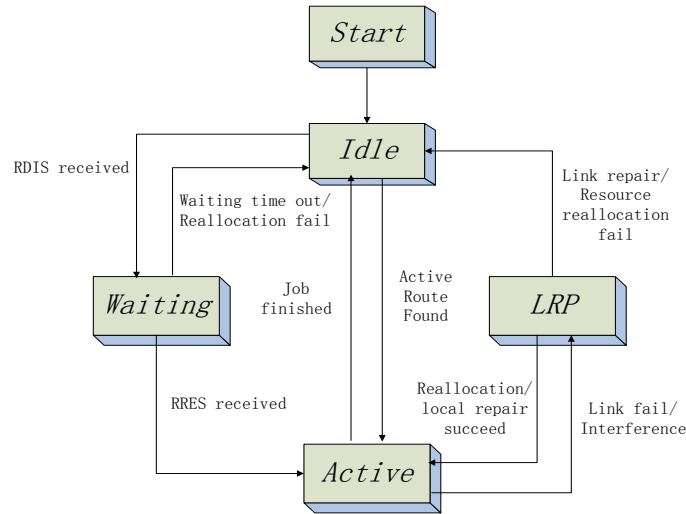
Figure 28 Transit diagram for the route maintenance

4.4.4 The route maintenance scheme

Normally, the route maintenance scheme is one of the most difficult tasks in ad hoc networks because of the frequent topology changes and large signalling overhead. Once a route has been discovered and established for a given source/destination pair, it is maintained as long as it is needed. In the multi-channel OFDMA based ad hoc networks, the movement of the mobile nodes may cause either link breakage or co-channel interference.

Traditionally, in a routing maintenance scheme such as the one in the AODV routing protocol, a so called *Hello* messages are used for maintaining the active route connectivity. Every node which is in an active route will periodically broadcast a *Hello* message to indicate its availability. However, this broadcasting mechanism will significantly increase the network overhead and decrease the system performance. Base on this consideration, this thesis proposes a new route maintenance scheme which avoids using the *Hello* messages. In the route maintenance scheme, there is one more

state called *LRP* (link repair) as shown in Figure 28. The node will enter into this state when it detects a forthcoming link breakage or the co-channel interference level is higher than a pre-defined threshold. The detailed algorithm is described below.

1) **Node in an active route is moving away**

A route has been setup for a session and this session is actively transferring packets. Instead of periodically broadcasting a Hello message, every mobile node participating in a session will continue sensing the signal strength of the sub-channels, especially the sub-channels used by the participant neighbouring nodes. Two signal strength thresholds are defined in here, a *low_threshold_trigger* and a *high_interference_trigger*. If the signal strength of the receiving sub-channel used by a neighbouring node reduces to less than the *low_threshold_trigger* and lasts for a pre-defined time span τ_1, it means the neighbouring node is continuously moving away. Therefore, a forthcoming route breakage is expected. The receiver nodes will flood a route repair message to its neighbours to look for alternative nodes in the neighbourhood. If there is no such node available, a route error message (*RERR*) will be generated and sent to both sides of the route. The node receiving the *RERR* message will release all the reserved resources for the session and forward the *RERR* message to the source or destination node. Once the source node receives the *RERR* message, it will initiate a new RDIS message to look for a new route. In the high mobility environment, if a link is broken before the node can predict due to the high mobility, the affected receiver node will notice this for the sudden dropped signal strength of the receiving sub-channel and send the *RERR* message toward the destination node. The affected transmitter will notice this sudden

link breakage by missing the ACK messages sent back from the receiver for each data packet. It will then also create a *RERR* message and send toward the source node.

There are two advantages for using this route maintenance mechanism. First of all, the mobile nodes do not need to periodically broadcast *Hello* messages. The network overall signalling overhead is largely reduced. Secondly, by continuously monitoring the signalling strength of the sub-channels, forthcoming link breakages are more predictable and the pro-active routing maintenance procedures can reduce the packet drop rate and increase the system throughput.

2) **Interfering nodes are approaching**

In the proposed subcarrier allocation scheme, the subcarriers can be reused by different nodes as long as they are away from each other by a certain distance, so that the two sessions will not interfere with each other. However, in mobile ad hoc network, every node within the network can move arbitrarily. So if two nodes with the same sub-channels being used are approaching each other, the interference between then will increase. In here, the *high_interference_trigge* threshold is introduced to measure the interference level of specific subcarriers. If the interference level of the receiving sub-channels which are being used for the current session is greater than the *high_interference_trigger*, then the node starts to check its local *Free_Channel* table to search for available sub-channels for the current session. Once it finds the alternative sub-channels, a *SREA* message will be created and sent back to the transmitter to indicate the new transmission sub-channels. If there is no available sub-channel in the *Free_Channel* table, a *RERR* message will be created and send back to the source node to launch a new route discover process.

Once a node receives the *SREA* message, it will check the reallocated sub-channels. If it accepts this assignment, a *SRRP* message will be generated and sent back to the node where *SREA* packet came from to confirm the reallocation. After that, the node will forward the data packets to the next hop through the new allocated sub-channels.

4.5 Concluding remarks

In this chapter, a signal strength based medium access protocol (SSMAP) and a SSMAP based QoS routing protocol are proposed aiming to reduce the overall signalling overhead and increase the system throughput. Section 4.2 describes the system architecture and the assumptions. Section 4.3 describes the SSMAP in details. Firstly, the effectiveness of the traditional *RTS/CTS* mechanism is investigated in detail. Based on the drawbacks of the *RTS/CTS* scheme, the SSMAP proposes a new sub-channel allocation scheme which can effectively overcome the hidden and exposed nodes problem in ad hoc networks. In order to maximize the system throughput and reduce the co-channel interference, the sub-channel selection criterion is based on the current interference level of each sub-channel. The one with the least interference level has the highest priority to be selected.

In section 4.4 the SSMAP based QoS routing protocol is described and explained. Firstly the signalling messages used for route discovery and maintenance as well as the information kept in each node are described in details. Secondly, how the sub-channel allocation mechanism is performed within the route discovery process is presented. The route discovery process is described as an event driven system that each action is triggered by an event from inside or outside of the node. Finally, a signal strength based

pro-active route maintenance scheme is proposed which is developed to reduce the packet drop rate and perform fast route recovery process. In the next chapter, a time and synchronization is presented to further support the proposed SSMAP based QoS routing protocol.

Chapter 5

A new time and frequency synchronization scheme in OFDMA based wireless networks

5.1 Introduction

As increasing demand for high-rate multimedia wireless communications, many current wireless standards have adopted a multicarrier air interface based on orthogonal frequency division multiplexing (OFDM). In OFDM systems, the high data-rate stream is splitted into multiple substreams which are mapped to orthogonal subcarriers through IFFT transform. These subcarriers are grouped together to form an OFDM symbol and transmitted in parallel. These subcarriers can be easily separated in the receiver side through a FFT transform (theoretically in absence of frequency and timing errors). However, because of the frequency offset between the transmitter and receiver's local oscillator as well as the Doppler effect due to the movement of either transmitter or receiver, the orthogonality among the OFDM subcarriers is destroyed, resulting in inter-carrier interference (ICI). Moreover, the receiver should correctly detect the beginning

of a new OFDM symbol, so that, it can align the FFT window to the correct position, in order to make sure all of the data samples are coming from the same OFDM symbol. Otherwise, it will cause inter-symbol interference (ISI). However, there is usually some tolerance for symbol timing error by using a cyclic prefix between adjacent OFDM symbols. With frequency and symbol timing offsets, the system performance can be largely degraded. Cimini [50] has shown that in order to maintain a signal-to-interference ratio of 20 dB the frequency offset should be less than 4% of the inter-carrier spacing.

The orthogonal frequency division multiple access (OFDMA) combines frequency division multiple access (FDMA) and OFDM protocols. OFDMA divides the total subcarriers into multiple sub-channels which can be assigned to different users to realize concurrent transmission. Inherent from OFDM, one of the prominent technical challenges in OFDMA design is related to stringent requirements for timing and frequency synchronization. Similar to OFDM, OFDMA is extremely sensitive to frequency and timing errors. The frequency offset between the incoming waveform and the local oscillator will destroy the orthogonality of subcarriers and produce ICI as in OFDM. Symbol timing error should be compensated at the receiver to avoid inter-symbol interference. In OFDMA systems, the frequency and timing synchronizations are even more difficult than in OFDM systems since in the uplink the received signal is a combination of multiple waveforms from different transmitters with different frequency and timing errors.

OFDM/OFDMA have been standardized in many commercial systems [10]-[13] and many corresponding synchronization algorithms have been proposed in recent years

[87]-[101]. However, there are only few proposals for implementing OFDMA in an Ad hoc wireless system and all of them do not consider the synchronization issues. This is because in a mobile ad hoc network the frequency and timing synchronization are much more difficult than in an infrastructure based wireless system. So far, to the author's knowledge, the only proposal for synchronization in mobile ad hoc networks is [51]. However, they only consider the situation when a new user enters into the network and it needs to synchronize with the leader. How the network synchronizes with each other from the beginning is not mentioned. Moreover, the authors do not solve the problem of dealing with the multiple time and frequency errors from multiple transmissions. A SSMAP based QoS routing protocol in OFDMA based ad hoc network has been presented in the previous chapter. However, the proposed routing protocol needs a proper synchronization in the physical layer. Therefore, how to perform the synchronization in each mobile node to guarantee the feasibility of the proposal is a crucial issue. In this chapter, a proposed synchronization solution is presented cooperating with the proposed SSMAP to realize a complete cross layer solution. Before going to the details of the proposed synchronization solutions, a literature review about the recent popular synchronization solutions on OFDM/OFMDA is fully investigated.

➢ **Notations**

The following notations are used throughout this chapter. The superscripts $(*)^T$, $(*)^H$, and $(*)^*$ stand for transposition, conjugate transposition, and element-wise conjugation, respectively. $E\{*\}$ denotes the expectation operator. $\Re\{*\}$ and $\mathfrak{H}\{*\}$ denote the real and imaginary components of a complex-valued quantity respectively, while $arg\{*\}$ and $|*|$ are used for the corresponding argument and amplitude. *N* represents the

total number of sub-carrier for one OFDM/OFDMA symbol and *S* represents the total number of sub-channels.

5.2 Timing and frequency synchronization in OFDM systems

In OFDM systems, the transmission in many standards is normally organized in frames which consist of several OFDM blocks as well as some reference blocks with a particular training pattern appended in front of the data segment to assist the synchronization process. An example of the frame architecture is shown in Figure 29 [55].
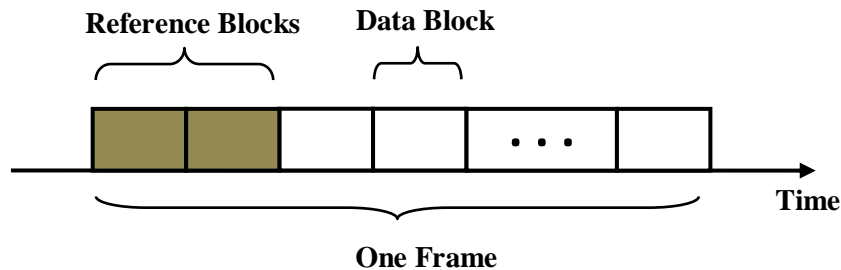
Figure 29 An example of frame structure

The synchronization process in OFDM system is normally carried out in two phases: acquisition and tracking. In the first phase, initial time and frequency estimation take place by exploiting the reference blocks to get coarse estimates of the synchronization parameters [52]-[63]. These estimates are further refined during the tracking phase to overcome the short-term variations introduced by the oscillator drift or by the time varying Doppler shifts. For this purpose, many synchronization techniques have been proposed. These synchronization techniques can be classified into two categories: data-aided (DA) and non-data-aided (NDA) synchronization schemes. The DA technique exploits the redundancy of the cyclic prefix or pilot subcarriers inserted in each OFDM block to estimate the frequency and timing errors [64]-[68]. The latter can be referred

to so-called blind estimation methods which can be further divided into pre-FFT blind estimation [69][70] and post-FFT blind method [71]-[81]. All of these techniques are proposed for single user OFDM systems. Synchronization techniques for multiuser OFDMA systems will be described in detail in section 5.3.

a. Timing Acquisition

Timing acquisition is usually the first step of the synchronization process. There are two main tasks. The first one is to detect a new received frame in the received data stream. Second, once a new frame is detected, it calculates the coarse timing error so that the FFT data window can be aligned correctly with the useful part of the symbol. Since the frequency offset has not been compensated at this stage, the timing recovery scheme is desirable to be robust against possibly large frequency offsets.

One of the first timing acquisition algorithms for OFDM transmission was proposed by Nogami and Nagashima (N&N) [82]. N&N was based on the use of a null reference block appended in the beginning of a frame. There is nothing to be transmitted in the reference block. The receiver will continuously detect the energy of the incoming data frame. The drop of the received power indicates arrival of a new frame. Unfortunately, N&N provides highly inaccurate timing estimation. Moreover, it is not suitable for burst-mode transmission since the null reference block cannot distinguish from the idle period between the successive bursts. A popular method to overcome these problems is using a reference block which consists of two identical halves. The accurate timing estimation can be achieved by searching for the peak of the correlation among the two identical halves. This idea was originally proposed by Schmidl and Cox (S&C) [53], where a

reference block with two repetitive parts of length *N/2* (N is the length of one OFDM block) is transmitted at the beginning of each frame.

As long as the cyclic prefix is longer than the channel impulse response, the two identical half references will remain identical after passing through the transmission channel unless there is a phase shift caused by frequency offset. Since the first half of the block is the same as the second half, the multiplying of the conjugate of a sample from the first half with its corresponding identical sample from the second half will have a phase of approximately $\theta = \pi T \Delta f$, where T is the duration of the reference block and Δf is the frequency offset. The product of each pair of these samples will have approximately same phase. Therefore, the magnitude of the sum will be a large value. In particular, assuming there are L complex samples in each of the half of the reference symbol, the sum of the pairs of products is:

$$P(d) = \sum_{m=0}^{L-1} r^*(d+m) \times r(d+m+L) \tag{5.1}$$

where $r(t)$ is the samples of the received signal at time instant t. d represents the time index corresponding to the first sample in a window of 2L samples. This window slides along in time as the receiver searches for the first training symbol. The timing metric to estimate the sampling starting point d is defined as:

$$\Gamma(d) = \frac{\sum_{m=0}^{L-1} r^*(d+m) \times r(d+m+L)}{\sum_{m=0}^{L-1} |r(d+m+L)|^2} \tag{5.2}$$

where $\sum_{m=0}^{L-1} |r(d+m+L)|^2$ is the received energy for the second half reference symbol. The resulting timing estimate is thus given by [53]:

$$d = arg \max_{d} \{|\Gamma(d)|\} \tag{5.3}$$

$|\Gamma(d)|$ is continuously monitored and the start of a frame is declared whenever it exceeds a pre-defined threshold. Once the presence of a new frame is detected, timing estimate d is computed by searching for the maximum of $\Gamma(d)$ as indicated in (5.3).

One of the drawbacks of the S&C algorithm in [53] is that the timing metric reaches a plateau which has a length equal to the length of the guard interval minus the length of the channel impulse response since there is no ISI within this plateau to distort the signal. This plateau may largely reduce the estimation accuracy. One of the solutions to overcome this problem is proposed by Shi and Serpedin (S&S) in [63]. S&S used a training block which consists of four repetitive parts [+B +B −B +B] with a sign inversion in the third segment. The training block architecture can be seen in Figure 30.
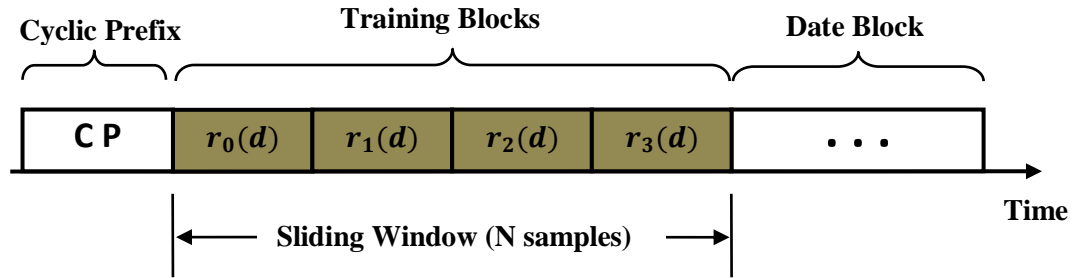
Figure 30 Training symbol for S&S algorithm

As shown in Figure 30, a training block of length N is composed of four vectors of samples $r_j(d) = \{r\left(l + \frac{jN}{4} + d\right); 0 \leq l \leq \frac{N}{4} - 1\}$ with $j = 0,1,2,3$. The timing metric is now calculated as:

$$\Gamma(d) = \frac{|\Lambda_1(d)| + |\Lambda_2(d)| + |\Lambda_3(d)|}{\frac{3}{2}\Sigma_{j=0}^{3} || r_j(d)||^2} \tag{5.4}$$

115

Where: $|\Lambda_1(d)| = r_0^H(d)r_1(d) - r_1^H(d)r_2(d) - r_2^H(d)r_3(d)$

$|\Lambda_1(d)| = r_1^H(d)r_3(d) - r_0^H(d)r_2(d)$

$|\Lambda_1(d)| = r_0^H(d)r_3(d)$

With this design, the plateau region in S&C timing metric is now significantly reduced. More accurate timing estimates are expected. The sharpness of the timing metric can be further improved by a reference block with more than four repetitive segments as indicated in [62].

Besides the reference block based timing acquisition algorithms described above, blind methods are preferable in a continuous-mode transmission since any unnecessary overhead can be avoided. An example of a blind scheme can be found in [66], where autocorrelation properties induced by the CP on the time-domain samples are exploited for timing estimation. Specifically, the following N-lag autocorrelation function is used as the timing metric:

$$\Gamma(k) = \sum_{q=0}^{N_g-1} r(k-q)r^*(k-q-N) \tag{5.5}$$

where k is the time index of the last received sample. Since the CP is just a duplication of the last N_g samples of each OFDMA block, it is expected that the magnitude of $\Gamma(k)$ may exhibit periodic peaks whenever samples $r(k-q-N)$ with $0 \leq q \leq N_g - 1$ belong to the CP. The peak locations indicate the beginning of the received blocks and are exploited to control the position of the DFT window.

b. Fine timing tracking

After the timing acquisition, the receiver can correctly align its FFT window to the proper position to avoid inter-symbol interference. However, due to the oscillator drift and the instability of the sampling clock, there still is a small fractional timing error for the rest of the signal. Fortunately after timing acquisition, the rest timing error is relatively small compared to the length of the cyclic prefix. If the remaining timing error $\Delta\theta$ belongs to interval $-N_g + L_c + 1 \leq \Delta\theta \leq 0$ where N_g is the length of the cyclic prefix and L_c is the length of channel impulse response, this situation only results in a cyclic shift of the received OFDM block. Recalling the time-shift property of the Fourier transform, after FFT transform the time error only appears as a linear phase across subcarriers. It is like the phase shifts caused by the channel and can be easily compensated by the channel equalizer [83].

C. Frequency Acquisition

After the detection and timing acquisition, each terminal must compute a coarse frequency estimation to compensate the frequency offset between the local oscillator and the received carrier frequency. One common approach is to employ a training block composed of some repetitive parts which remain identical after passing the transmission channel with the help of cyclic prefix except for a phase shift produced by the frequency error [52]-[63]. This method is originally proposed by Moose in [52], where the phase shift between two successive identical blocks is measured after the FFT output. In particular, assuming that the time synchronization has already been done and let $R_1(n)$ and $R_2(n)$ be the nth FFT outputs corresponding to the two reference blocks which can be expressed as:

$$R_1(n) = S(n) + V_1(n) \tag{5.6}$$

$$R_2(n) = S(n)e^{\frac{j2\pi\varepsilon N_T}{N}} + V_2(n) \tag{5.7}$$

where $S(n)$ is the signal component of the repetitive reference pattern which is the same for both reference blocks while $V_1(n)$ and $V_2(n)$ are noise terms. N_T and N represent the length of one OFDM block with and without cyclic prefix respectively. From the above equations, an estimate of frequency error ε can be calculated as [52]:

$$\varepsilon = \frac{1}{2\pi(\frac{N_T}{N})} \arg\left\{\sum_{n=0}^{N-1} R_2(n)R_1^*(n)\right\} \tag{5.8}$$

However, since the arg{} function returns values only in the range $[-\pi, \pi)$, $|\varepsilon| \leq N/(2N_T)$ which is less than one-half of the subcarrier spacing. Therefore, any frequency error outside the range cannot be recovered. An improving method to enlarge the acquisition range is proposed by Schmidl & Cox (S&C) in [53]. S&C decomposes the frequency offset into two parts. One is a fractional part which is less than one subcarrier spacing $1/(NT_s)$ in magnitude and the other is an integer part which is a multiple of $2/(NT_s)$ where T_s is one sample duration in one OFDM block. The normalized frequency error is thus rewritten as:

$$\varepsilon = v + 2\eta \tag{5.9}$$

where $v\epsilon(-1,1]$ and η is an integer. The S&C estimator relies on the transmission of two reference blocks. The first one consists of two identical halves which are used for the time acquisition. The second block contains a differentially encoded pseudo-noise sequence PN_1 on even subcarriers and another pseudo-noise sequence PN_2 on odd subcarriers. Since the main difference between the two halves of the first reference block is a phase difference of $\theta = \pi T v$ which can be estimated from equation (5.10) [53]:

$$\hat{\theta} = angle\big(P(d)\big) = angle\big(\sum_{m=0}^{L-1} r^*(d+m) \times r(d+m+L)\big) \qquad (5.10)$$

near the best timing point. The fractional frequency offset can be calculated as [53]:

$$\hat{v} = \frac{\hat{\theta}}{\pi T} = \frac{angle(\sum_{m=0}^{L-1} r^*(d+m) \times r(d+m+L))}{\pi T} \qquad (5.11)$$

The next step is to estimate the integer part of the frequency error. For this purpose, the samples from the two reference blocks are first compensated by the fractional frequency offset estimate, by multiplying the signal by $exp(-\frac{j2\pi\hat{v}}{N})$. Next, they are fed to the FFT unit. Let their FFT's output be $R_1(n)$ and $R_2(n)$ $for\ 0 \le n \le N-1$. Assuming perfect compensation of v, the FFT outputs are not affected by ICI since the uncompensated integer frequency offset will only result in phase shift in frequency domain by a quantity 2η. The FFT outputs of these two reference blocks can be expressed as:

$$R_1(n) = H(n-2\eta)d_1(n-2\eta) + V_1(n) \qquad (5.12)$$

$$R_2(n) = H(n-2\eta)d_2(n-2\eta) + V_2(n) \qquad (5.13)$$

where $H(n)$ is the channel response after FFT and $d(n)$ is the data vector. Re-calling that the differentially encoded PN sequence p(n) on the even subcarriers of the second block, $p(n) = d_2(n)/d_1(n)$. From (5.12) and (5.13) by neglecting the noise terms for simplicity, One can derive that:

$$R_2(n) = p(n-2\eta)R_1(n) \qquad (5.14)$$

for any even n. Therefore, an estimate of η can be calculated as [53]:

$$\hat{\eta} = \max_{\hat{\eta}} \frac{|\sum_{n \text{ even}} R_2(n)R_1^*(n)p^*(n - 2\eta)|}{\sum_{n \text{ even}} |R_2(n)|^2} \tag{5.15}$$

d. Frequency tracking

The coarse frequency offset estimate $\hat{\varepsilon}$ obtained from the frequency acquisition phase is applied to the received samples *r(k)* to produce the new sequence $r'(k) = r(k)e^{-j2\pi k \hat{\varepsilon}/N}$. Because of the time-varying Doppler shifts, the residual frequency error $\Delta\varepsilon = \varepsilon - \hat{\varepsilon}$ needs to be continuously tracked and compensated in order to avoid ICI at theFFT output. This operation is usually accomplished on a block-by-block basis by resorting to a closed-loop structure. The system architecture can be seen in Figure 31. Here, $r_i'(k)$ $(-N_g \leq m \leq N - 1)$ are the coarse compensated time-domain samples including CP of *ith* received OFDM block. e_i is an error signal that provides information on $\Delta\varepsilon_i$. The residual frequency error is updated block by block according to the following recursion:

$$\Delta\hat{\varepsilon}_{i+1} = \Delta\hat{\varepsilon}_i + \alpha e_i \tag{5.16}$$

Where $\Delta\hat{\varepsilon}_i$ is the estimated residual frequency error over the ith block and α is a step-size which is needed to design properly to achieve a reasonable tradeoff between convergence speed and accuracy in the steady state. The input samples $r_i'(k)$ is finally compensated with the updated $\Delta\hat{\varepsilon}_i$ to form the final FFT inputs $x_i(k)$.
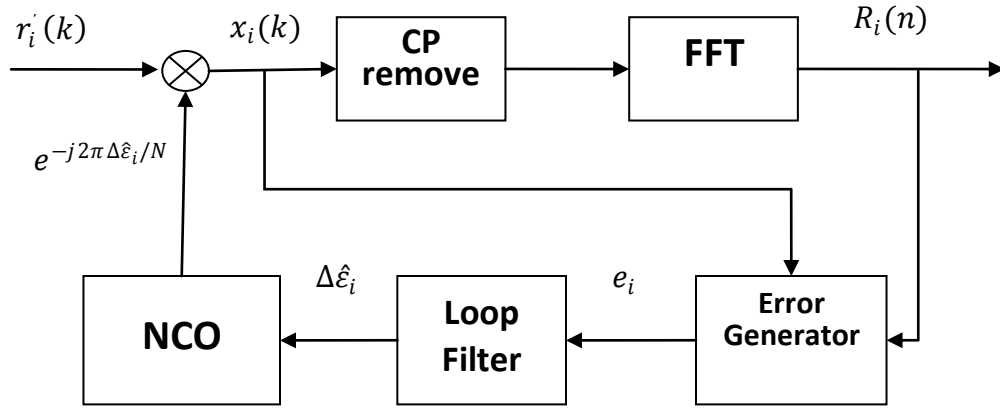
Figure 31 The framework for closed-loop fine frequency estimation

Several frequency tracking protocols proposed in the literature are based on the closed-loop structure in Figure 31, where the NCO stands for number-controlled oscillator. The only difference among them is the specific error signal e_i [65]-[70]. These frequency tracking schemes can be categorized into frequency domain and time domain algorithms, depending on how the e_i is calculated from $x_i(k)$ or from $R_i(n)$. A frequency domain scheme is given in [69], where e_i is derived using a Maximum likelihood approach and reads:

$$e_i = \Re\{\sum_{n=1}^{N-1} R_i^*(n)[R_i(n+1) - R_i(n-1)]\} \tag{5.17}$$

where $\Re\{*\}$ represents the real part of the inside formula. A similar approach with improved performance is proposed in [70], which employs the following error signal:

$$e_i = \Re\{\sum_{n=1}^{N-1} \frac{R_i^*(n)[R_i(n+1) - R_i(n-1)]}{1 + \rho|R_i(n)|^2}\} \tag{5.18}$$

where ρ is a design parameter related to the operating SNR. The time-domain schemes can be found in [65][66], where the phase shift between the CP and the last N_g samples of each block is used as an indicator of the residual frequency offset. The resulting error signal is given by:

$$e_i = \frac{1}{N_g} \mathfrak{H}\{ \sum_{m=-N_g}^{-1} x_i^*(m)x_i(m+N)\} \tag{5.19}$$

where $x_i(m)(-N_g \leq m \leq -1)$ are samples taken from the CP of the ith received OFDM block and $\mathfrak{H}\{*\}$ represents the imaginary components of a complex-valued quantity. *N* is equal to the number of subcarriers for one OFDM symbol.

5.3 Timing and frequency synchronization in OFDMA systems

The downlink timing and frequency synchronization for OFDMA systems is similar to the way in single-user OFDM systems. Therefore, this report mainly focuses on the timing and frequency synchronization schemes in the OFDMA uplink. Multiuser uplink synchronization is a much more difficult task than in corresponding downlink situation since the uplink received signal at the BS is a combination of all of the users' signals each of which has distinct timing and frequency errors. Accordingly, the BS has to estimate much more parameters than in the downlink. Moreover, the frequency and timing offset compensation are different than that in downlink. In the downlink situation, the frequency correction is easily accomplished by multiplying the downlink time domain signal by $e^{2\pi\widetilde{\varepsilon_m}/N}$ and timing adjustment is achieved by shifting the FFT window by $\widetilde{\theta_m}$ sampling interval. However, because the uplink received signal is composed of multiple users' signals with different frequency and timing error, compensation of one user's

timing and frequency offset will affect the other initially aligned users. Based on this consideration, this report categorizes the uplink timing and frequency synchronization scheme into two phased timing and frequency offset estimation and timing and frequency offset compensation.

a. Timing and frequency offset estimation

The most common methods to simplify the synchronization process in the uplink transmission are to use sub-band subcarrier allocation strategy where each sub-channel contains adjacent subcarriers. A specified number of null subcarriers is inserted in the beginning of each sub-channel to provide adequately large guard intervals. If the frequency offset is significantly small compared to the guard interval, users' signals can be easily separated by using a bank of digital band-pass filters. This filtering operation allows the BS to perform the timing and frequency estimation independently for each user. Therefore, the time and frequency estimation can be obtained by using the similar approaches in the downlink transmission. One possibility is using the method mentioned in [66] by exploiting the CP. In particular, assuming that the output signal from the filter tuned on the mth sub-channel is denoted as $x_m(k)$, the timing and frequency estimates can be expressed as:

$$\widetilde{\theta_m} = arg \max_{\tilde{\theta}}\{|\Gamma_{\mathrm{m}}(\tilde{\theta})|\} \tag{5.20}$$

$$\widetilde{\varepsilon_m} = \frac{1}{2\pi} \arg\{\Gamma_m(\widetilde{\theta_m})\} \tag{5.21}$$

Where:

$$\Gamma_{\mathrm{m}}(\tilde{\theta}) = \sum_{k=\tilde{\theta}-N_g}^{\tilde{\theta}-1} x_m(k+N)\, x_m^*(k) \tag{5.22}$$

Is the N-lag autocorrelation of sequence $x_m(k)$. Another proposal based on this is presented in [84] which shows that the estimator's performance depends heavily on the number of the subcarriers in one sub-channel. The performance will deteriorate as the number of subcarrier decreases because of the increased correlation among samples $x_m(k)$. It also points out that the accuracy of the timing and frequency estimates can be improved by averaging $\Gamma_{\mathrm{m}}(\tilde{\theta})$ over several successive blocks which can be expressed using mathematics as:

$$\widehat{\Gamma_{\mathrm{m}}}(\tilde{\theta}) = \sum_{q=0}^{Q} \Gamma_{\mathrm{m}}(\tilde{\theta} + qN_T) \tag{5.23}$$

which is used in equation (5.20) (5.21) in place of $\Gamma_{\mathrm{m}}(\tilde{\theta})$.

Another timing and frequency estimation scheme based on sub-band subcarrier allocation strategy is proposed in [85]. This method utilizes the null subcarriers which are inserted in the beginning of each sub-channel to estimate the time and frequency offset. The basic idea behind this method is to estimate the timing and frequency offset to minimize the energy of the FFT outputs of the null subcarriers which can be expressed as:

$$\left(\widetilde{\theta_m}, \widetilde{\varepsilon_m}\right) = \arg \min_{\widetilde{\theta_m}, \widetilde{\varepsilon_m}} \{J(\widetilde{\theta_m}, \widetilde{\varepsilon_m})\} \tag{5.24}$$

where $J(\widetilde{\theta_m}, \widetilde{\varepsilon_m})$ is the cost function representing the energy falling in the null subcarriers and $\widetilde{\theta_m}, \widetilde{\varepsilon_m}$ are trial values of timing and frequency offsets respectively. One

of the drawback of this method is that perfect users' separation cannot be achieved by using band-pass filter in presence of frequency offsets and the idea brickwall filters is not possible in practice. One improved method based on [85] is presented in [86].

In [86] the author also proposed a null-subcarrier based carrier frequency offset estimation for OFDMA uplink systems. It assumes that each sub-channel consists of contiguous subcarriers and the null subcarriers are placed in the middle of each sub-channel. The sub-channel architecture is shown in Figure 32. It also assumes that all the users are synchronized in time so that the time offset for each user is perfectly compensated. The main idea behind the proposed carrier frequency offset (CFO) estimation is that it places the null subcarriers in the sub-band in such a way that the multiuser interference is minimized and by calculating and minimizing the energy falling in the null subcarriers in each sub-channel, the frequency offset for each user can be estimated without separating the received signal using band-pass filter. However, this algorithm can only apply to the situation that there is no timing offset. Moreover, the author did not mention how these estimated frequency offsets can be compensated without separating the users' signals.

The synchronization algorithms described above are all based on the sub-band subcarrier allocation strategy (SAS). One of the prominent advantages of SAS is that it offers the possibility of successfully separating the signals from different users through a simple filter bank even in a completely asynchronous scenario with arbitrarily large timing errors. Many other synchronization schemes are also proposed for other SAS. However, synchronization in other SAS especially in generalized SAS is much more difficult than sub-band SAS.
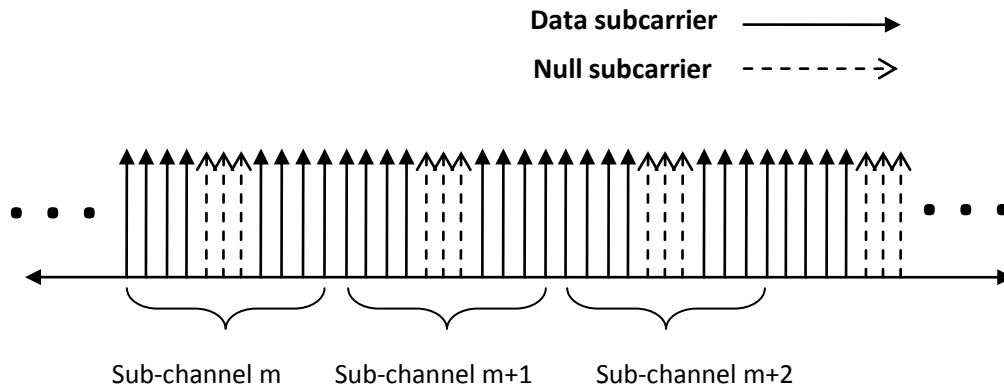
Figure 32 The sub-channel architecture with null subcarriers in the middle

Pengfei and Morelli [87] proposed a carrier frequency offset tracking scheme for an OFDMA based IEEE 802.16e uplink by exploiting a set of pre-located pilot tones in each users sub-channel. It assumes that a course frequency offset estimation has already been done in the BS and it only focus on the remaining frequency offsets introduced by a consequence of estimation errors and/or time-varying Doppler shifts caused by terminals' mobility. The total subcarriers are divided into K sub-channels and each sub--channel has 48 data subcarriers and 24 fixed-position pilots. Any sub-channel is constructed from six tiles. Figure 33 illustrates the structure of a given tile. From Figure 33 it can be seen that the pilot tones are inserted at the tile edges in each of the outer most blocks. The least-squares algorithm is used as the main frequency offset tracking manner. The main drawback of this method is the use of large number of pilot subcarriers which significantly reduces the system throughput.

One of the first synchronization schemes for OFDMA system with generalized SAS was proposed by Morelli in [88]. This method employs the maximum likelihood (ML)

principle to compute estimates of the timing and frequency offset of a new coming user

entering the network. However, it relies on the fact that the system is already

Uplink Tile
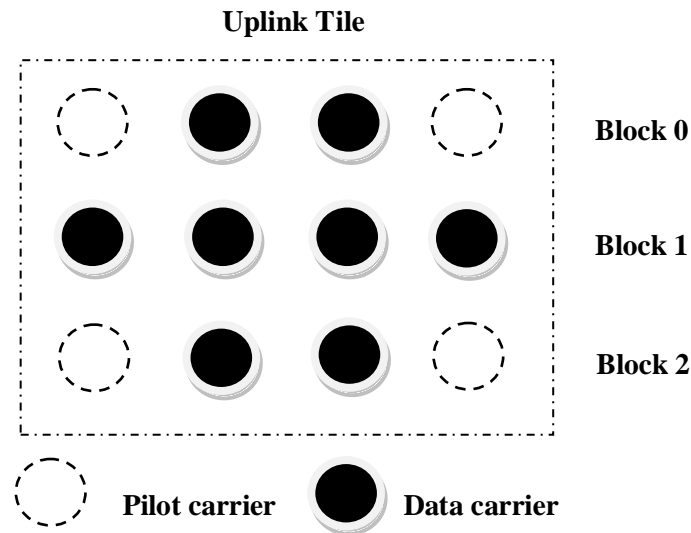
Figure 33 Tile structure proposed in [45]

synchronized and all the existing users are perfectly synchronized with each other.

Another ML based synchronization scheme for generalized SAS are described in [89] [90].

This method estimates the timing and frequency offsets based on a training block which

is transmitted by each user at the beginning of the uplink frame. However, such

solutions are too computationally demanding as they require a complete search in order

to locate the maximum of the likelihood function. An improved version has been

proposed by Na and Minn in [91]. It reduces the synchronization complexity by replacing

the exhaustive search with a line search. However, this is based on the sacrifice of

convergence speed. Based on this consideration, Luca [92] has proposed a low-

complexity scheme for frequency estimation in uplink OFDMA system. Specifically, the

transmission is organized in frames and each uplink frame is preceded by at least two

identical training blocks. The subcarriers are partitioned into disjoint sub-channels and each sub-channel is further divided into a given number of sub-bands which consists of a small group of adjacent subcarriers carrying known pilot symbols. The frequency offset can be retrieved by measuring the phase shift between the pilot tones transmitted over adjacent OFDMA blocks. It assumes a quasi-synchronous system where no interblock-interference is present at the BS receiver. In particular, denoting by $Y_j(m,n)$ the discrete FFT output over the nth subcarrier of the mth training block:

$$Y_j(m,n) = e^{\frac{j2\pi m\, \varepsilon_j N_T}{N}} X_j(n) + W_j(m,n) \tag{5.25}$$

where $X_j(n)$ is the signal component of the *jth* user over the *nth* subcarrier and $W_j(m,n)$ accounts for background noise. Then the frequency offset ε_j can be obtained as:

$$\widetilde{\varepsilon_j} = \frac{N}{2\pi N_T} \arg\{\sum_{n\epsilon\, \zeta_j} \sum_{m=0}^{M-2} Y_j(m+1,n) Y_j^*(m,n)\} \tag{5.26}$$

where ζ_j is a set of subcarriers belonging to *jth* sub-channel and *M* is the number of training blocks.

b. Timing and frequency offset compensation

As mentioned above, the timing and frequency compensation in uplink is much more complex than that in downlink since timing and frequency offset correction will influence other users. One solution is proposed in [84] that the estimated frequency and timing offsets are transmitted back to the corresponding users so that they can adjust their transmitted signals. Another solution is described in [85] where each user's signal

is separated in the BS through a bank of band-pass filter so that the time and frequency compensation can be done independently for each user in a way similar to the downlink transmission. Multiple FFT units are needed since each user needs one distinct N-point FFT operation. One of possible shortcomings of this approach is that perfect signal separation is not possible due to the non-ideal brick-wall filters and frequency leakage caused by synchronization error.

Another frequency compensation method is proposed in [93] by Choi-Lee-Jung-Lee (CLJL). The system architecture is shown in Figure 34. This solution avoids multiple FFT operations but cannot perform timing adjustment. Therefore, it can only apply to quasi-synchronous scenario where all of the users are already aligned in time. The detail algorithm is explained as follow:
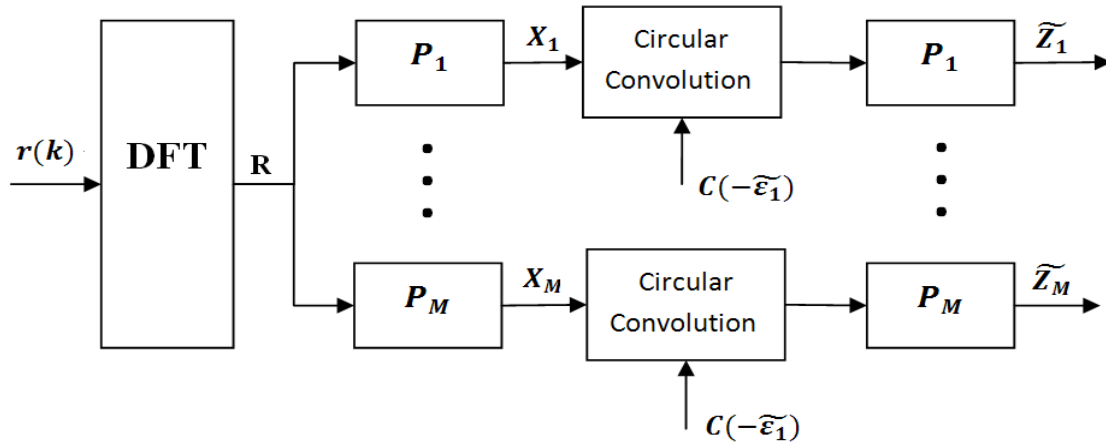
Assuming the N received samples $r_i(k)\ 0 \le k \le N-1$ belonging to the ith FFT window, which can be written as:

$$r_i(k) = \sum_{m=1}^{M} z_{m,i}(k)e^{j2\pi\varepsilon_m k/N} + w_i(k), \quad 0 \le k \le N-1 \tag{5.27}$$

where $w_i(k)$ is the thermal noise and $z_{m,i}(k)$ is the multipath distorted version of the *mth* user signal for ith FFT window. *M* is the number of users. For convenience the index *i* is omitted in the following. Since a multiplication in the time domain corresponds to a circular convolution in the frequency domain, the FFT output of $r_i(k)$ can be written as:

$$\boldsymbol{R} = \sum_{m=1}^{M} \boldsymbol{Z}_m \otimes \boldsymbol{C}(\varepsilon_m) + \boldsymbol{W} \tag{5.28}$$

where $\boldsymbol{C}(\varepsilon_m)$ is the N-point FFT of $e^{j2\pi\varepsilon_m k/N}$ and $\otimes$ represents N-point circular convolution. $\boldsymbol{Z}_m$ represents the contribution of the *mth* uplink signal to the FFT output in the absence of any interference. The goal is to recover $\boldsymbol{Z}_m$ from $\boldsymbol{R}$. CLJL achieves this through two steps as it can be seen from Figure 34. First, it computes the N-dimensional vector $\{\boldsymbol{X}_m\}_{m=1}^{M}$ for *mth* user from FFT output by selecting the entries of $\boldsymbol{R}$ with indexes $n \in \zeta_m$ while forcing the others to zero. This is accomplished by setting $\boldsymbol{X}_m = \boldsymbol{P}_m \boldsymbol{R}$, where $\boldsymbol{P}_m$ is a diagonal matrix whose *(n, n)th* entry is unitary of $n \in \zeta_m$ and is zero otherwise. Actually, $\boldsymbol{P}_m$ acts as a band-pass filter which isolates the *mth* user's FFT output from others. By ignoring for simplicity the effect of the frequency leakage, $\boldsymbol{X}_m$ can be expressed as:

$$\boldsymbol{X}_m \approx \boldsymbol{Z}_m \otimes \boldsymbol{C}(\varepsilon_m) + \boldsymbol{W}_m \tag{5.29}$$

One can easily observe that $\boldsymbol{Z}_m \otimes \boldsymbol{C}(\varepsilon_m) \otimes \boldsymbol{C}(-\tilde{\varepsilon}_m) = \boldsymbol{Z}_m$ for the perfect frequency offset estimation. Therefore, an estimate of $\boldsymbol{Z}_m$ can be obtained from the following equation [93]:

$$\widetilde{\mathbf{Z}}_m = \mathbf{P}_m [\mathbf{X}_m \otimes \mathbf{C}(-\tilde{\varepsilon}_m)] = \mathbf{Z}_m + \mathbf{P}_m [\mathbf{W}_m \otimes \mathbf{C}(-\tilde{\varepsilon}_m)] \qquad (5.30)$$

The frequency compensation scheme described above is only suitable for sub-band (SAS). This is because the bank of matrices $\mathbf{P}_m$ can successfully separate the users as long as the subcarriers from one user are grouped together and sufficiently large guard intervals are inserted among adjacent sub-channels. However, for interleaved or generalized SAS, CLJL cannot significantly reduce the multiple access interference (*MAI*) induced by frequency errors. Under this consideration, Huang and Letaief (HL) in [94] has proposed an improved version which uses an interference cancellation scheme proposed by [95]. The method operates in the following iterative fashion, where $\widetilde{\mathbf{Z}}_m^j$ is the restored signal of *mth* user after the *jth* iteration:

1) Initialization: Use the CLJL vectors defined in (5.30) as initial estimates of $\{\mathbf{Z}_m\}_{m=1}^{M}$ i.e.

$$\widetilde{\mathbf{Z}}_m^0 = \mathbf{P}_m [(\mathbf{P}_m \mathbf{R}) \otimes \mathbf{C}(-\tilde{\varepsilon}_m)] \qquad (5.31)$$

2) *jth* iteration: for each active user (m=1, 2, 3……,M), perform interference cancellation in form of:

$$\widehat{\mathbf{Z}}_m^j = \mathbf{R} - \sum_{k=1, k \neq m}^{M} \widetilde{\mathbf{Z}}_m^{j-1} \otimes \mathbf{C}(\tilde{\varepsilon}_m), 1 \leq m \leq M \qquad (5.32)$$

And then compensate for ε_m in a way similar to CLJL as:

$$\widetilde{\mathbf{Z}}_m^j = \mathbf{P}_m [(\mathbf{P}_m \widehat{\mathbf{Z}}_m^j) \otimes \mathbf{C}(-\tilde{\varepsilon}_m)] \qquad 1 \leq m \leq M \qquad (5.33)$$

5.4 The proposed time and frequency synchronization algorithm for ad hoc networks

5.4.1 The system architecture

Multiple access systems based on OFDM technique, namely OFDMA, has received a lot of attention in the past due to its capability of providing perfect elimination of multiuser interference (MUI) and inter-symbol interference (ISI) in transmissions using a simple fast Fourier transform-based receiver. However, compared to CDMA, OFDM systems are more sensitive to time and frequency synchronization errors. There are two deleterious effects caused by frequency offset; one is the reduction of signal amplitude in the output of the filters matched to each of the subcarriers and the second is the introduction of ICI from the other subcarriers which are now no longer orthogonal to the filter [96]. On the other hand, with time offset the FFT windows may not align with the right part of the received signals and in result inter-symbol interference. In the previous chapter 5.2 and chapter 5.3, a lot of synchronization schemes which are used in infrastructure based cellular networks are presented. However, the task for synchronization in mobile ad hoc networks is more difficult compared to the one in cellular networks. Firstly, complex signal processing algorithms, such as the maximum likelihood used in [88] [91], are not suitable for mobile ad hoc networks because of the hardware limitation of the mobile equipment itself. Secondly, global time synchronization is extremely difficult in mobile ad hoc networks. Therefore, each mobile terminal in the network should synchronize with each other in a fully distributed manner, which is much more complex than the base station controlled synchronization found in cellular networks. Based on these considerations, in this thesis, a time and

frequency synchronization schemes are proposed specifically for mobile ad hoc networks.

In this thesis, a Rayleigh multipath fading mode is assumed in the proposed synchronization schemes and a frame transmission is used among the mobile users. Each frame consists of one training block which is used for coarse time synchronization and channel estimation, and several data blocks. Each transmitted block contains one OFDM symbol which is composed of $N+L$ samples. As shown in the Figure 35, the first L samples are the same as the last L ones which are the cyclic prefix. The maximum number of orthogonal sub-carriers is N. The total sub-carriers are divided into S sub-channels. Each sub-channel is the minimum data transmission unit for a user to transmit data. For easily illustrating the proposed synchronization scheme, this system uses equal energy for each data sub-carriers although it can be easily extended to any other energy allocation scheme. A contiguous permutation is used in this model. In particular, each sub-channel contains a set of sub-carriers with contiguous frequencies so that the spectrum between different sub-channels is maximized. Each node can be allocated one or more contiguous sub-channels according to its QoS requirements. In the beginning of each sub-channel, there are some sub-carriers which do not carry any information symbol. These sub-carriers are called $Null$ sub-carriers. Specifically, in the pth OFDMA block, the sth sub-channel is assigned the set of frequency indexes: $(sJ + l), J = J_a + J_0$, where J_a is the number of subcarriers used in one sub-channel to carry useful information and J_0 is the number of $Null$ sub-carriers within one sub-channel. l is the index of the sub-carriers in one sub-channel which is between 0 to N/S. Figure 35 shows

the frame architecture and an example of subcarrier allocation scheme with 6 data subcarriers and 2 null subcarriers in one sub-channel.
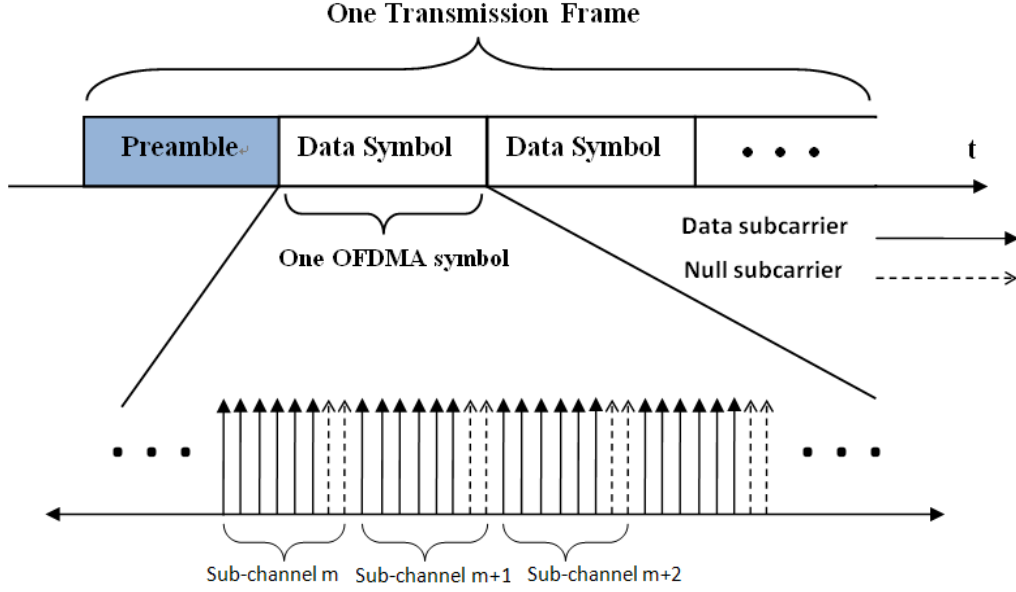
Figure 35 The frame architecture and subcarrier allocation strategy

Denoted by $U^k(p;b;l;s^k_m)$ the transmitted symbol modulated on the lth sub-carrier of s^k_mth sub-channel within the pth block of bth frame for the kth transmitter to a common receiver, where $s^k_m \in [0, S)$. k is the index of the transmitters for the same potential receiver and m is the index of the receiver. Inter-block interference is relieved by the use of a cyclic prefix that is added in the beginning of each OFDM block. The cyclic prefix has the length L that is equal to or greater than the channel order. After N IFFT mapping, and the addition of the cyclic prefix, a pth block of length $N+L$ OFDMA symbol in bth frame generated from the kth transmitter can be expressed as follow:

$$x^k(p;b;l;n) = \sum_{\{s^k_m\}} \sum_{l=0}^{J_a-1} U^k\left(p;b;l;s^k_m\right) e^{j\frac{2\pi}{N}(s^k_m J + l)n}, \quad n \in [-L, N-1] \qquad (5.34)$$

134

Where $\{s_m^k\}$ is the set of sub-channels used by the *k*th transmitter to send data to the *m*th reciever. n represents the time domain of one OFDMA sample. The total length of an OFDMA symbol is $N + L$ which consists of the duration of N data samples and the duration of L cyclic prefix. In the equation (5.34) the complex exponentials has periodicity of order N and $x^k(p; b; l; n)$ is the output OFDMA symbol from the IFFT transformation which transfers the signal from frequency domain to time domain. The last L samples of $x^k(p; b; l; n)$ is the same as the first L ones.

The system model in this thesis considers the following assumptions:

1. A mobile node can transmit or receive packets to or from multiple receivers or transmitters at the same time with different sub-channels. Simultaneous transmissions are supported in the nodes by using OFDMA.

2. Contiguous permutation is used in the system model that each sub-channel consists of contiguous sub-carriers.

3. Each mobile terminal can access to any sub-channel and can assign any sub-channels for each potential receiver.

4. There are transmit/receive duplexer filters in each mobile which can pass the desired frequency and reject as much of the undesired frequencies as possible. The cavity duplexer filter [97] is one of the options to fulfil the requirements and other duplexer filter designs are also proposed in recent years [98].
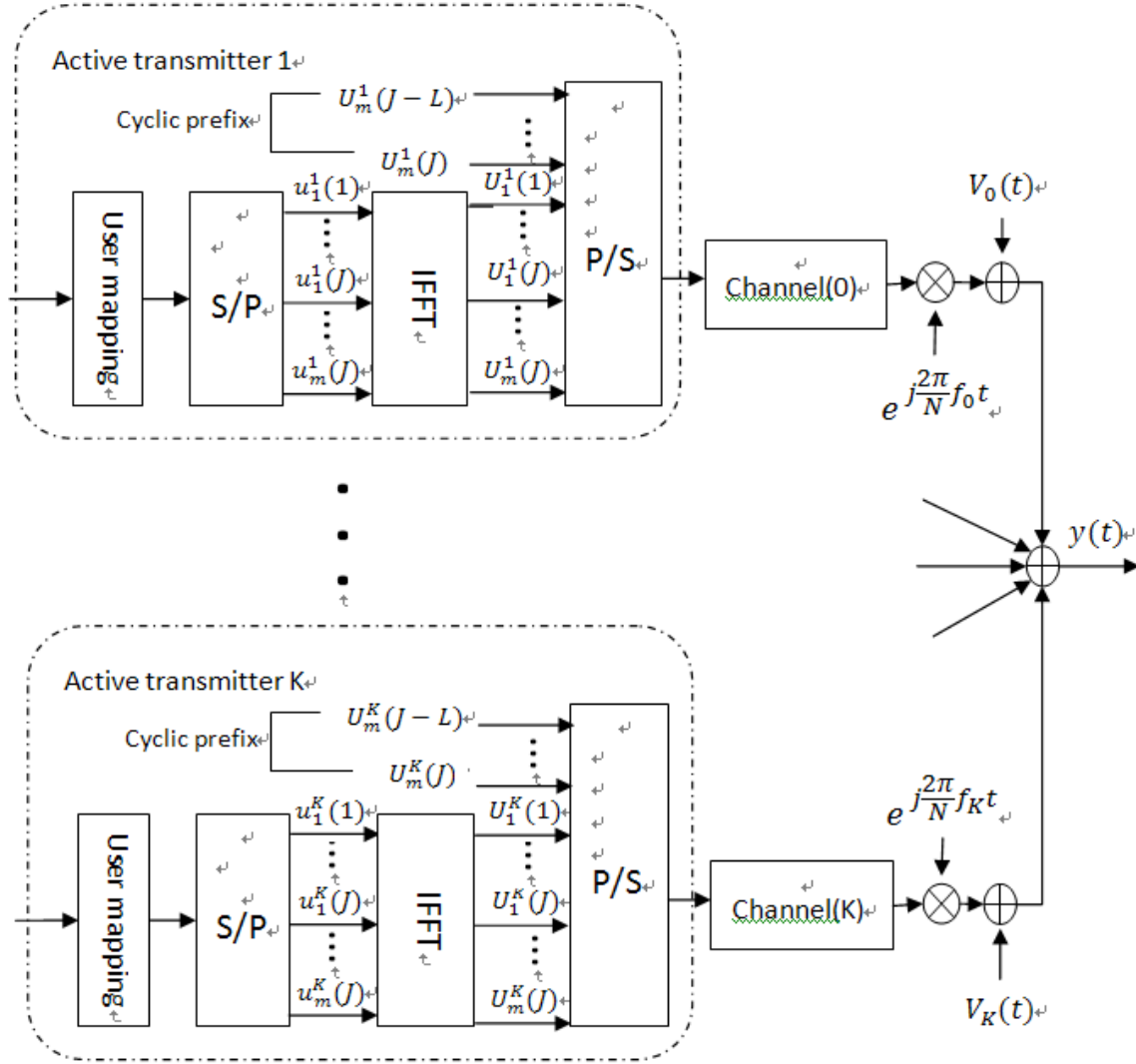
Figure 36 System architecture

5.4.2 The received signal in the receiver side and the proposed partial time synchronization scheme

As shown in Figure 36, the receiver may have total K corresponding transmitters. Each of them travels through a different channel and has distinct channel impulse response. In practice, the received signal at the receiver side is the combination of the signals transmitted from all related transmitters. Let's define $g_T(t)$ and $g_R(t)$ the pulse shaping and the match filter at the transmitter and the receiver sides respectively. $h_k(t)$ is the channel impulse response for *kth* transmitter. So the transmitter-channel-receiver filter

impulse response for *kth* transmitter is $C_k(t) = g_T(t) \otimes h_k(t) \otimes g_R(t)$, where $\otimes$ represents convolution. The ad hoc network can be based on any of the wireless ad hoc standards, such as IEEE 802.11n. In 802.11n [7], the subcarrier spacing Δf is equal to 312.5 kHz. Therefore, one OFDM symbol duration T_s is equal to $\frac{1}{\Delta f} + T_g = 3.2 \mu s + 0.8 \mu s = 4 \mu s$, where T_g is the length of the cyclic prefix. One frame duration $T_f = 9 * T_s = 80 \mu s$. The carrier frequencies f_c for 802.11n is 2.4GHz or 5GHz. Assuming the central frequency is equal to 2.4 GHz and the velocity v of the moving transmitter and receiver is 10m/s, according to [54], $B_d \leq \frac{v}{c} * f_c$, where B_d is the Doppler shift and c is the velocity of light which is $3.0 * 10^8 m/s$ in the air. Recalling that the coherence time of the channel T_c is equal to $1/B_d$ [99], it can derive that $T_c = \frac{1}{B_d} \geq \frac{c}{v*f_c} = \frac{3.0*10^8}{10*2.4*10^9} = 12.5 ms$. From the derivation above it can be seen that the channel coherence time T_c is much larger than one frame duration T_f. Therefore, for each frame, the channel impulse response can be seen as invariant.

With perfect synchronization the received baseband continuous-time waveform received by the receiver node from the *kth* transmitter can be expressed as:

$$y_k(t) = x^k(t) \otimes C_k(t) + v_k(t) = \sum_{\tau=-\infty}^{+\infty} x^k(\tau) C_k(t-\tau) + v_k(t) \qquad (5.35)$$

Where τ is the channel delay spread which can be expressed as $\tau = p(N+L)T + nT$. 1/T is the transmission rate and $v_k(t)$ is the additive noise which in this thesis is zero-mean white Gaussian noise with power spectra density σ_v^2. However, because the mismatch between the *kth* transmitter and receiver node's local oscillators as well as Doppler effect, there is a carrier offset f_k added to the received signal and the shift f_k

in the frequency domain can be seen as the time domain signal multiplied by $e^{j2\pi f_k t}$. So the equation (5.35) can be rewritten as:

$$y_k(t) = \sum_{p=-\infty}^{+\infty} \sum_{n=-L}^{N-1} x^k(p;n) \, C_k(t - p(N+L)T - nT) \times e^{j2\pi f_k t} + v_k(t) \quad (5.36)$$

The first sum in equation (5.36) operates across successive OFDMA blocks and the second sum runs within each OFDMA block. $p(N+L)T + nT$ is the channel delay. Therefore the combined received signal from all related transmitters at the receiver side is:

$$y(t) = \sum_{k=0}^{K} y_k(t)$$

$$= \sum_{k=0}^{K} \sum_{b=-\infty}^{+\infty} \sum_{n=-L}^{N-1} x^k(p;n) \, C_k(t - p(N+L)T - nT) \times e^{j2\pi f_k t} + \sum_{k=0}^{K} v_k(t) \quad (5.37)$$

Because the signal from different transmitters have different channel impulse response and frequency offsets, it is better to separate the waveform from different transmitters so that the synchronization can be performed individually. One OFDM symbol duration without cyclic prefix is NT and the sub-carrier spacing is 1/NT. Each sub-channel consists of contiguous sub-carriers and there are J_0 null sub-carriers in the end of each sub-channel. Therefore, the spectral gap between two adjacent sub-channel is J_0/NT. The minimum spectral gap between two transmitters is J_0/NT. Multiple sub-channels used by one transmitter should be adjacent to each other. Therefore, if the frequency offset of each transmitter is smaller than this spectral gap, the received signal can be successfully separated by using a set of bandpass filters. By doing this, a separate

synchronization algorithm can be applied to each transmitter. In order to successfully separate the individual signals from different transmitters, the sub-channels used by the transmitters to a common receiver must be different. The number of concurrent transmissions allowed in a node is equal to the number of bandpass filters existing in this node. The effect on the system throughput caused by the number of bandpass filters a node has is investigated later.

With the help of the bandpass filters, the signals from different transmitters do not need to arrive at the common receiver at the same time. To successfully receive an OFDMA packet the receiver needs to determine the correct symbol starting position, so that it can align the FFT window for each transmitter. This task can be achieved by using a training symbol with repetitive parts [100]. In this thesis, a frame transmission is used among the ad hoc users as described before. Each frame consists of one training block at the beginning, which is used for time synchronization and channel estimation, followed by data blocks. Each block contains one OFDMA symbol which is composed of N+L samples where L is the length of the cyclic prefix. Each transmitter will create its own training blocks using the assigned subcarriers. Without global synchronization, each node needs to continuously monitor all the sub-channels. This excessively consumes energy. In this thesis, a cross layer solution that uses a separate common signalling channel is proposed. The synchronization in the physical layer will be jointly coordinated with the proposed SSMAP based routing protocol through a cross layer signalling. The signalling common channel consists of one sub-channel which is shared by all ad hoc nodes. Two sets of transceivers are required in each node: one for the data traffic, and the other for the signalling sub-channel. The partial time synchronization scheme is

described as follow:

In an ad hoc network, each node only needs to continuously monitor the signalling channel looking for the training symbol. In the route discovery process, when a source node wants to transmit data to a destination node, each transmitter along a multi-hop path will inform its expected transmission time and its local time reference to its potential receiver. This information is encapsulated in the *RDIS* messages (section 4.4.1). Accordingly, each node will create a time reference table which records the time reference for each one hop transmitter. Since the transmission range for each node is limited (around 100m to 250m), the time offset due to the propagation delay can be neglected. In this case, global time synchronization becomes not necessary. Each node will only need to set proper filters and look for the training block for a specific sub-channel at a proper timing according to the estimated transmission time and time reference for each transmitter.

The received signal is passed through the bandpass filters for each transmitter and the receiver will get the time acquisition by looking for the training block. After successful time acquisition, the receiver will sample the filtered signal for FFT transfer. The sampling rate should be the same as the transmission rate of the transmitter which is set to *1/T*. The qth block of dth frame of the filtered received signal from the kth transmitter $y_k(t)$ collected by the receiver with sampling rate *1/T* is composed of samples taken at instant $t = (q + dB)(N + L)T + iT - \tau_k$, where τ_k is the remaining time offset from the partial time synchronization. Replacing (5.34) into (5.36), the entries of the qth block of dth frame of the received kth transmitter signal become:

$$y_k(q;d;i) = e^{j2\pi f_k((q+dB)(N+L)T+iT-\tau_k)}$$

$$\times \sum_{p=-\infty}^{+\infty} \sum_{n=-L}^{N-1} \sum_{\{s_m^k\}} \sum_{l=0}^{J_a-1} U^k(p;b;l;s_m^k)e^{j\frac{2\pi}{N}(s_m^k J+l)n}$$

$$\times C_k((q-p)(N+L)T + (i-n)T - \tau_k) + v_k(q;i) \qquad (5.38)$$

Although the proposed partial time synchronization scheme can effectively solve the global time synchronization problem to realize OFDMA in an ad hoc network, there might still be some small scale time offset due to the misalignment of the received signal with the FFT window. The time offset effects can be illustrated in Figure 37.

Figure 37 Effects of different sampling positions
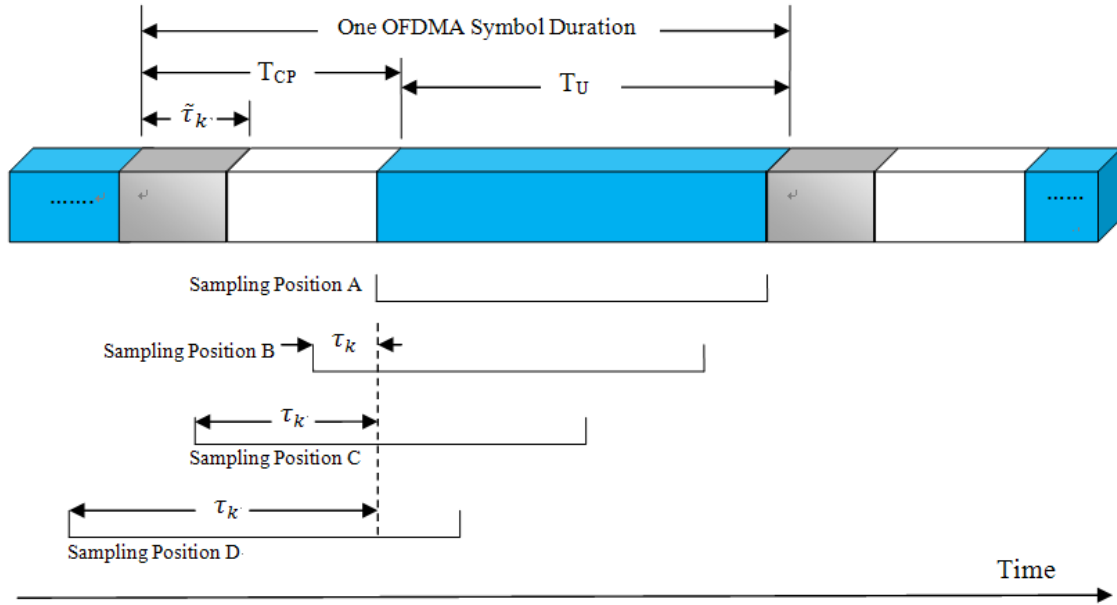
The blue part is the data symbol in one OFDMA symbol which has duration of T_U. The guard time interval (cyclic prefix duration) is T_{CP} which is equal to LT and the grey part represents the channel delay spread for the **k**th transmitter which has duration of $\tilde{\tau}_k$. The guard time interval should be set properly so that it is longer than the total channel

delay spread which means $\tilde{\tau}_k \leq T_{CP}, \forall k$. One OFDMA symbol is composed of both guard time interval and data symbol. The τ_k is denoted as time offset between **k**th transmitter and receiver. There are four possible sampling starting positions as shown in Figure 37.

- **Case one:** For sampling position *A* there is no time offset and the FFT window is exactly match the data part of the OFDMA symbol. In this case the cyclic prefix can be removed completely and there is no error on the outputs after FFT transform.

- **Case two:** For sampling position *B*, the symbol timing offset τ_k is greater than the zero but less than $T_{CP} - \tilde{\tau}_k$. The FFT samples are taken from part of cyclic prefix and part of the useful data symbol. Since the samples in cyclic prefix are copies from the tails of the useful date part. The set of samples from the data window at position B is effectively a shifted version from the original data symbol. The only effect due to this time offset is a phase shift in the frequency domain for all sub-carriers which will not cause inter-symbol interference. Therefore, the time offsets can be seen as part of the unknown channel impulse response which can be compensated through the equalization performed at the receiver. In this case, it considerably reduces the system synchronization complexity since the time offsets are incorporated as part of the channel impulse response and the only element that needs to be considered is frequency offset.

- **Case three:** For sampling position *C*, the symbol time offset is greater than $T_{CP} - \tilde{\tau}_k$. The sum of the channel delay spread and time offset exceeds the guard interval. As a consequence, some samples in the FFT window will come from the region that is corrupted by the channel delay components. This will cause inter-symbol

interference. As a result, the demodulated data from one sub-carrier is affected by adjacent sub-carriers and they are not orthogonal to each other anymore. This will lead to inter-carrier interference.

- **Case four:** For sampling position *D* the data window is covering two adjacent OFDMA symbols which will introduce strong inter-symbol interference. In this case, there is an ambiguity about which symbol is being observed. This can be resolved by using extra an OFDMA symbol detection technique introduced in [96] [101]. The effect of the time offset under this situation is beyond the scope of this thesis and will not be studied.

In ad hoc networks considered in this thesis, the transmission range for two nodes is less than 300 meters. Therefore, if a proper size of the cyclic prefix is used, the sum of the time offset and channel delay spread will be less than the guard time interval so that the time offset can be considered as part of the unknown channel impulse response. In this case, the receiver only needs to consider the frequency offset. In particular, the time offset τ_k in equation (5.38) is set to zero. As defined above, the channel impulse response for any transmission channel is less than the guard time interval. Therefore, the overall channel delay spread has at most duration *LT* which means for any *t<0 or t>LT*, the channel impulse response $C_k(t)$ is equal to zero. Therefore, in equation (5.38), for $C_k((d-b)(N+L)T + (i-n)T - \tau_k)$, $\tau_k = 0$ and the only non-zero term comes from the qth term while $q = p$ and $0 \leq (i-n)T \leq LT - 1$. Setting $r = i - n$ and removing the cyclic prefix to avoid the inter-symbol interference, the *q*th block samples of length *N* in *d*th frame from *kth* transmitter becomes:

$$y_k(q;d;i) = e^{j2\pi((q+dB)(N+L)T+iT)f_k} \times \sum_{\{s_m^k\}} \sum_{l=0}^{J_a-1} \tilde{U}^k(q;d;l;s_m^k) \times e^{j\frac{2\pi}{N}(s_m^k J+l)i}$$

$$+ v_k(q;d;i) \quad i\epsilon[0,...,N$$

$$- 1] \tag{5.39}$$

where: $\tilde{U}^k(q;d;l;s_m^k) = U^k(q;d;l;s_m^k) \times \sum_{r=0}^{L-1} C_k(rT) \times e^{j\frac{2\pi}{N}(s_m^k J+l)r}$ (5.40)

As it can be seen from equation (5.39) that the sum $\sum_{p=-\infty}^{+\infty} \sum_{n=-L}^{N-1}$ is replaced by

$\sum_{r=0}^{L-1}$ and n is replaced by $i-r$. $\sum_{r=0}^{L-1} C_k(rT) \times e^{j\frac{2\pi}{N}(s_m^k J+l)r}$ can be seen as **k**th

channel transfer function for **k**th transmitter.
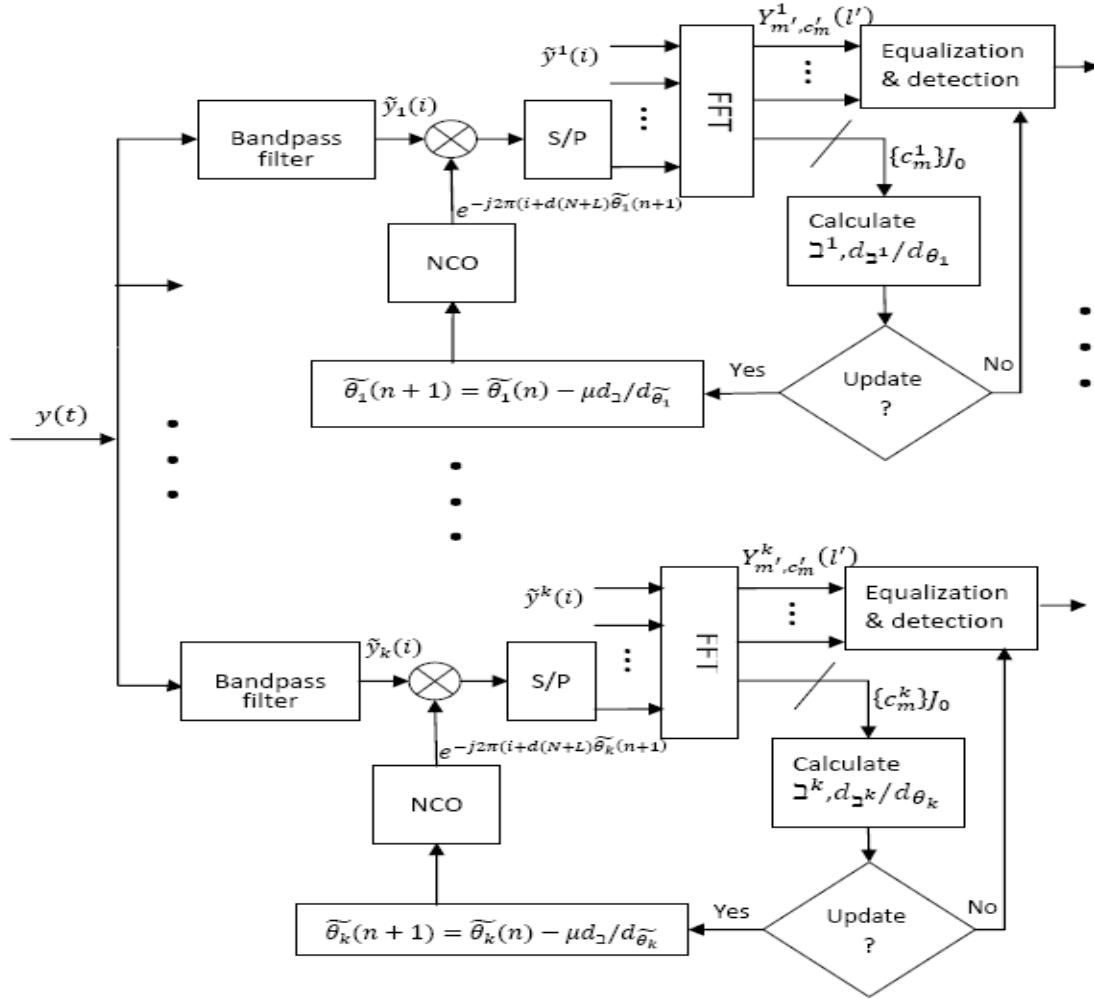
Figure 38 The receiver architecture for frequency synchronization

5.4.3 The frequency synchronization algorithm

As it can be seen from the equation (5.39) that the effect due to the frequency offset is equivalent to multiply the time domain signal by a sequence $e^{-j2\pi(iT+(q+dB)(N+L)T)f_k}$. Therefore, the main task of frequency synchronization is to find and compensate the frequency offset f_k. The proposed frequency synchronization exploits the property of the null subcarriers inserted in each sub-channel [85] and adapts to ad hoc networks. The main idea of the proposed frequency synchronization is: for each sub-channel in **k**th received OFDMA signal, there are J_0 null sub-carriers. The receiver will continuously measure the average energy falling across these null subcarriers in every sub-channel for each transmitter. Without frequency offset and noise, there will be no energy falling in these null sub-carriers. If there is some energy detected, the receiver will run an iterative search which creates an estimated frequency offset $\widetilde{f}_k(n)$ and applies it to the filtered **k**th transmitter signal by multiplying the filtered signal by the sequence $e^{-j2\pi(iT+(q+dB)(N+L)T)\widetilde{f}_k(n)}$. The initial guess of the **k**th frequency offset $\widetilde{f}_k(0)$ is set to zero. In each round the receiver will calculate the energy for the null sub-carriers and update the estimated frequency offset $\widetilde{f}_k(n)$ accordingly. The detailed frequency offset searching algorithm will be discussed below. The receiver architecture for the frequency synchronization is shown in Figure 38, where $\widehat{\theta}_k(n) = \widehat{f}_k(n) \times T$. Therefore, the compensated **k**th transmitter signal can be expressed as:

$$\tilde{y}_k(q;d;i) = y_k(q;d;i) \times e^{-j2\pi((q+dB)(N+L)T+iT)\widetilde{f}_k}$$

$$= \sum_{\{s_m^k\}} \sum_{l=0}^{J_a-1} \tilde{U}^k(q;d;l;s_m^k) \times e^{j\frac{2\pi}{N}(s_m^k+l)i} \times e^{j2\pi f_k((q+dB)(N+L)T+iT)}$$

$$\times e^{-j2\pi(iT+(q+dB)(N+L)T)\hat{f}_k} + \tilde{v}_k(q;i)$$

$$= \sum_{\{s_m^k\}} \sum_{l=0}^{J_a-1} \tilde{U}^k(q;d;l;s_m^k) \times e^{j\frac{2\pi}{N}(s_m^k+l)i} \times\times e^{j2\pi((q+dB)(N+L)T+iT)\Delta f_k}$$

$$+ \tilde{v}_k(q;i) \tag{5.41}$$

where $\Delta f_k = f_k - \tilde{f}_k$ and $\widetilde{v_k}(q,i) = v_k(q,i) \times e^{-j2\pi\widetilde{f}_k((q+dB)(N+L)T+iT)}$. Then

according to Figure 38, after passing through the FFT window the signal becomes:

$$Y^k(q;d;l';s_m^{k'}) = \frac{1}{N}\sum_{i=0}^{N-1} \tilde{y}_k(q;d;i)e^{-j\frac{2\pi}{N}(s_m^{k'}J+l')i}$$

$$s_m^{k'} \in [0,S-1], \quad l' \in [0,J-1] \tag{5.42}$$

where *S* is the total number of sub-channels in one OFDMA symbol. Substituting (5.41)

into (5.42), it becomes:

$$Y^k(q;d;l';s_m^{k'})$$

$$= \frac{1}{N}\sum_{i=0}^{N-1}\sum_{\{s_m^k\}}\sum_{l=0}^{J_a-1} \tilde{U}^k(q;d;l;s_m^k) \times e^{j\frac{2\pi}{N}(s_m^k J+l)i} \times e^{j2\pi(q+dB)(N+L)T+iT)\Delta f_k}$$

$$\times e^{-j\frac{2\pi}{N}(s_m^{k'}J+l')i} + V_{s_m^k}^{k'}(q;l')$$

$$= \frac{1}{N}\sum_{i=0}^{N-1}\sum_{\{s_m^k\}}\sum_{l=0}^{J_a-1} \tilde{U}^k(q;d;l;s_m^k)$$

$$\times e^{j\frac{2\pi}{N}(s_m^k J+l)i+j\frac{2\pi}{N}iT\Delta f_k N-j\frac{2\pi}{N}(s_m^{k'}J+l')i+j\frac{2\pi}{N}(q+dB)(N+L)T\Delta f_k N} + V_{s_m^k}^{k'}(q;l')$$

$$= \frac{1}{N} e^{2\pi(q+dB)(N+L)T\Delta f_k} \sum_{\{s_m^k\}} \sum_{l=0}^{J_a-1} \tilde{U}^k(q;d;l;s_m^k) \times \sum_{i=0}^{N-1} e^{j\frac{2\pi}{N}\left[\left(s_m^k - s_m^{k'}\right)J + (l-l') + TN\Delta f_k\right]i}$$

$$+ V_{s_m^k}^{k}{}'(q;l') \tag{5.43}$$

where $V_{s_m^k}^{k}{}'(q;l') = \left(\frac{1}{N}\right)\sum_{i=0}^{N-1} \breve{v}_k(q,i) \times e^{-j\frac{2\pi}{N}\left(s_m^{k'}J+l'\right)i}$ is the FFT output noise which

remains zero-mean white complex Gaussian with variance σ_v^2. Setting $\varpi_k =$

$[\left(s_m^k - s_m^{k'}\right)J + (l - l') + TN\Delta f_k]/N$, $e^{j\frac{2\pi}{N}\left[\left(s_m^k - s_m^{k'}\right)J + (l-l') + TN\Delta f_k\right]i}$ can be written as

$\sum_{i=0}^{N-1} e^{j2\pi\varpi_k i}$. This is the discrete version of the express $\int_0^{N-1} e^{j2\pi\varpi_k i}\, di$ which is equal

to $\frac{\sin(\pi\varpi_k N)}{\pi\varpi_k} e^{j\pi\varpi_k N}$. Thus, the equation (5.43) can be rewritten as:

$$Y^k\left(q;d;l';s_m^{k'}\right)$$

$$= \frac{1}{N} e^{2\pi(q+dB)(N+L)T\Delta f_k} \sum_{\{s_m^k\}} \sum_{l=0}^{J_a-1} \tilde{U}^k(q;d;l;s_m^k) \times \frac{\sin(\pi\varpi_k N)}{\pi\varpi_k} e^{j\pi\varpi_k N}$$

$$+ V_{s_m^k}^{k}{}'(q;l') \tag{5.44}$$

In equation (5.44), if there is no noise and there is no frequency offset, $Y^k\left(q;d;l';s_m^{k'}\right)$

is different from zero only when $s_m^k = s_m^{k'}$ and $l = l'$:

$$Y^k\left(q;d;l';s_m^{k'}\right) = \begin{cases} U^k\left(q;d;l';s_m^{k'}\right), & for\ s_m^k = s_m^{k'}\ and\ l = l' \\ 0, & otherwise. \end{cases} \tag{5.45}$$

From (5.45) it can be seen that, for perfect synchronization and no noise, for any null

sub-carriers where $U^k\left(q;d;l';s_m^{k'}\right)$ is equal to 0 there is no energy detected. However,

for non perfect synchronization, there is inter-symbol interference among OFDM

symbols. Therefore, there will be a non null energy falling in the band of null sub-

carriers. Based on this, this thesis introduces an energy detection function ב which is the summation of the energy of the null sub-carriers for the **k**th transmitter. The energy detection function calculates the energy in null sub-carriers for each OFDMA block and average over N_b blocks. The ב can be expressed as:

$$ב_{N_b}^k(\Delta f_k) = \frac{1}{N_b} \sum_{q=0}^{N_b} \sum_{s_m^{k'} \in \{s_m^k\}} \sum_{l'=J_a}^{J-1} |Y^k\left(q; d; l'; s_m^{k'}\right)|^2 \qquad (5.46)$$

Many methods can be used here to find the value of $\widetilde{f_k}$ to minimize $ב_{N_b}^k(\Delta f_k)$ such as a steepest-gradient-descent algorithm used in [85]. In this thesis, the similar approach is used to find the $\widetilde{f_k}$:

$$\widetilde{f_k} = \underset{\Delta f_k}{argmin} \; ב_{N_b}^k(\Delta f_k) \qquad (5.47)$$

In case of perfect synchronization and no noise, the value of ב is null. However, with noise and frequency offset, there is some energy falling in the null sub-carriers band. This thesis averages the energy detection function over successive OFDMA blocks. When the number of the OFDMA blocks calculated over several OFDMA blocks, the detection function becomes:

$$\beth_{N_b}^k(\Delta f_k) = \frac{1}{N_b}\sum_{q=0}^{N_b}\sum_{s_m^{k'}\in\{s_m^k\}}\sum_{l'=J_a}^{J-1}|Y^k\left(q;d;l';s_m^{k'}\right)|^2 = \sum_{s_m^{k'}\in\{s_m^k\}}\sum_{l'=J_a}^{J-1}E\{|Y^k\left(q;d;l';s_m^{k'}\right)|^2\}$$

$$= \sum_{s_m^{k'}\in\{s_m^k\}}\sum_{l'=J_a}^{J-1}E\{|\frac{1}{N}e^{j2\pi(q+dB)T(N+L)\Delta f_k}\sum_{\{s_m^k\}}\sum_{l=0}^{J_a-1}U^k\left(q;d;l;s_m^k\right)$$

$$\times\sum_{r=0}^{L-1}C_k(rT)\times e^{j\frac{2\pi}{N}(s_m^kJ+l)r}\times\frac{\sin(\pi\varpi_kN)}{\pi\varpi_k}e^{j\pi\varpi_kN}|^2\}$$

$$+ \sum_{s_m^{k'}\in\{s_m^k\}}\sum_{l'=J_a}^{J-1}E\{|V_{s_m^k}^{k'}(q;l')|^2 \tag{5.48}$$

where $E\{*\}$ is the expected value operator. According to the system model described in the beginning of this chapter, equal energy is allocated for each data sub-carriers. Therefore $E\{|U^k(q;d;l;s_m^k)|^2$ can be represented by the signal power spectra density which is equal to σ_u^2. Since $|e^{j\theta}|^2$ is equal to 1, for $\forall\theta$, the equation (5.48) can be rewritten as:

$$\beth_{N_b}^k(\Delta f_k) = N\sigma_u^2\sum_{s_m^{k'}\in\{s_m^k\}}\sum_{l'=J_a}^{J-1}\sum_{\{s_m^k\}}\sum_{l=0}^{J_a-1}|C_k\left(s_m^k;l\right)|^2\times sinc^2(\pi\varpi_kN)+\sigma_v^2\frac{J_0}{J}$$

$$= N\sigma_u^2\sum_{s_m^{k'}\in\{s_m^k\}}\sum_{l'=J_a}^{J-1}\sum_{\{s_m^k\}}\sum_{l=0}^{J_a-1}|C_k\left(s_m^k;l\right)|^2\times sinc^2(\pi\left[(s_m^k-s_m^{k'}\right)J$$

$$+ (l-l')+TN\Delta f_k]) + \sigma_v^2\frac{J_0}{J} \tag{5.49}$$

where $C_k\left(s_m^k;l\right)=\sum_{r=0}^{L-1}C_k(rT)\times e^{-j\frac{2\pi}{N}(s_m^kJ+l)r}$ and $sinc(x)$ is defined as $\frac{\sin(x)}{x}$.

From equation (5.49) it can be seen that, $\beth^k(\Delta f_k)$ is composed of all positive terms and it reaches its absolute minimum value when all the $sinc(*)$ terms are zero. This will happen only when Δf_k is equal to zero. Moreover, the additive white noise is not going

to affect the global minimum position of $ב^k(\Delta f_k)$. It only add a pedestal to the energy

detection function $ב^k(\Delta f_k)$.
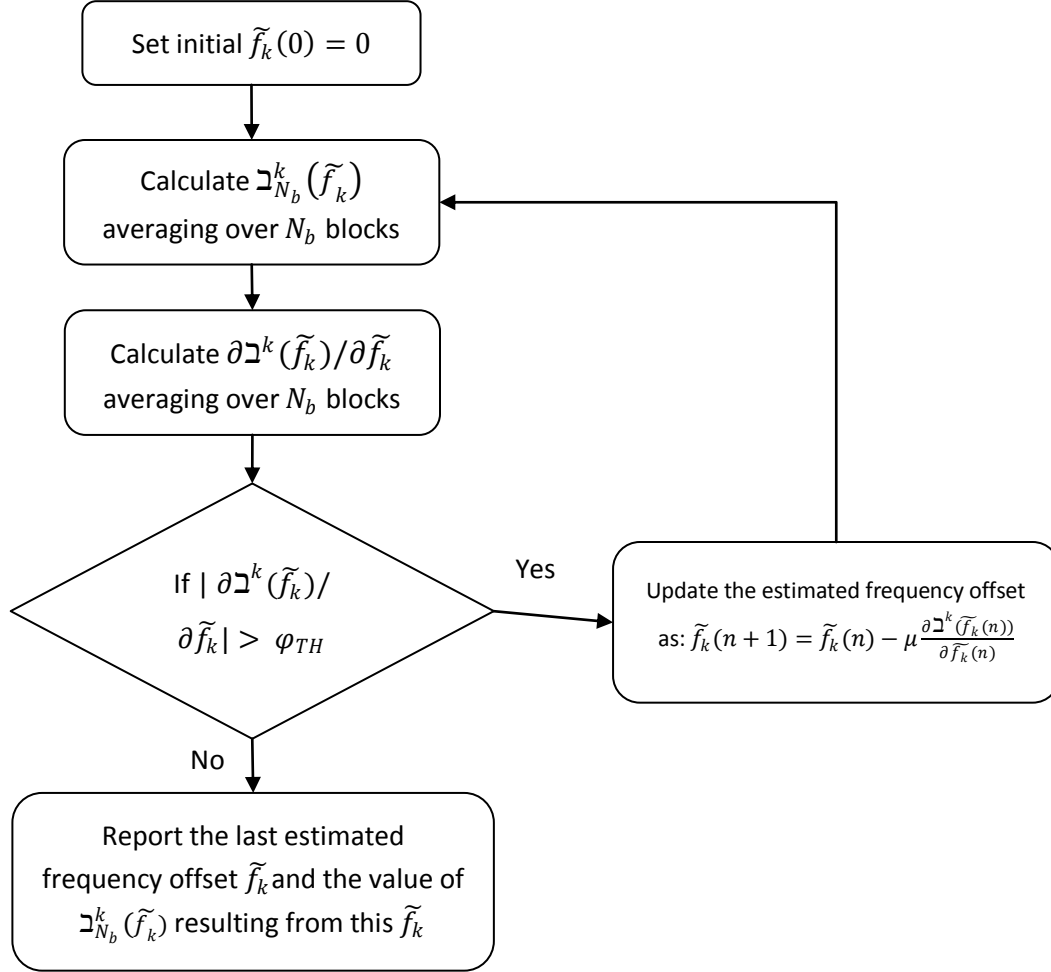
Figure 39 The flowchart for frequency offset estimation procedure

One thing has to be noted from equation (5.49) that $ב^k(\Delta f_k)$ has several minimum

points when Δf_k is integral multiple of the sub-carrier spacing *1/NT*. Therefore, any

iterative carrier offset estimation can guarantee to find the global minimum point as

long as the frequency offset is less than *1/2NT* [85]. If the frequency offset is greater

than *1/2NT*, then one has to try multiple initializations of the initial frequency offset

differing by integer multiples of *1/NT*, and choose the frequency offset which gets the minimum value of $\beth^k\left(\Delta f_k\right)$.

The main task is to find out the value of $\widetilde{f}_k$ that makes the energy detection function to reach its global minimum value. In this thesis, a conventional steepest-gradient-descent algorithm is used to find the frequency offset. Specifically, at the *nth* step, the received signal is multiplied by $e^{-j\,2\pi(iT+(q+dB)(N+L)T)\widetilde{f}_k(n)}$, where $\widetilde{f}_k(n)$ is the estimated frequency offset at step *n*. This algorithm starts with $\widetilde{f}_k(0)=0$. Each round the algorithm calculates the energy level of the detection function and its gradient $\partial\beth^k(\widetilde{f}_k(n))/\partial\widetilde{f}_k(n)$ averaging over a finite number of blocks N_b. If the modulus of the gradient exceeds a predefined threshold φ_{TH}, $\widetilde{f}_k(\mathrm{n})$ is upgraded as follows:

$$\widetilde{f}_k(n+1)=\widetilde{f}_k(n)-\mu\frac{\partial\beth^k\left(\widetilde{f}_k(n)\right)}{\partial\widetilde{f}_k(n)} \tag{5.50}$$

Otherwise the algorithm exits from the loop. The detail procedure can be illustrated in Figure 39. The step size μ is selected as a compromise between convergence speed and tracking capability. Larger μ will converge rapidly but with large variance for the frequency offset. Small μ will increase the number of repetitional iteration searching for the f_k and therefore increasing the computational complexity. The effect of different values of μ to the estimation performance will be evaluated through the simulation later. One thing has to be noted that the proposed synchronization scheme is different from the traditional Costas loop [106]. The proposed synchronization scheme is detecting and compensating the energy falling in the null sub-carriers to compensate the frequency and time offsets where the Costas loop is a phase-locked loop based circuit which uses

local oscillator to generate generates an output signal whose phase is related to the phase of an input "reference" signal and adjusts the frequency of its oscillator to keep the phases matched.

5.4.4 The effect of the number of bandpass filters in a node on the system throughput

As described before, the most distinguished advantages of OFDMA over other MAC layer protocols such as multi-band MAC protocol is that with OFDMA concurrent transmissions can be realized across a node. However, more concurrent transmissions that can be supported in a node means that more bandpass filters are needed, increasing the node complexity. Unlike the infrastructure based cellular networks, in ad hoc networks, the number of active corresponding transmitters for each node is limited. The number of concurrent transmissions supported by a node can be adapted depending on the number of bandpass filters equipped in each node. In this section, the system throughput of an ad hoc network where the nodes are equipped with different number of bandpass filters is investigated to see its effect on system performance. All the experiments are based on the proposed routing scheme described in chapter 4.

The uniform node distribution scenario is used in this simulation. 30 nodes are uniformly distributed in an area of 1000 by 1000 meters. The transmission range for each node is 250 meters. Each active source node randomly selects a destination node and starts a session at a random time from 0 to 10 seconds. Each session lasts until the end of the simulation. The simulation time is set to 2 minutes. The simulation parameters are shown in Table 6. The number of sub-channels is set to 8 and the

number of data sub-channel is 7. The proposed OFDMA scheme is compared with the

signal interface multiband [102] and the multi-interface multiband [103].

Table 6   Simulation configurations

Attribute Name	Value
Transmit power (dBm)	0
Number of subcarriers	64
Subcarrier frequency spacing	0.3125 (20M/64)
Receiver sensitivity (dBm)	-65
Packet size (B)	1024
Path loss model	Open space
Modulation scheme	QPSK
Error correction threshold	0
Noise figure	1
Bandwidth (MHz)	20
SIR_{min}	20dB [99]
Minimum frequency band (GHz)	2.4

In the first simulation, each node has only two bandpass filters. In this case, each node

can only support two concurrent transmissions at the same time. Figure 40 shows the

simulation results for the system throughput. If in OFDMA based networks, a node

cannot support concurrent transmissions, the system is the same as the multi-band

based networks. The simulation results show that allowing concurrent transmissions,

the system throughput can be largely increased compared to other MAC protocols.
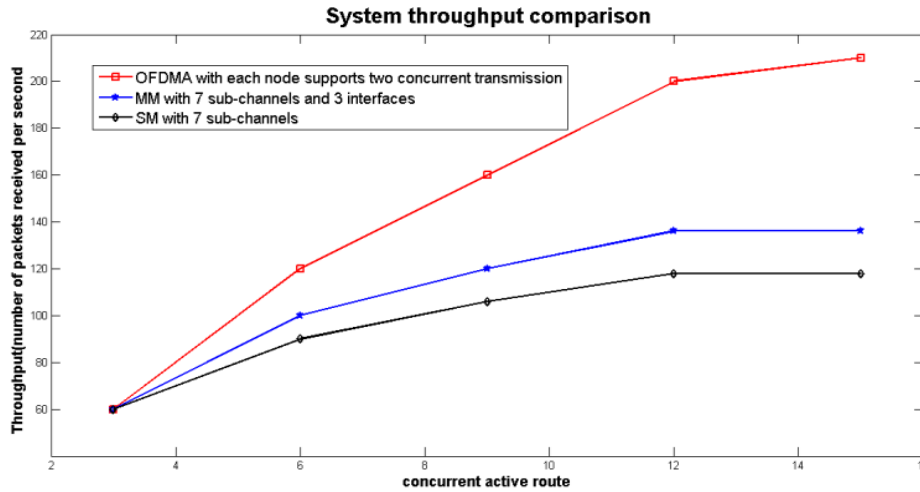
Figure 40 Comparison with two bandpass filters in each node
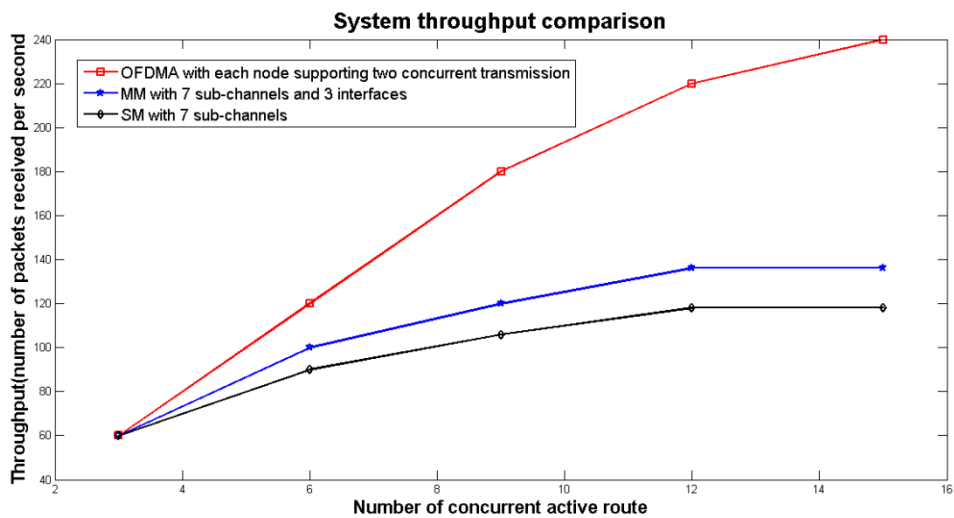
Figure 41 Comparison with three bandpass filters in each node

Figure 41 shows the simulation results for the throughput comparison between an OFDMA based system with three bandpass filters in each node and the multi-band MAC systems. Compared to Figure 40, when increasing the number of bandpass filters, the system throughput also increases accordingly. If the network is not heavily loaded (the number of concurrent active routes is smaller than 15), the system throughput with

154

three bandpass filters per node is not significantly greater than the system with two bandpass filters per node. Therefore, there are two conclusions from the above experiments: (1) By allowing concurrency, the performance of OFDMA system is significantly better than the other MAC protocols. (2) A small number of bandpass filters is sufficient to provide a substantial increase in system performance.

5.4.5 Performance evaluation of the proposed frequency offset estimation algorithm

In this section, several experiments are taken place to test the performance of the proposed frequency offset estimation algorithm. In here, a receiver is simultaneously receiving signals from two transmitters. There are 64 sub-carriers in total and the sub-carriers are divided into $S = 8$ sub-channels with $J = 8$ sub-carriers per sub-channel. Therefore, $N = S \times J = 64$. The system parameters are the length of the cyclic prefix $L = 2$, the sub-channels used for user traffic $J_a = 6$, and the null sub-carriers $J_0 = 2$. The frequency offset for these two transmitters are generated as independent random variables, distributed uniformly in (-*1/2NT, 1/2NT*) [85]. As mentioned in section 5.4.3, if the frequency offset is less than *1/2NT*, the steepest-gradient-descent algorithm converges to the global minimum point of the energy detection function. A Rayleigh multipath fading mode is used in this simulation with overall channel delay spread less than the length of cyclic prefix LT.

a) The effectiveness of the proposed frequency offset estimation algorithm

Figure 42 shows the simulation results of the frequency estimation errors of these two transmitters as a function of the iteration index for a received SNR=20dB, where SNR is defined as $\sigma_u^2 E\{|C_k(\{c_m\}, l)|^2\}/\sigma_v^2$. The step size is set to 0.05. From Figure 42, it can be seen that for both transmitters the frequency estimation errors converge approximately on the 30th step and forward. The final estimated error is slightly different for the two transmitters because the initial frequency offset for each transmitter is different from each other, consequently each frequency offset induces a different interference level for each transmitter. From the simulation results it can be seen that the proposed frequency offset estimation algorithm has good performance. The frequency estimation errors for both transmitters are less than 0.01 times the sub-carrier spacing 1/NT.
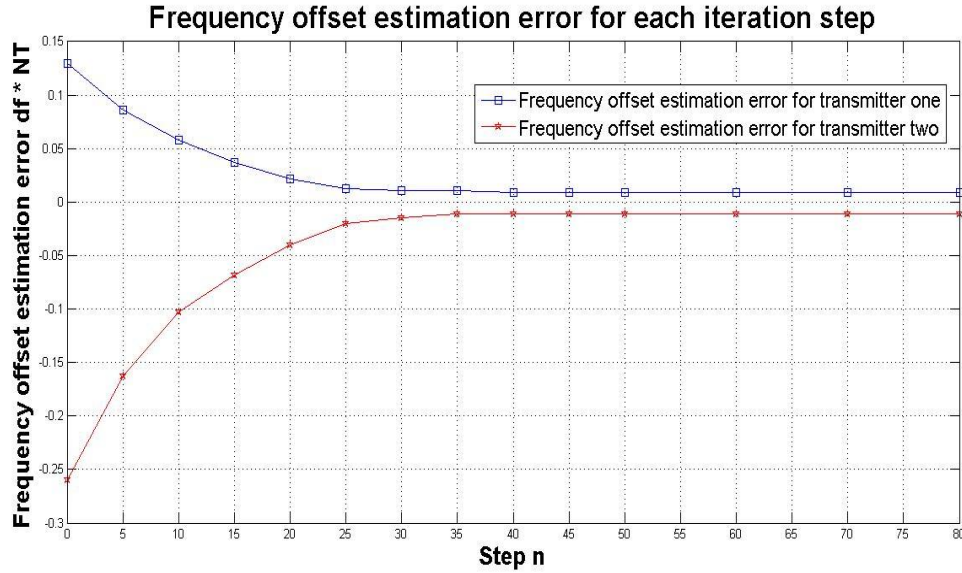
Figure 42 Frequency estimation error for step size μ = 0.05

Figure 43 shows the simulation results of the frequency offset estimation error with step size increased to 0.1. In order to compare the results with the previous experiment, the same seed is used to create the random frequency offsets for the two transmitters.

The simulation results show that with a larger step size the algorithm can converge much faster. However, the final frequency estimation errors for both transmitters increase. There is a tradeoff between the convergence speed and the tracking precision. In the next experiment, the frequency estimation error variance is evaluated for different SNRs.
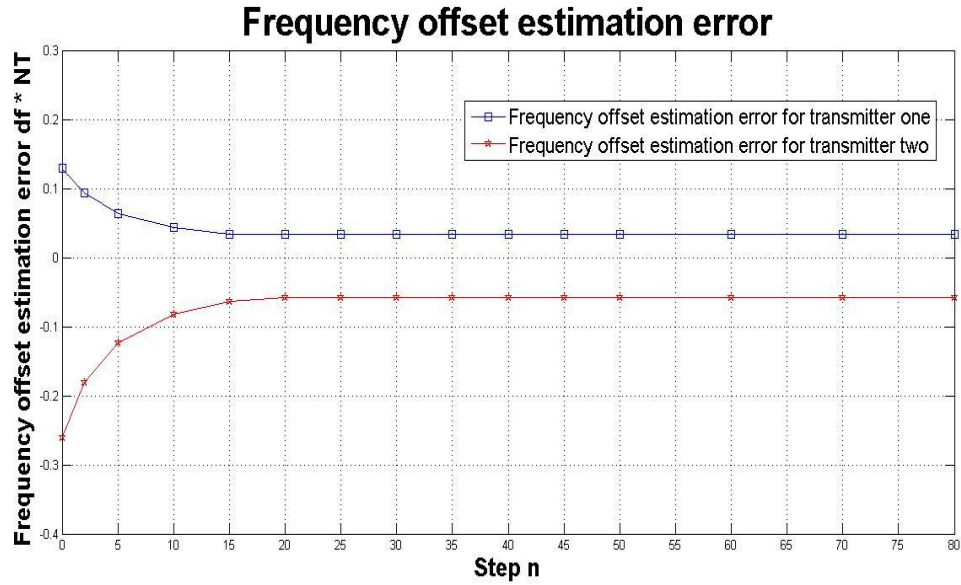
Figure 43 Frequency estimation error for step size μ = 0.1

b) The frequency offset estimation error variance for different SNRs
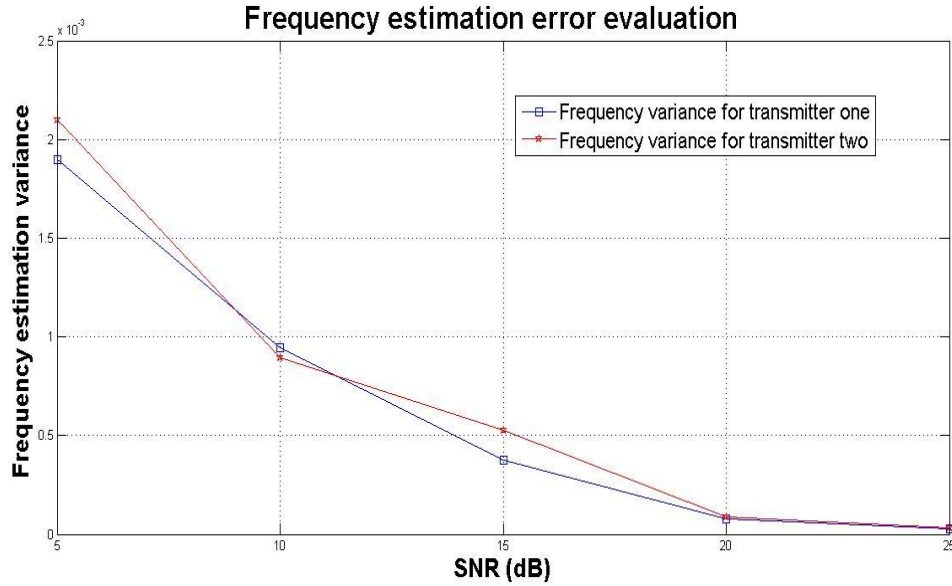
Figure 44 The estimation error variance

In order to quantify the frequency errors for different users and evaluate the SNR influence to the frequency estimation, the frequency offset estimation error variance for these two transmitters are evaluated in terms of different SNRs. The step size $\mu = 0.05$ is used. Figure 44 shows the simulation results of the frequency estimation error variance for different received SNRs. From the simulation results it can be seen that the error variance is largely decreased with the increase of the received SNR. This is because the higher SNR will reduce the effect of the interference between these two transmitters due to imperfect filtering. To evaluate the frequency synchronization scheme, in the simulation, the two transmitters are moving with different speeds and the distances between the transmitters and the receiver are different. The synchronization processes are applied individually with bandpass filters in the receiver. These result in the variance of the simulation results for these two transmitters.

However, with the increase of the received SNR the frequency estimation error variances for both transmitters are the same.

5.5 Concluding remarks

This chapter presents a novel time and frequency synchronization scheme to realise concurrent transmission in OFDMA based ad hoc networks. Section 5.2 describes some of the most popular time and frequency synchronization schemes in an OFDM system while in section 5.3, some classic time and frequency synchronization proposals in an OFDMA system are presented in details. Most of the proposed synchronization approaches in the literature are based on cellular networks which have centralized coordinators (the base station). From the extensive literature review, the only proposal for synchronization for ad hoc networks is found in [51]. However, they only consider the situation when a new user enters into the network and it needs to synchronize with the leader. Moreover, the authors do not consider the situations with multiple concurrent transmissions.

Based on this consideration, section 5.4 presents a partial time synchronization scheme as well as a null sub-carrier based frequency synchronization algorithm to support concurrency in ad hoc networks. A new system architecture is also presented in section 5.4. With the proposed partial time synchronization scheme, global time synchronization becomes unnecessary. For the frequency synchronization algorithm, null sub-carriers are inserted in the end of each sub-channel. By continuously detecting and compensating the energy falling in the null sub-carriers, the frequency offsets can be successfully estimated and compensated. The performance of the proposed

frequency offset estimation mechanism is evaluated through the simulation in the end of section 5.4.

The next chapter presents the simulation modelling for the proposed SSMAP based cross layer routing protocol and validations.

Chapter 6

SSMAP based QoS routing protocol simulation modelling and validations

6.1 Simulation modelling for the signal strength based QoS routing protocol

The proposed signal strength based medium access protocol as well as the QoS routing protocol will be evaluated though simulations using OPNET Modeller 14.5 PL1. The OPNET simulator is used as the main implementation and investigative tool which has been recently widely used as primary simulation tool for performance evaluation in various wired and wireless communication scenarios. OPNET is an event driven software tool which includes a hierarchical structure of models composed of a network model, node models and process models. Each node in the network level is composed of interconnected modules in the node level. Each module is defined as a process of a set

of processes in the process level. A process is described by a finite state machine (FSM). The C/C++ language is used as programming language in description of the functionality of each state at the code level of OPNET.

In this thesis, the proposed SSMAP is compared with other techniques such as, single radio multi-channel, multi-radio multi-channel and other OFDMA based medium access algorithm in an ad hoc environment. The implemented QoS routing protocol will be simulated both in a static ad hoc network and a mobile network. Before illustrating the simulation performance, the details of the simulation model along with the simulation validation are described in this chapter.

6.1.1 The network model
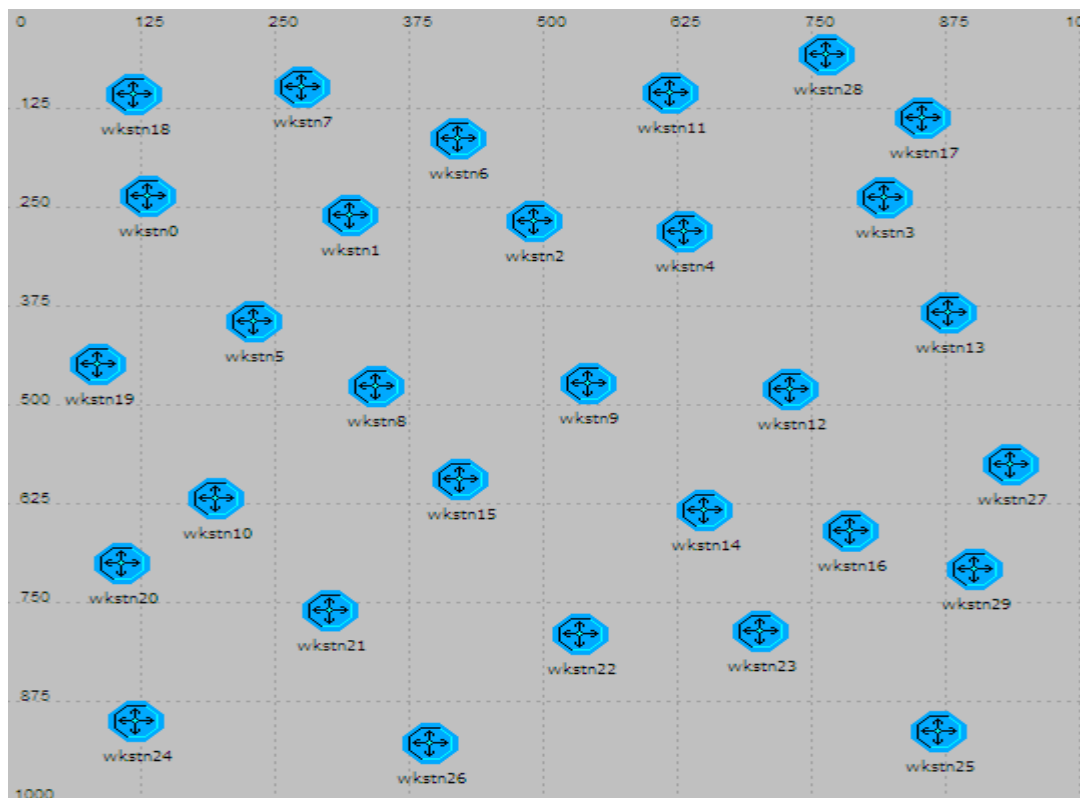
Figure 45 Network architecture and topology

A network model describes a network structure and its topology. Figure 45 shows an example of an ad hoc network with 30 nodes distributed in a $1000\,m \times 1000\,m$ area. These 30 nodes can move inside the network area and each node in the network is distinguished by a unique IP address. Every node has the same functionality and each node is built from one type of node model. The detail node model will be described in the following sub-section.

6.1.2　The node model

As shown in Figure 46, the node model is composed of four layers (from application layer to physical layer). The proposed sub-channel allocation scheme and routing protocol mainly focus on the MAC layer and the network layer. Therefore, some of the upper layers have been legitimately omitted for simplicity. There are some statistic wires starting from the transmitter and receiver to *OFDMA_based_mac* module in the MAC layer. These connections are used by the CSMA-CA process to verify the busy status of the signalling channel and monitoring the signal strength of each subcarrier.

a) **The application layer**: In the application layer, there are three modules. The *Src* module is a simple traffic generator which can generate data with constant bit rate to simulate a source for a real-time application. The generated data packets are transmitted directly to the *uplayer_manager* module which will check the data rate of the packet and install an *interface control information* (ICI) with the bandwidth requirement or/and end to end delay constraints. When the first packet is generated, it randomly assigns a destination address to the packet and sends it to the lower layer with the associated ICI. After that, the other packets are assigned

the same destination address as the first packet. When a packet arrives from the lower layer instead, the *uplayer_manager* will forward it to the *app_sink* module to destroy the packet. The *app_sink* module is also designed to collect and record the statistics from the simulation.
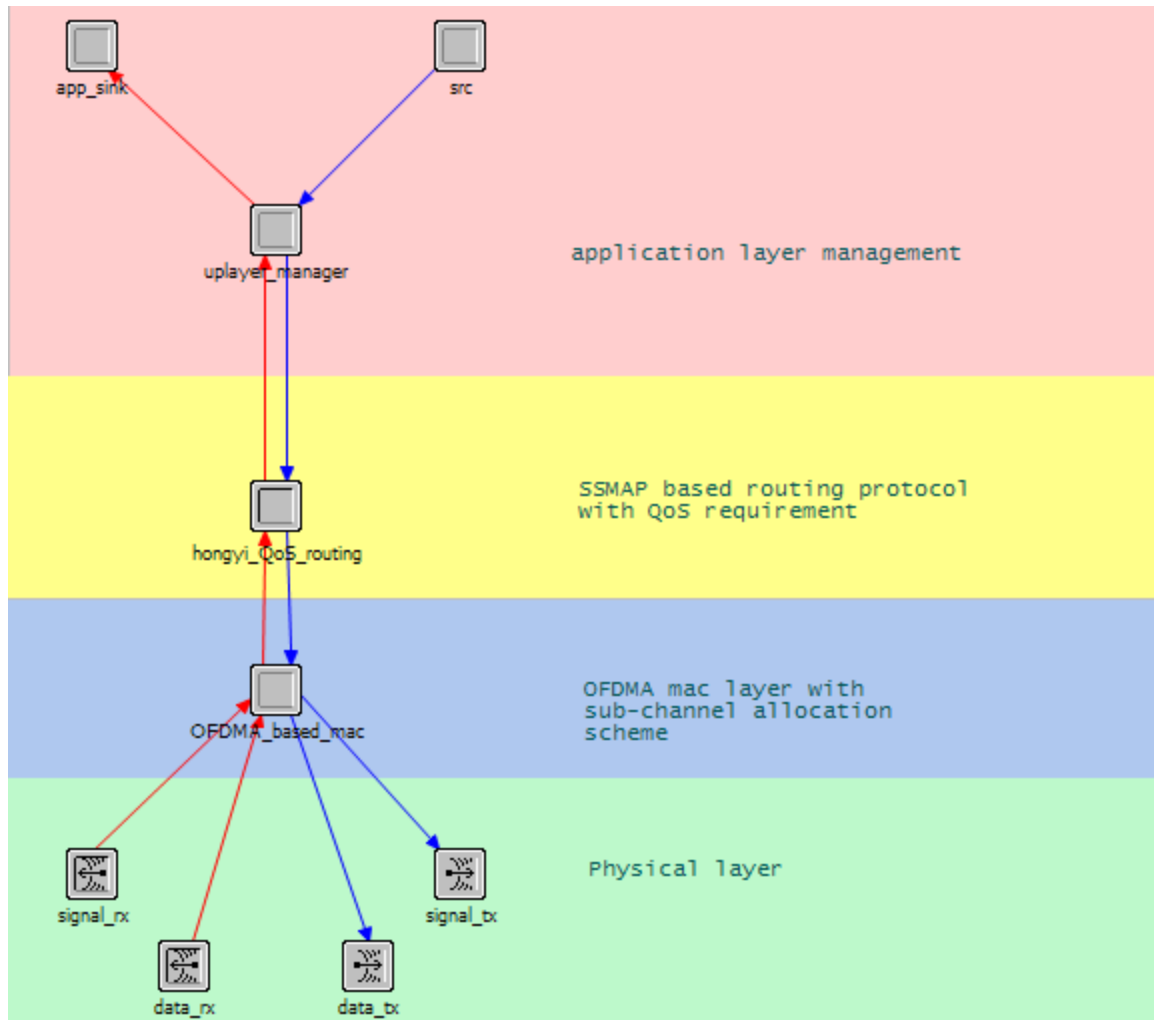
Figure 46 Node model

b) **The network layer**: The network layer only consists of one process module called *hongyi_QoS_routing,* where the proposed QoS routing protocol is implemented. There are two main tasks in this module. First it receives the packets from the upper

layer and retrieves the QoS requirement from the *ICI*. It will calculate the number of subcarriers needed for this session and send this information to the lower layer along with the packets. Secondly it also deals with the received packets from the lower layer. The entire routing algorithm is designed in this module according to the routing algorithm described in the previous chapter.

c) **The MAC layer**: The functions of the MAC layer are implemented in the *OFDMA_based_MAC* module. It is constructed based on the proposed signal-strength based medium access control. In this module, the subcarrier allocation algorithm is implemented. The subcarriers are divided into **n** *Cluster of Sub-carriers* (CoS) and each of them can be used to transmit data packets. The receiver in the physical layer will continuously monitor every sub-channel and record its instantaneous signal strength. Once a packet is received from the upper layer, the *OFDMA_based_MAC* can decide which sub-channel will be used to transmit the current data packet depending on the signal strength of each sub-channel. If there is no sub-channel available, the packet will be discarded. Accordingly, the *OFDMA_based_MAC* cooperates with *hongyi_QoS_routing* module to setup a QoS route and achieve optimal subcarrier allocation based on the information obtained from the physical layer.

d) **The physical layer**: Two sets of transceiver modules are implemented in this layer. One is for signalling transmission and the other transceiver is for data transmission. The modeling of a radio transceiver in OPNET is achieved through a mechanism known as pipeline stages. In each stage, a certain effect of the radio channel is attached to a transmitted or received packet. The cumulative effects of previous

pipeline stages determine if a packet proceeds to the next pipeline stage or is dropped. There are 14 pipeline stages and each stage is detailed in the following.

The first pipeline stage is known as *receiver group*, this stage is used to facilitate the simulation by limiting the number of nodes to receive the packets. By default, a packet generated in one node will be transmitted to all other nodes. This will largely decrease the simulation efficiency because each node will calculate 14 pipeline stages to determine if the packet is received or not. By using a *receiver group*, the node can choose the preferred nodes to transmit. However, this is achieved as the expense of decreasing the simulation result accuracy.

The next stage is known as *link closure*, this stage verifies if a transmitted packet is able to get to the destinations giving the effects of the physical terrain. The effects taken into consideration include obstacles on the path between the transmitter and receiver, interfering transmissions and so on. If any of these characteristics is found, the packet will be destroyed immediately. Otherwise it proceeds to the next stage.

Two stages referred to as the *transmission delay* and *propagation delay* are used to compute the end to end delay. Also they will be used later to calculate the packet path loss and the receiver power.

The next stage is the *channel match stage*. In this stage, the receivers check to see if the packet transmission frequency band matches their receiver frequency band. If a packet does not pass the *channel match stage* it is not dropped, but it is regarded as noise and would be used for later computations. In this model, the default setting is modified to implement OFDMA. 48 channels are used in the physical layer

to simulate 48 subcarriers. The subcarrier frequency spacing is 0.3125 MHz as specified by the IEEE 802.11 standard [10]. If the bandwidth mismatch is equal to the subcarrier frequency spacing then this signal is considered as orthogonal and will not generate interference to the system. Otherwise, the packet will be considered as interference noise.

Another two stages referred to as the *transmitter* and *receiver antenna gain* are used to compute the transmission and reception gain. This is based on the angles and directions of the transmitter and the receiver antennas, in relation to one another. This information is used along with other parameters to calculate the received power of the packet at the later stage.

The next stage is the *received power stage*. In this stage, the power of the received packet is computed based on the distance between the transmitter and receiver, the frequency band, the receiver and transmitter antenna gain, the implemented path loss model, the deployment terrain etc. If the power is below the receiver sensitivity the packet is rejected and will not proceed to the next stage.

The *Interference Noise* stage computes the amount of noise experienced by a packet during its reception. The information obtained from the *channel match* stage is one of the inputs to this stage. The computed noise information is attached to the packet, and then handed off to the next pipeline stage.

The background noise stage is similar to the *interference noise* stage. It collects other common noises which are not captured by the *interference noise* stage, such

as thermal noise of the receiver, emissions from surrounding electronics etc. The calculated information is attached to the packet and sent to the next stage.

The next stage is the *signal-to-noise ratio* (SNR) stage. In this stage the information obtained from the *receiver power stage* and those from the *interference noise* and *background noise* stages are used to compute the SNR. The computed SNR information is used as input to the next stage.

The following pipeline stage is the *bit error rate* (BER) stage. The probability of errors contained in the packet based on the SNR is computed and attached to the packet. The BER computation will be based on the type of modulation implemented by the transceivers.

The next is the *error allocation* stage that assigns the bit errors to the packet based on the information from the *BER* stage.

The last stage is the *error correction* stage. In this stage, the bit errors in a received packet are corrected based on the forward error correction (FEC) scheme chosen for the wireless receivers. If the error correction scheme is unable to correct the errors the packet is dropped, otherwise it is passed to the upper layer.

6.1.3 The process model

The functionality of each module is described by the process model. In a process model, a specific circular icon represents a state, and the lines represent transitions between states. Green circles represent a *forced* state, which means once the functions are performed in this state, the process automatically transits to the next appropriate state.

Red circles represent *unforced* states, which means once all functions are performed, the process pauses in this state and waits for a next specific event. The operations performed in each process are coded using C or C++ language. In the following, the main process models are described in detail.

1. **The *uplayer_manager***

This process module is in charge of assigning the destination address and QoS requirement for the received packets from the *src* module. It also forwards the received packets from the lower layer to the *sink* module. The Finite State machine (FSM) of the *uplayer_manager* module is shown in Figure 47.
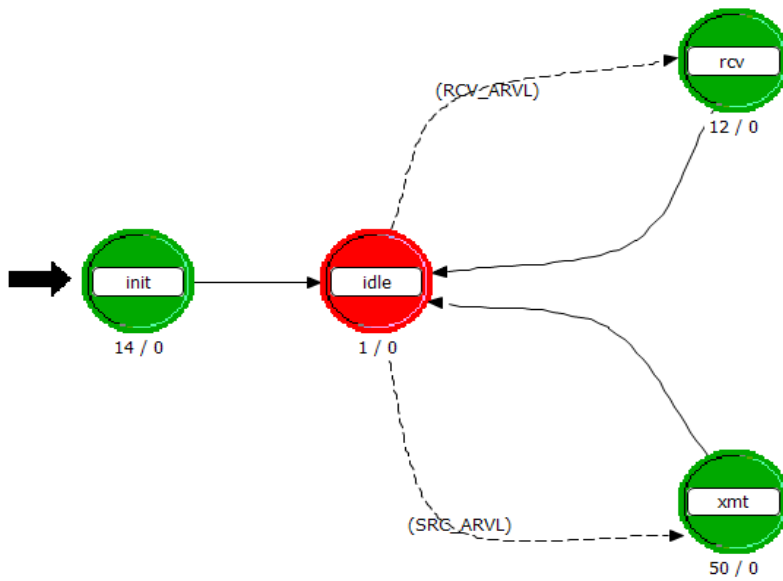
Figure 47 uplayer manager

First, each mobile node at the *init* state initialize the attributes of the node itself, such as the initial starting time of the session and the IP address of this node. Then it transits into the *idle* state waiting for a new coming packet. When a packet is received from the *resource* process, if this is the first packet received from the *resource* process, a new session is triggered and it will assign the destination node and QoS requirement for this new session. Once the destination address and QoS requirement are assigned, it transits to *xmt* state in order to start the session with assigned parameters. This process will install an *ICI* to carry the destination address and QoS requirement information of this session for the received packet and forward it to the lower layers along with the packet. Once the session is triggered, any following packet from this session received from the *resource* process will be directly forwarded to the lower layer with destination address and QoS requirement. On the other hand, if a packet is received from the lower layer. It will just simply forward it to the *sink* module for recording the simulation statistics.

2. *hongyi_QoS_routing*

The *hongyi_QoS_routing* process module is the most important process which includes all the functionality for the proposed QoS routing protocol. There are nine forced states and an *idle* state. Each state is described by a set of actions and triggered by an event. Figure 48 shows the OPNET model for the *hongyi_QoS_routing* process.
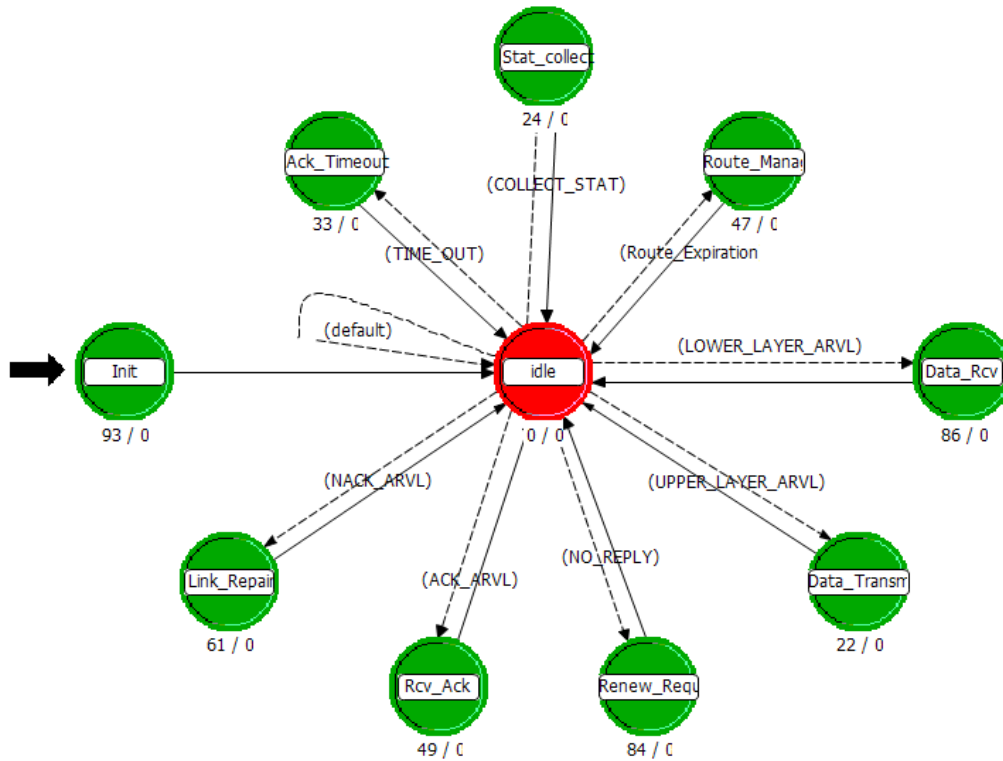
Figure 48 The hongyi_QoS_routing process

- *Init state*: In this state, the user attributes and different variables are initialized, such as the routing table, *Free_Channel* table, transmission and reception sub-channels, statistic tables, etc. After all the initializations, the process goes to *Idle* state to wait for the next trigger event.

- *Data_Transmission*: When a data packet is received from the application layer, the process transmits to *Data_Transmission* state. In this state, the process will extract the *ICI* associate with the received packet to get the QoS information as well as the destination address. Then it will check its local routing table and the *Free_Channel* table to decide if it needs to trigger a route discovery process or not. All the routing functions are implemented according to the previous chapter.

- *Data_Rcv*: On the other hand, if a data is received from the MAC layer, the process

transits to this state. First, the type of the packet is checked. It can be either a signalling message (*RDIS, RRES*, etc.) or a data packet. A corresponding function is called according to the packet type that sets up, maintains and releases routes or updates the routing and *Free_Channel* table. If it is a data packet, the process just forwards it to the next hope or transmits to the upper layer if it is the final destination.

● *Renew_Request*: If a route discovery message (*RDIS)* is sent out to look for a new route, the process will keep a timer to wait for the *RRES* message. If the timer is expired and this node still did not receive the route reply message, the process will check whether it will re-broadcast a new *RDIS* message or not, depending on the number of retries threshold. If the maximum number of retries is exceeded, the current state will cancel the route discovery process for that destination and discard all the related queued packets. Otherwise, a new route discovery message for that destination is generated.

● *Link_Repair*: The process transits to this state when a *NACK* message is received from the lower layer which means that either there is a forthcoming link breakage on one of the active route or an interference node is approaching. An appropriate function will be triggered either to start a local repair or reallocate the transmission sub-channel for the previous hop link.

● *Route_Manage*: When an active route is expired, this state is triggered. The route entry for that destination will be deleted from the routing table and all the reserved network resources will be released. The *Free_Channel* table will be updated accordingly.

● *Stat_Collect*: After the simulation is finished, the process will transit to this state to

collect and handle the simulation results. The statistics are written into a file which is created at the beginning of the simulation.

- *Rcv_ACK* and *ACK_Timeout*: These two states are only triggered when TCP transmission is used. There is a *TCP_flag* attribute which is needed to be initialized when the simulation starts to indicate the data transmission is based on TCP or best-effort.

3. *OFDMA_based_MAC*

This process is in charge of sub-channel allocation, and sub-channel monitoring. Figure 49 shows the OPNET *OFDMA_based_MAC* process model. The *Init* state initializes the process and the *idle* state waits for events. The other four states are described as follow:

- *Xmt*: When a packet is received from the upper layer, the process transits to this state. It just forwards the packet to the physical layer to the next hop.

- *Rcv:* When a packet is received from the physical layer, the process transits to the *Rcv* state. It will first extract the sub-channel occupation information, destination IP address and packet type, and received signal strength from the received packet. After that, the *Free_Channel* table is updated according to the obtained information. Finally, the packet will be transferred to the upper layer for further processing.

- *Interf_Dealing:* Once the *Free_Channel* table is updated and the interference signal strength on an active route exceeds a pre-defined threshold, the process will transit to this state. A *hongyi_subchannel_reallocation* function will be called to look for alternative available sub-channels in the *Free_Channel* table for the affected route and send a *SREA* message to the previous hop transmission node.

173
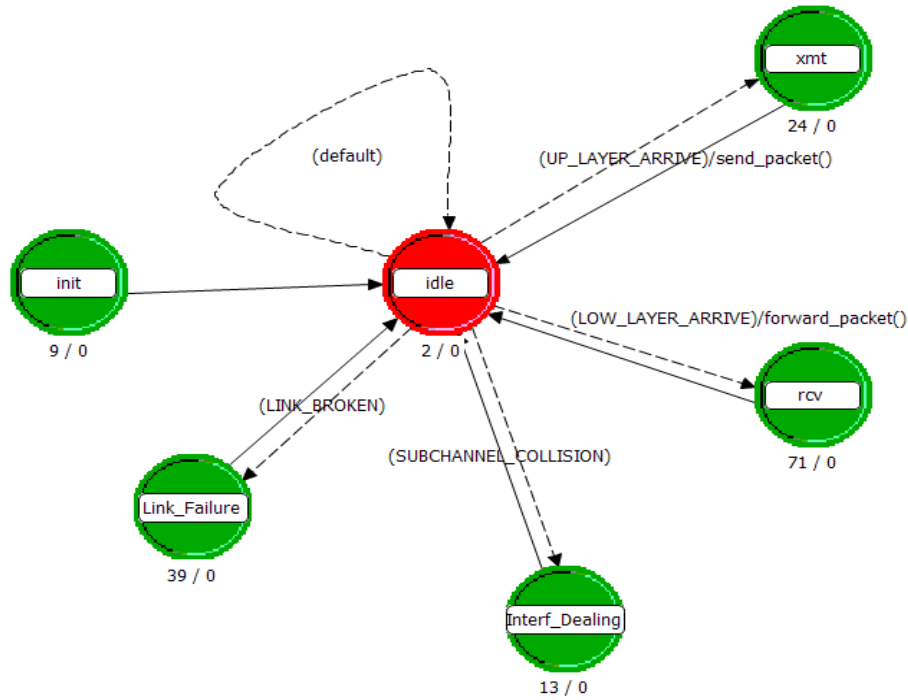
Figure 49 The OFDMA_based_MAC process

- *Link_Failure:* If the signal strength of the received packet from the previous hop node is lower than a pre-defined threshold for a period of time, the process will enter into this state. This means that a forthcoming link breakage is predicted and the local link repair function is called to fix the problem.

6.2 Validation of the SSMAP based QoS routing protocol modelling and explanation of partial time synchronization

The simulation model described in this chapter aims to evaluate the performance of the proposed signal-strength based sub-channel allocation scheme and the proposed routing protocol. Therefore, a validation is required to prove the proposed protocol is correctly implemented in the simulation model and a verification is needed to prove the simulation results are correct. Model validation is usually defined as the "substantiation

that a computerised model within its domain of applicability possesses a satisfactory range of accuracy consistent with the intended application of the model" and model verification is often defined as "ensuring that the computer program of the computerised model and its implementation are correct" [107].

The validation will prove the correctness of three major implementations. Firstly, the validation will prove that the routes can be successfully established through the route discovery process. Secondly, it will prove that the route maintenance mechanism works as required. Finally, it will prove that the proposed sub-channel allocation scheme is implemented correctly during the route discovery process. The verification will be investigated in chapter 7 through simulations. In order to validate the SSMAP based routing protocol, a series of simulations were carried out through different scenarios.

Event traces are used in [104] and [40] to verify routing protocols implemented using NS (network simulator) and OPNET respectively. In this thesis, a similar technique is used to validate the SSMAP based routing protocol using OPNET. Trace files are created by the C code in OPNET to trace the signalling messages and record the living states of the different layers in each node.

6.2.1 Validation of route discovery process and sub-channel allocation mechanism

The validation of the route discovery process should consider three issues. Firstly, for each routing request, a path can be successfully set up for a required destination. Secondly, the sub-channel can be allocated properly for each link with the *RDIS* message reaching the source node through the reverse path and the partial time synchronization

mechanism is running successfully. Finally, once the route has been successfully set up, the route maintenance scheme is running successfully to deal with the link breakage and co-channel interference when nodes are moving.

To successfully receive an OFDM packet, the first task for each receiver is to determine the correct symbol starting position so that the FFT window can align to the right part of the symbol. As described previously, this task can only be achieved by using additional information. In this simulation, a training symbol with two identical halves is used. To support the routing discovery and maintenance mechanism, one separate common sub-channel is needed to perform the signalling message and time reference exchanges. This sub-channel is distinct and orthogonal to other sub-channels that are used to transmit data information. In the ad hoc network, the time acquisition is done by exploiting the repetition structure of the training blocks. Each mobile node needs to continuously monitor the common signalling sub-channel in order to get the update session request information. By monitoring the signalling sub-channel the mobile node is able to find the training block and perform the synchronization procedure. Two sets of transceivers are equipped in each mobile terminal: one for the data traffic and the other for the signalling messages. Each mobile node will continuously monitor the signalling sub-channel by finding the beginning of a new frame and the right position for FFT windows. Any received time-domain signal will be sampled and the samples are fed into a sliding window correlator of lag N/2 (where N is the length of the training symbol). The correlation of two continuous N/2 samples is calculated. Once the correlation exhibits a peak, it represents the sliding window is perfectly aligned with the received training block that shows the right FFT window starting point. The data structure for both

signalling channel and data channel is frame based where each frame consists of one training symbol followed by 20 data symbols. The first training symbol is specifically designed and used to detect the beginning of each frame to coarse timing synchronization.
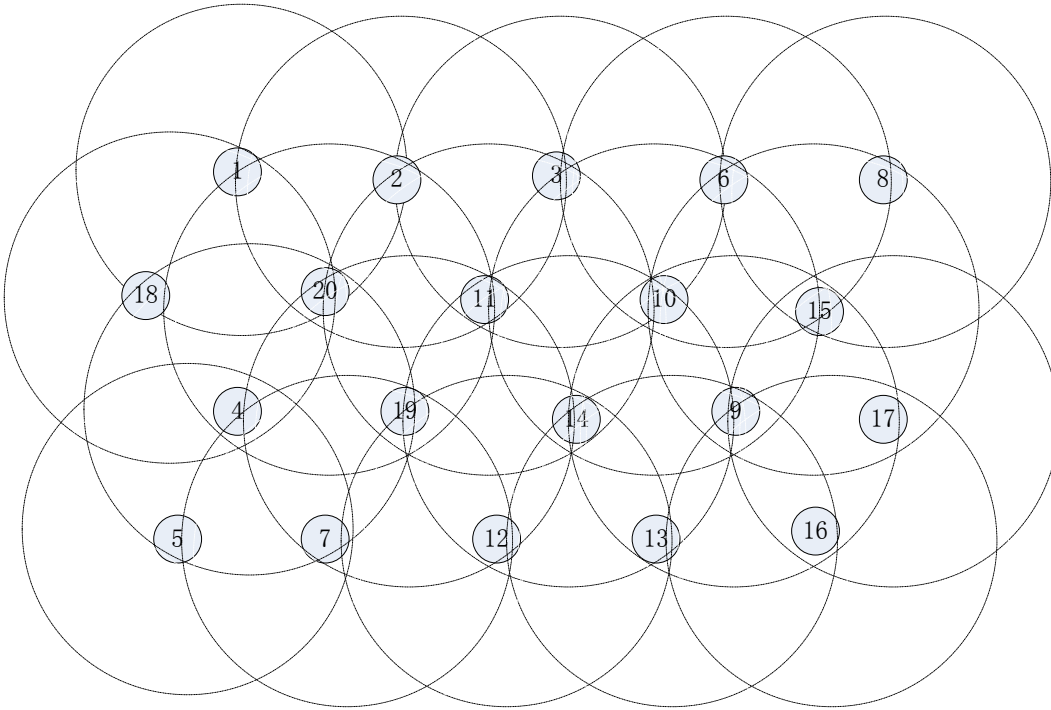
Figure 50 The overage of each node

Therefore, an independent channel is needed to be in charge of the signalling messages. In this thesis, an additional transceiver is proposed to carry out the signalling through the specific sub-channel as described in section 6.1.2. Two sets of transceivers are required in each mobile node: one for the data traffic, and the other for the signalling sub-channel. The first transceiver is only in charge of the signalling messages. This signalling channel is also OFDM based and uses a different sub-channel from the other data sub-channels. Since the two transceivers are using different sub-channels, the

signalling and data packets can be transmitted simultaneously without interfering with each other.

A simulation scenario is setup for the validation. An ad hoc network with 20 nodes is generated in an area of 1000m×1000m as shown in Figure 50. The dash circle is the transmission range for each node. The network model in OPNET is shown in Figure 51. Each mobile node in the network can only communicate with the other nodes within one hop distance. For example, node 1 can only directly communicate with nodes 2, 20 and 18. The total bandwidth is 20 MHz and the total spectrum is divided into 8 sub-channels from 0 to 7. Each sub-channel has bandwidth of 2 MHz. Sub-channel 0 is for signalling messages and the other 7 sub-channels are used to transmit data packets. In the simulation, 5 sessions are requested by the nodes 1, 2, 3, 6, 8 to the destination nodes 12, 4, 16, 8, 17 respectively. The simulation is set to 2 minutes and the route requests are triggered in nodes at a sequence of node 1, 2, 3, 6, 8. Once a route is established, each session will last until the end of the simulation. The five selected routes are shown in Figure 51 with five different colours.

A file is created to trace the partial time synchronization and the sub-channel allocation with the travel of the *RDIS* and *RRSP* packets. The file firstly records the route request from a source node to a destination node with the partial time synchronization. Once the destination node receives the *RDIS* packet, it sends a *RRSP* packet back along the reverse path. Each node receives the *RRSP* packet will write the file with the sub-channel allocation information until the *RRSP* packet reaches the source node.
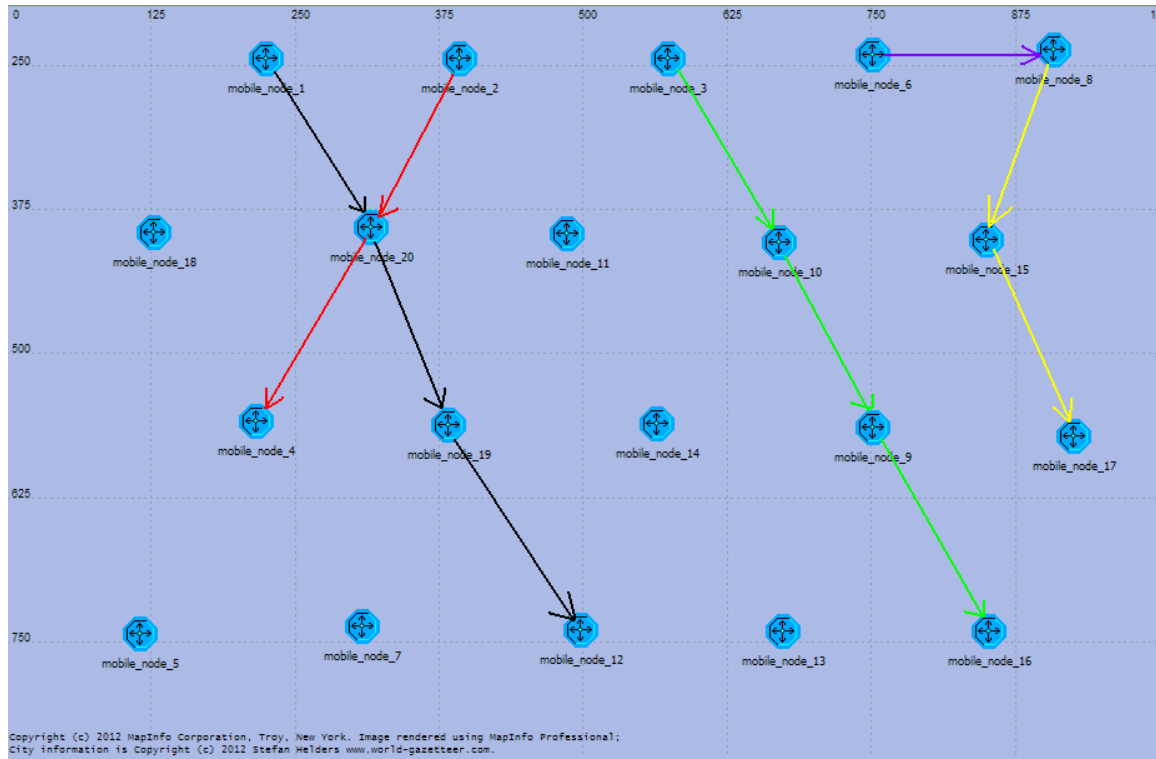
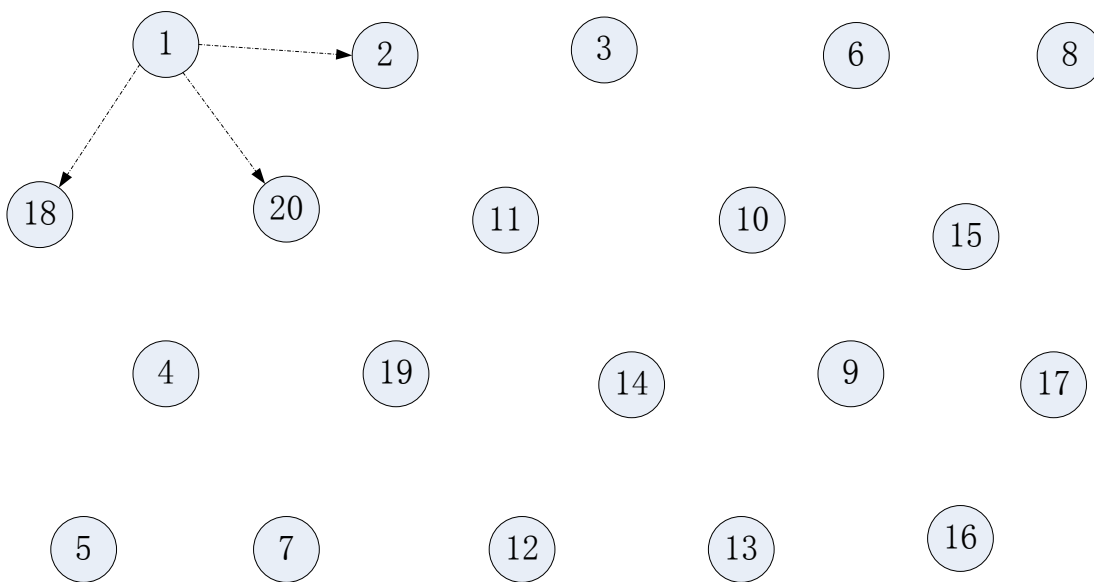Figure 51 Network topology with five selected routes

Figure 52 Route discovery process for node 1

At the very beginning of the simulation, all the nodes in the network are silent. At one time, node 1 needs to transmit packets to node 12. It firstly sees if the signalling channel is free by monitoring the signalling channel using the separate transceiver. Once the signalling channel is free, the node starts to broadcast a *RDIS* message to its neighbouring nodes through the signalling transceiver using the specific signalling sub-channel 0 (as shown in Figure 52) with its local time reference. The current active sub-channel information which is used for transmission in node 1 is still encapsulated into the *RDIS* packet. The signalling message sent by node 1 is framed based with the training preamble followed by 20 data blocks. Since every node in the network is continuously monitoring the signalling channel. Upon receiving the signal from the signalling sub-channel, nodes 2, 18 and 20 will sample the received signal and look for the preamble by calculating the correlation until they find the peak. After they get the right position of the FFT window, they will perform the frequency synchronization according to the algorithm described in chapter 5. In this work, it is assumed that the cyclic prefix is longer than the channel delay spread. Because there is only one shared signalling channel, for this signalling channel each node will perform the CSMA/CA mechanism to avoid collisions. This means that each node will broadcast the routing discovery message only when the signalling channel is idle. When a node wants to transmit, if the signalling channel 0 is being used by other nodes inside the waiting node's coverage area, the waiting node will backoff and try again after the backoff timer has expired.

After successful time and frequency synchronization, the route discovery message is successfully received in nodes 2, 20, 18. After that, these nodes will decide if they

should generate a new route discovery message or not depending on the local available resources. If they can fulfil the request, they will record the time reference of node 1 from the *RDIS* packet and generate a new route discovery message in the frame format with a predesigned preamble in front of data blocks using the signalling sub-channel and broadcast it to its neighbours. Nodes 2, 18, 20 will compete to send the route discover message using CSMA/CA mechanism with the signalling channel.

The route discovery messages will continuously been broadcasted until it reaches the destination node 12. After node 12 receives the *RDIS* packet, it will check its local *Free_Channel* table to see if it has available sub-channel to carry this request. If it does, it will assign the transmission sub-channel for node 19 and encapsulate this information into a *RRSP* packet. The *RRSP* packet will be sent back to node 1 through the reverse path. Each node along the reverse path receives the *RRSP* packet and writes down the assigned transmission sub-channel and forward this packet to the next reverse hop until it reaches node 1. In order to validate the route discovery process, a trace file is created. The content of the trace file for this session along the path is as follows:

————————************************************————————

Session from node 1 to node 12:

`route request is initiated by node 1:`

`***** The node address is 1 *****`

`****** The start time is 0.172461 *****`

`****** The dest node is 12 *****`

`***** The rdis packet is created *****`

***** The rdis packet is created *****

RDIS reaches the node 20:

***** node 20 ***** time: 0.176814

*****Preamble detected. Packet received successfully*****

***** The rdis comes from 1 *****

****** There are 7 sub-channels available in Free_channel
table******

***** subchannel 1 is reserved for receiving*****

***** The rdis packet is created *****

RDIS reaches the node 19:

***** node 19 ***** time: 0.180963

*****Preamble detected. Packet received successfully*****

***** The rdis comes from 20 *****

****** There are 6 sub-channels available in Free_channel
table******

***** subchannel 2 is reserved for receiving*****

***** The rdis packet is created *****

RDIS reaches the destination node 12:

***** node 12 ***** time: 0.184793

*****Preamble detected. Packet received successfully*****

***** The rdis comes from 19 *****

***** There are 5 sub-channels available in Free_channel

table*****

***** subchannel 4 is reserved for receiving*****

***** The rres packet is created *****

subchannels are allocated in node 19 when receiving a rres:

***** node 20 --> node (19) --> node 12 ***** time: 0.188781

*****Preamble detected. Packet received successfully*****

***** The rres comes from 12 *****

***** Transmission subchannel is 4 *****

***** Receiving subchannel is 2 *****

subchannels are allocated in node 20 when receiving a rres:

***** node 1 --> node (20) --> node 19 ***** time: 0.193214

*****Preamble detected. Packet received successfully*****

***** The rres comes from 19 *****

***** Transmission subchannel is 2 *****

***** Receiving subchannel is 1 *****

subchannels are allocated when rres reaches the source node 1:

***** node (1) --> node 20 ***** time: 0.196871

*****Preamble detected. Packet received successfully*****

***** The rres comes from 20 *****

***** Transmission subchannel is 1 *****

The route has been built for node 1

— — — — — — — — ∗ — — — — — — —

The trace file shows the simulation results match the expected behaviour of the network. The route for the session from node 1 to node 12 has been successfully established with proper sub-channel allocation along the path.

Once the first route (node 1 to node 12) is established, node 1 starts to forward the data packets through the allocated channel. Similar process is taken place for the session from node 2 to node 4. It will first check if the signalling channel is busy. If not, it will generate a route discovery message and put it into the frame format with a predesigned training symbol in front of each frame. After that, it will broadcast the route discovery message to nodes 1, 20, 11 and 3 through the signalling channel (as shown in Figure 53). Since each node in the network will continuously monitor the signalling channel, nodes 1, 20, 11 and 3 can successfully receive the broadcast message by looking for the training symbol. Upon receiving the broadcasted signal the receiving node will perform the time and frequency synchronization accordingly to the signalling channel the same as the synchronization procedure described in the route establish process of the first session.

Although there is one session going on between nodes 1 and 12, the channel used to transmit data packets is different from the signalling channel. Therefore, there is no interference between them which means the routing discovery process and the data transmission can be carried on simultaneously. A similar trace file is built for the route discovery process of the route from node 2 to node 4. Once the route is established node 2 will start forwarding the queued packets to node 4 through the data channel as

shown in Figure 54. To successfully receive the data frame from the one hop transmitter, as described previously, the transmitter will inform the corresponding receiver roughly what time it will transmit the data frame according to the transmitter's local time reference through the route discovery signalling messages. Therefore, each hop receiver will know when it should prepare to calculate the correlation of the receiver signal to successfully get the forthcoming data frame.
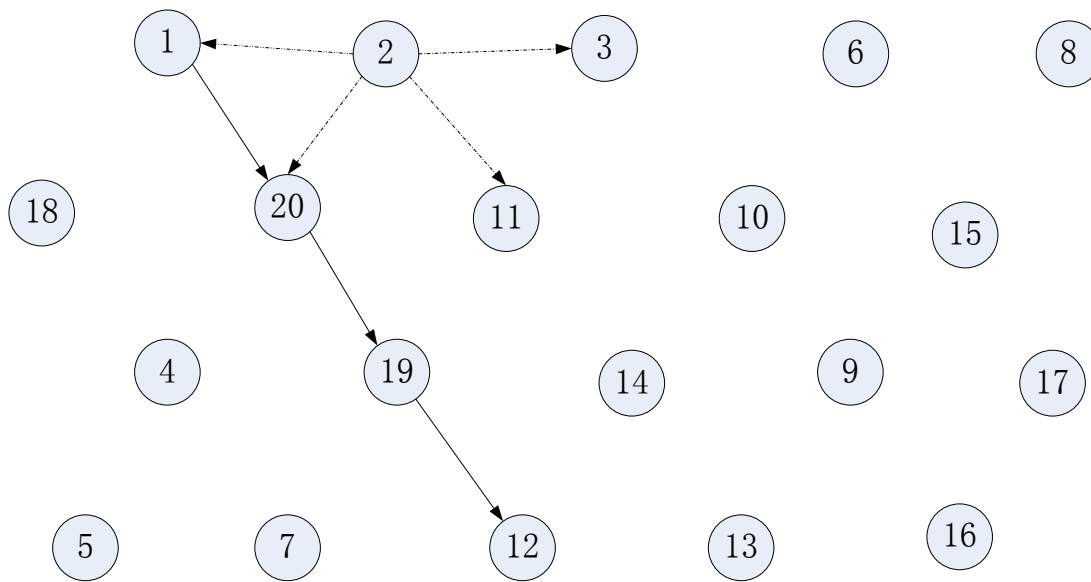

Figure 53 A new session starts
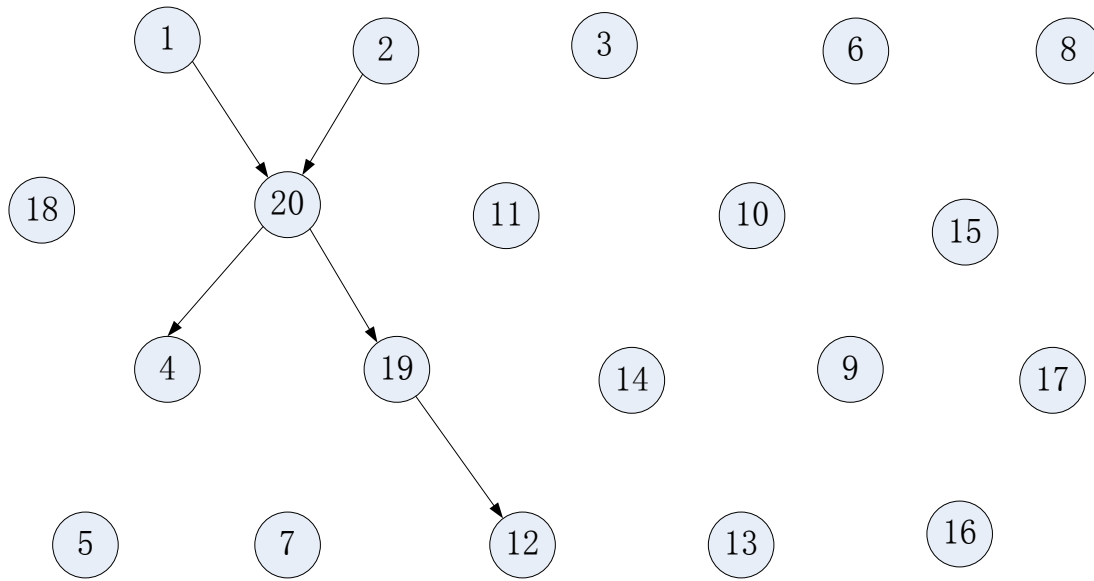
Figure 54 Second route is established

One thing should be noted that, node 20 is a cross node which participates in both sessions. Since the signals from nodes 1 and node 2 will have different frequency offsets, the received data signals from different sources should be separated in node 20 before performing the time and frequency synchronization procedure. According to the proposed routing protocol, the sub-channel used by node 1 and node 2 will be different and orthogonal to each other to avoid co-channel interference. Therefore, two bandpass filters are used in node 20 to separate the signals so that the time and frequency synchronization can be done individually for each of the transmitters. One of the advantages of the proposed synchronization scheme is that, the transmitted signals from nodes 1 and 2 do not need to arrive at node 20 at the same time which can largely relieve the time synchronization task. Similarly, the routes for the sessions from node 3 to node 16, node 6 to node 8 and node 8 to node 17 will be established sequentially

through the route discovery process. The trace file for these four routes establishments is as follows:

_____***_____

Session from node 2 to node 4:

route request is initiated by node 2:

***** The node address is 2 *****

****** The start time is 2.471658 *****

****** The dest node is 4 *****

***** The rdis packet is created *****

RDIS reaches the node 20:

***** node 20 ***** time: 2.474292

*****Preamble detected. Packet received successfully*****

***** The rdis comes from 2 *****

***** There are 4 sub-channels available in Free_channel

table*****

***** subchannel 3 is reserved for receiving*****

***** The rdis packet is created *****

RDIS reaches the destination node 4:

***** node 4 ***** time: 2.476941

*****Preamble detected. Packet received successfully*****

***** The rdis comes from 20 *****

***** There are 3 sub-channels available in Free_channel*

*table*****

***** subchannel 5 is reserved for receiving*****

***** The rres packet is created *****

subchannels are allocated in node 20 when receiving a rres:

***** node 2 --> node (20) --> node 4 ***** time: 2.480721*

*****Preamble detected. Packet received successfully*****

***** The rres comes from 4 *****

***** Transmission subchannel is 5 *****

***** Receiving subchannel is 3 *****

subchannels are allocated when rres reaches the source node 2:

***** node (2) --> node 20 ***** time: 2.483122*

*****Preamble detected. Packet received successfully*****

***** The rres comes from 20 *****

***** Transmission subchannel is 3 *****

The route has been built for node 2

Session from node 3 to node 16:

route request is initiated by node 3:

***** The node address is 3 *****

****** The start time is 4.114623 *****

***** The dest node is 16 *****

***** The rdis packet is created *****

RDIS reaches the node 10:

***** node 10 ***** time: 4.116821

*****Preamble detected. Packet received successfully*****

***** The rdis comes from 3 *****

***** There are 3 sub-channels available in Free_channel

table*****

***** subchannel 6 is reserved for receiving*****

***** The rdis packet is created *****

RDIS reaches the node 9:

***** node 9 ***** time: 4.118913

*****Preamble detected. Packet received successfully*****

***** The rdis comes from 10 *****

***** There are 5 sub-channels available in Free_channel

table*****

***** subchannel 7 is reserved for receiving*****

***** The rdis packet is created *****

RDIS reaches the destination node 16:

***** node 16 ***** time: 4.122103

*****Preamble detected. Packet received successfully*****

***** The rdis comes from 9 *****

***** There are 5 sub-channels available in Free_channel*

*table*****

***** subchannel 1 is reserved for receiving*****

***** The rres packet is created *****

subchannels are allocated in node 9 when receiving a rres:

***** node 10 --> node (9) --> node 16 ***** time: 4.124982*

*****Preamble detected. Packet received successfully*****

***** The rres comes from 16 *****

***** Transmission subchannel is 1 *****

***** Receiving subchannel is 7 *****

subchannels are allocated in node 20 when receiving a rres:

***** node 3 --> node (10) --> node 9 ***** time: 4.127611*

*****Preamble detected. Packet received successfully*****

***** The rres comes from 9 *****

***** Transmission subchannel is 7 *****

***** Receiving subchannel is 6 *****

subchannels are allocated when rres reaches the source node 3:

***** node (3) --> node 10 ***** time: 4.129976*

*****Preamble detected. Packet received successfully*****

***** The rres comes from 10 *****

***** Transmission subchannel is 6 *****

The route has been built for node 3

Session from node 6 to node 8:

route request is initiated by node 6:

***** The node address is 6 *****

****** The start time is 6.614298 *****

****** The dest node is 8 *****

***** The rdis packet is created *****

RDIS reaches the destination node 8:

***** node 8 ***** time: 6.617563

*****Preamble detected. Packet received successfully*****

***** The rdis comes from 6 *****

****** There are 4 sub-channels available in Free_channel
table******

***** subchannel 5 is reserved for receiving*****

***** The rres packet is created *****

subchannels are allocated when rres reaches the source node 6:

***** node (6) --> node 8 ***** time: 6.620124

*****Preamble detected. Packet received successfully*****

***** The rres comes from 8 *****

***** Transmission subchannel is 5 *****

The route has been built for node 6

Session from node 8 to node 17:

route request is initiated by node 8:

***** The node address is 8 *****

****** The start time is 8.342153 *****

****** The dest node is 17 *****

***** The rdis packet is created *****

RDIS reaches the node 15:

***** node 15 ***** time: 8.344324

*****Preamble detected. Packet received successfully*****

***** The rdis comes from 8 *****

***** There are 3 sub-channels available in Free_channel table*****

***** subchannel 3 is reserved for receiving*****

***** The rdis packet is created *****

RDIS reaches the destination node 17:

***** node 17 ***** time: 8.347524

*****Preamble detected. Packet received successfully*****

***** The rdis comes from 15 *****

***** There are 3 sub-channels available in Free_channel table*****

***** subchannel 2 is reserved for receiving*****

***** The rres packet is created *****

subchannels are allocated in node 15 when receiving a rres:

***** node 8 --> node (15) --> node 17 ***** time: 8.349924

*****Preamble detected. Packet received successfully*****

***** The rres comes from 17 *****

***** Transmission subchannel is 2 *****

***** Receiving subchannel is 3 *****

subchannels are allocated when rres reaches the source node 8:

***** node (8) --> node 15 ***** time: 8.353124

*****Preamble detected. Packet received successfully*****

***** The rres comes from 15 *****

***** Transmission subchannel is 3 *****

The route has been built for node 8

--------**--------

From the trace file it can be seen that the simulation results match the expected behaviour described above. The five routes are established successfully through the route discovery process and the sub-channels are allocated for each session correctly. The allocated sub-channels avoid collisions within the transmission range of neighbouring nodes. At a cross node, different sub-channels are used to support concurrent transmissions. In a node, the transmission and reception can be performed at the same time by using different sub-channels.

6.2.2 Validation of route maintenance scheme

The implementation of the route maintenance scheme (described in section 4.4.4) should be validated. The mobility of nodes can cause link breakage or sub-channel collision. Both cases need to be simulated and validated. In order to validate the route maintenance scheme, another simulation is taking place. At the beginning of the simulation, 5 routes are established and the source nodes are starting to forward the packets as shown in Figure 51. At 1 minute simulation time, all the nodes in the network move at speeds from 2m/s to 10m/s. The links break because of the movement of nodes. The route breakage and the rebuilding of a route are traced in a file. Sub-channel collisions and reallocations are also traced and recorded in another file. As shown in Figure 56, Node 2 is the source node of the route to node 4 (2 -> 20 -> 4) at the beginning of the simulation. However, link $l(2, 20)$ breaks because of the mobility of the nodes. That is traced in a file as follows:

```
_____*************************_____

The node 20 detects a forthcoming breakage from 2 at 7.985471

**** The node 20 detects a link breakage from node 2 at 7.985471
on session 2 ****

**** No alternative node found. Link repair failed ****

**** The RERR packet is created ****

**** The reserved subchannels for session 2 will be released at
8.485471 ****
```

RERR is received by node 2 *at 7.990471*

*****Preamble detected. Packet received successfully*****

***** The RERR comes from 20 *****

**** The node 2 is the source node. Rebuild a route for the

session ****

**** The RDIS packet is created ****

RERR is received by node 4 *at 7.990471*

*****Preamble detected. Packet received successfully*****

***** The RERR comes from 20 *****

**** The node 4 is the destination node ****

**** The reserved subchannels for session 2 will be released at

8.490471 ****

——————**************************——————

The forthcoming route breakage is reported to both sides of the route. The node receiving the *RERR* message releases all the resource reservation for the session and forwards the *RERR* message to the source or destination nodes. Node 2 is the source node. It releases the resources as well, and rebroadcasts a *RDIS* message to search for a new path to the destination node 4. The new path (2 -> 19 -> 4) is found as follow:

RDIS reaches the node 19:

***** node 19 ***** time: 8.004324

*****Preamble detected. Packet received successfully*****

***** The rdis comes from 2 *****

****** There are 2 sub-channels available in Free_channel
table******

***** subchannel 7 is reserved for receiving****

***** The rdis packet is created *****

RDIS reaches the destination node 4:

***** node 4 ***** time: 8.007324

*****Preamble detected. Packet received successfully*****

***** The rdis comes from 19 *****

****** There are 2 sub-channels available in Free_channel
table******

***** subchannel 6 is reserved for receiving*****

***** The rres packet is created *****

subchannels are allocated in node 19 when receiving a rres:

***** node 4 --> node (19) --> node 2 ***** time: 8.010324

*****Preamble detected. Packet received successfully*****

***** The rres comes from 4 *****

***** Transmission subchannel is 6 *****

***** Receiving subchannel is 7 *****

subchannels are allocated when rres reaches the source node 2:

***** node (2) --> node 19 ***** time: 8.013324

*****Preamble detected. Packet received successfully*****

***** The rres comes from 19 *****

***** Transmission subchannel is 7 *****

The route has been built for node 2

————————***************************************————————

The trace file shows that the route breakage detection and route re-establishment scheme have been implemented successfully.

Another problem comes from the co-channel interference. Node 15 forwards packets to node 17 using sub-channel 2. Node 17 detects that a node (node 20) using the same transmission sub-channel is moving toward to it. That is traced in a file as follows:

——————*************************——————

The node 17 detects a co-channel collision at 12.827653

**** The node 17 detects interference on receiving subchannel 2 at 12.827653 ****

**** There are 2 subchannel available in Free_channel table, subchannel reallocation successful ****

**** Sub-channel 4 is selected for the new receiving subchannel

**** The SREA packet is created and sent to node 15****

——————*************************——————

The *SREA* packet is created in node 17 and sent back to the previous hop node 15 for

the sub-channel re-allocation. After the calculation, the node 15 confirms the sub-channel re-allocation and starts to forward the packets to node 17 through the new allocated sub-channel which is traced in the file as follows:

```
_____**************************_____

The node 15 receives the SREA from node 17 at 12.830653

*****Preamble detected. Packet received successfully*****
***** The SREA comes from 17 *****

**** subchannel reallocation accept, transmission subchannel to
node 17 is changing to 4 ****

**** The SRRP packet is created ****

_____**************************_____
```

After node 15 sends the *SRRP* packet back to node 17 to confirm the sub-channel reallocation, it starts to forward the data packets to node 17 through the new allocated sub-channel 4. The co-channel interference is repaired successfully. The simulation shows the co-channal collision repair scheme is implemented correctly in the simulation model.

In summary, the functions of the proposed SSMAP based cross layer routing protocol have been implemented and validated step by step. The simulation model proved to be implemented correctly and it does behave as expected.

6.3   Concluding remarks

This chapter firstly presents the simulation modelling for the proposed SSMAP based cross layer routing protocol using OPNET simulation tools. The system design is divided into network model, node model and process model. Each model is described in detail. The functionality of the sub-channel allocation scheme and routing protocol are implemented in the process model.

In order to validate if the simulation model is implemented correctly, a series of simulations are carried out in section 6.2. Trace file is used to validate the implementation of the sub-channel allocation scheme and the routing protocol. The trace files are created by C code in OPNET to trace the signalling messages and record the living states of the different layers in each node.

Section 6.2.1 validates the route discovery mechanism as well as the sub-channel allocation scheme along with the routing discovery process. Five sessions are triggered sequentially. The trace files show that the route discovery process is implemented correctly for each session. The route can be found successfully for each session. The simulation results also show that the sub-channel allocation scheme is implemented correctly in the route discovery process.

In section 6.2.2, the route maintenance scheme is validated. In this simulation, all nodes in the network are moving. Two trace files are created to trace the link breakage and co-channel interference respectively. The simulation results show that the node can successfully detect the forthcoming link breakage by monitoring the active sub-channel and a new route can be established before the link's real breakage. Moreover, the trace

file also shows that when an interference node is approaching, the sub-channel reallocation is running successfully through the *SREA/SRRP* mechanism.

The next chapter evaluates the SSMAP based QoS routing protocol in various scenarios and presents the simulation results.

Chapter 7

Performance evaluation and analysis of the proposed SSMAP based QoS routing protocol

In this thesis, the proposed routing scheme is tested in two different scenarios with different node topologies: random distribution topology where each node is arbitrarily distributed in the network and uniform distribution topology where two restrictions are added upon the random distribution topology: firstly there is no isolated node and secondly the maximum number of neighbouring nodes for each node is 6. All the simulations are processed at packet level.

Table 7   Simulation configurations

Attribute Name	Value
Transmit power (dBm)	0
Number of subcarriers	64
Subcarrier frequency spacing	0.3125 (20M/64)
Receiver sensitivity (dBm)	-65
Packet size (B)	1024
Path loss model	Open space Two-Way Ground model
Modulation scheme	QPSK
Error correction threshold	0
Noise figure	1
Bandwidth (MHz)	20
SIR_{min}	20dB [99]
Minimum frequency band (GHz)	2.4

7.1 Simulation environment

In this section, an ad hoc network of 30 nodes is generated in an area of 1000 by 1000 meters. The transmission range for each node is 250 meters. Packets are generated according to a Poisson distribution with a Maximum Transmission Unit (MTU) of 1024B.

The simulation results are the average of 10 different random seeds and open space environment is assumed. Some important parameters are shown in Table 7 which are selected to closely match one optional specification in the IEEE 802.11n standard [7].

7.2 Performance analysis of the SSMAP in ad hoc networks

7.2.1 System performance evaluation for SSMAP

The simulations of the signal strength based medium access control algorithm are implemented using OPNET simulator. First of all, the simulations are taking place in a uniform node distribution scenario. 30 nodes are distributed in a 1000 by 1000 meters area. In this experiment, the total subcarriers are divided into 8 sub-channels. The minimum reception SIR_{min} in this experiment is set to 20 dB. Each node in the network starts a session at a random time from 0 to 10 seconds and the session lasts until the end of the simulation. The simulation time is set to 10 minutes. Figure 55 shows the average network throughput with different values of signal-to-interference ratio (SIR_{TH}). The throughput calculated here is actually equal to system throughput because the packet counted is the data packet received in each receiver. From the simulation results it can be seen that, when the SIR_{TH} is equal to the minimum reception SIR_{min}, there is packet loss due to the co-channel interference. A value of SIR_{TH} increases, the system throughput also increases. However, when the SIR_{TH} continues increasing, the system throughput starts decreasing. This is because when the SIR_{TH} is more than a certain value, the number of available sub-channels for each mobile node decreases. If a node cannot find suitable sub-channels for the requested

session, the packets will be discarded. From the simulation results it can be seen that, the system throughput reaches its maximum value when $SIR_{TH} = 1.2 \times SIR_{min}$.
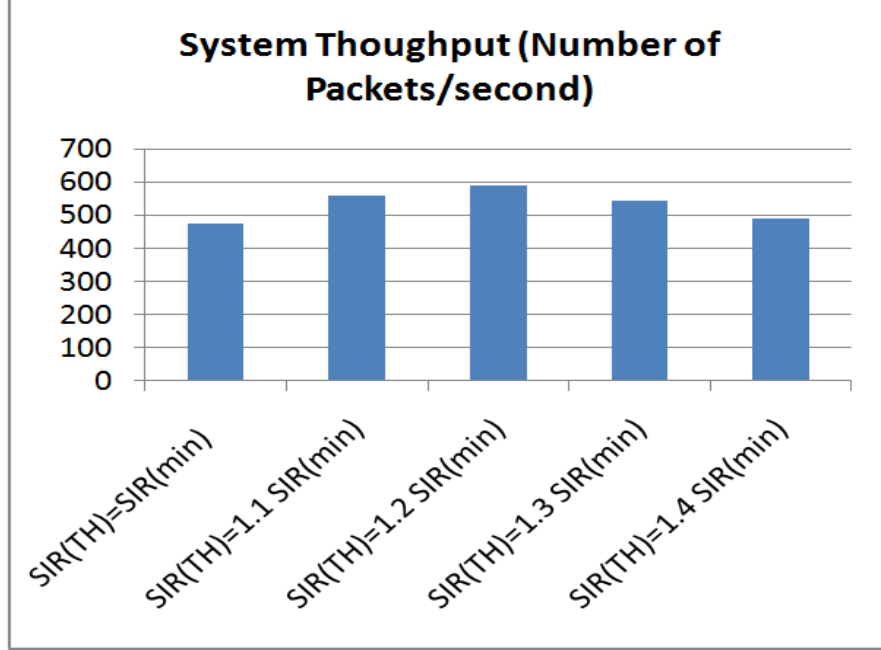

Figure 55 System throughput for different SIR threshold in uniform distribution scenario

After evaluating the SSMAP for an initial uniformly distributed network topology, another simulation is executed to investigate the system performance for the case where the nodes are initially randomly placed in the network. In this scenario, 30 nodes are randomly distributed in an area of 1000 by 1000 meters. The simulation parameters are the same as the ones shown in Table 6. The simulation time is set to 10 minutes as well. The simulation results are averaged over ten different simulations with different seeds.
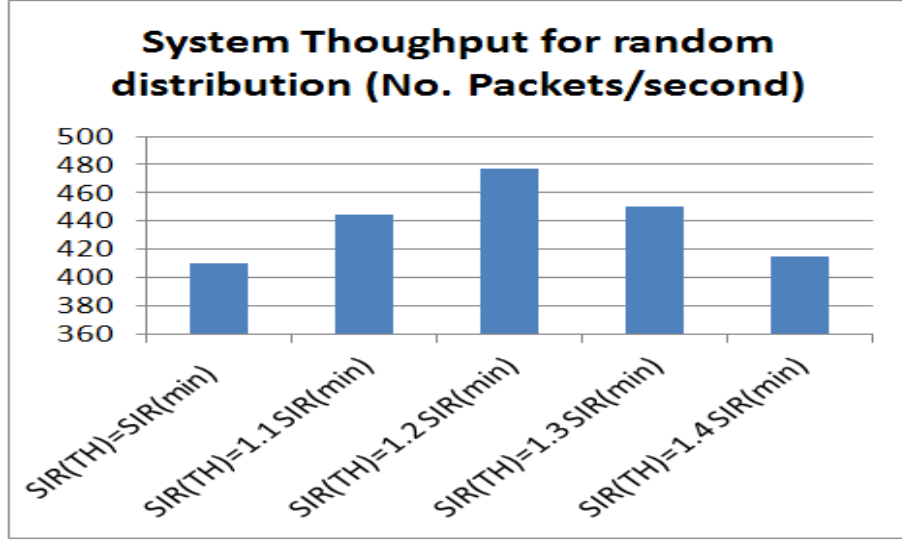
Figure 56 System throughput for different SIR threshold in random node distribution scenario

Figure 56 illustrates the simulation results for the random node distribution scenario with different signal to interference ratios. Each node in the network starts a session at a random time from 0 to 10 seconds and the session lasts until the end of the simulation. Compared to Figure 55, the system throughput is largely reduced. This is because with the random node distribution, there are some isolated nodes whose transmission coverage does not reach any other node. Any session started or ended in those nodes will fail. Moreover, there are some areas where many nodes are close to each other increasing the congestion and subsequently increasing the interference and reducing the availability of sub-channels. Consequently, the packet dropping rate will increase as well. However, it can be seen from Figure 56 that, similarly to the uniform node distribution case, the system throughput reaches its highest point when the pre-defined carrier sensing threshold SIR_{TH} is equal to $1.2\ SIR_{min}$. Therefore the value $SIR_{TH} = 1.2\ SIR_{min}$ will be used for the remaining experiments.

7.2.2 Comparison of SSMAP with other multi-band MAC protocols in an uniform node distribution topology

In order to compare the proposed OFDMA based SSMAP with other multi-band medium access control protocols. Another experiment is designed. The experiment compares the performance of the SSMAP with the performance of a single-radio multi-channel MAC protocol (SM), a multi-radio multi-channel MAC protocol (MM) and other OFDMA based medium access protocol. For the SM, the approach described in MMAC [102] is used and for MM, the approach MM-MAC in [103] is adopted with similar input parameters. The CTRMA (Concurrent Transmission or Reception Multiple Access) [17] is used as an OFDMA based MAC for ad hoc networks to be compared with SSMAP. In this experiment, only the MAC protocol will be tested and compared. 30 nodes are firstly uniformly distributed in a 1000 by 1000 meters ad hoc network. The number of sessions requested by the nodes varies from 5 to 25. Each session randomly starts from 0 to 8 minutes and lasts 2 minutes. The simulation time is 10 minutes. Each active node randomly selects a destination node and the shortest path from the source node to the destination node is chosen as the route between them to forward the data packet.
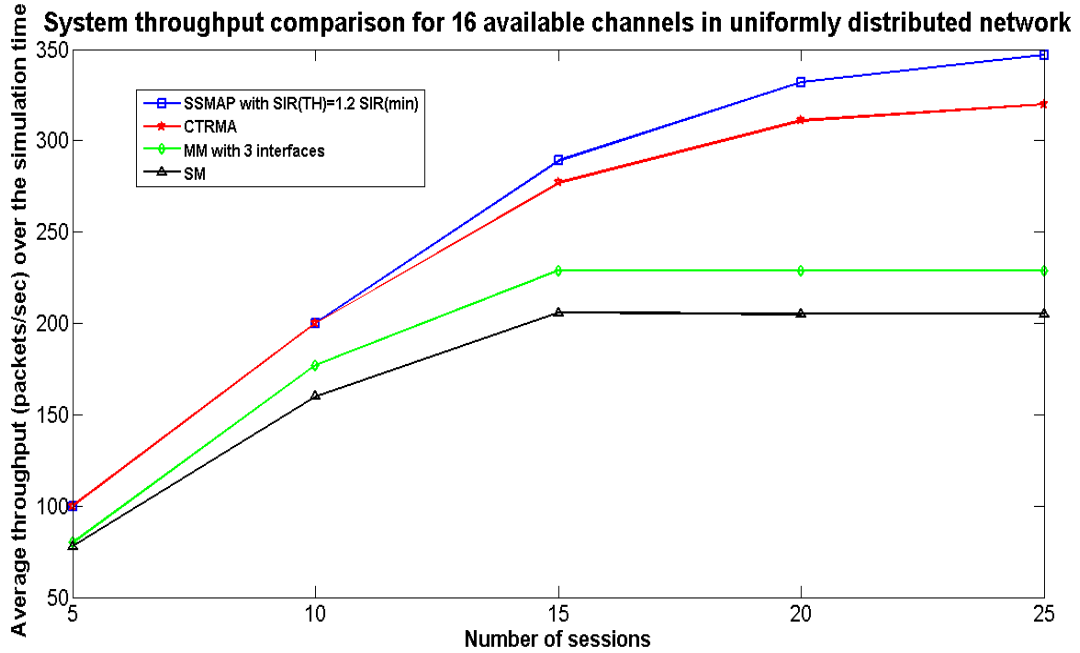
Figure 57 Throughput comparison in uniformly distributed network with 16 available channels

Figure 57 shows the achievable throughput of SSMAP, SM, MM and CTRMA. The number of total data sub-channels for SSMAP and CTRMA and the number of frequency band for SM and MM is set to 16. When the number of session is small, all schemes have good performance since there are enough available channels. However, when the number of sessions continuously increases, the system throughputs for SM and MM are significantly reduced due to the lack of efficient resource allocation scheme and increased co-channel interference and collisions. From the system throughput point of view, the performance of CTRMA is just slightly worse than SSMAP in most of cases.
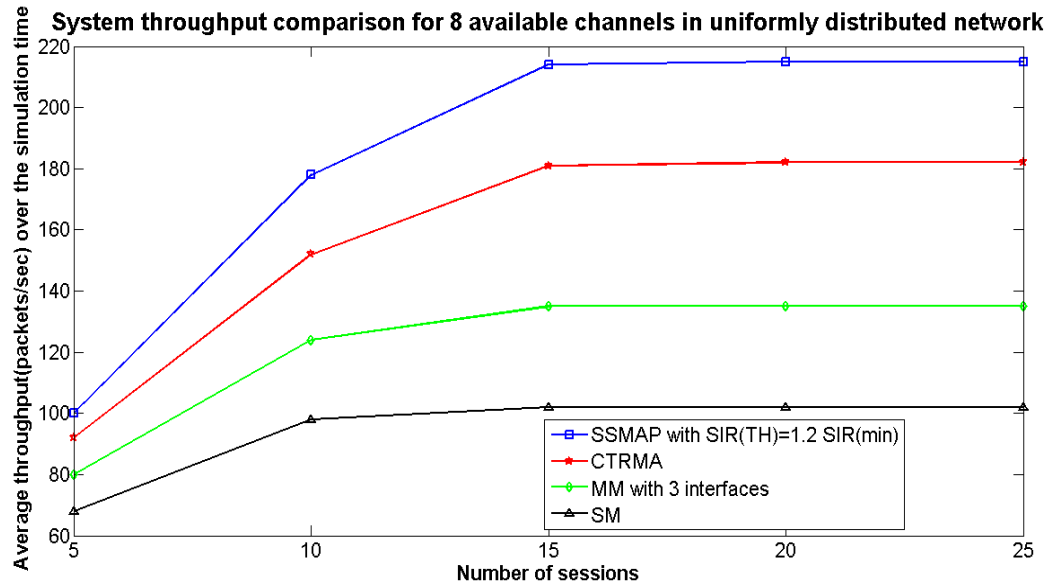
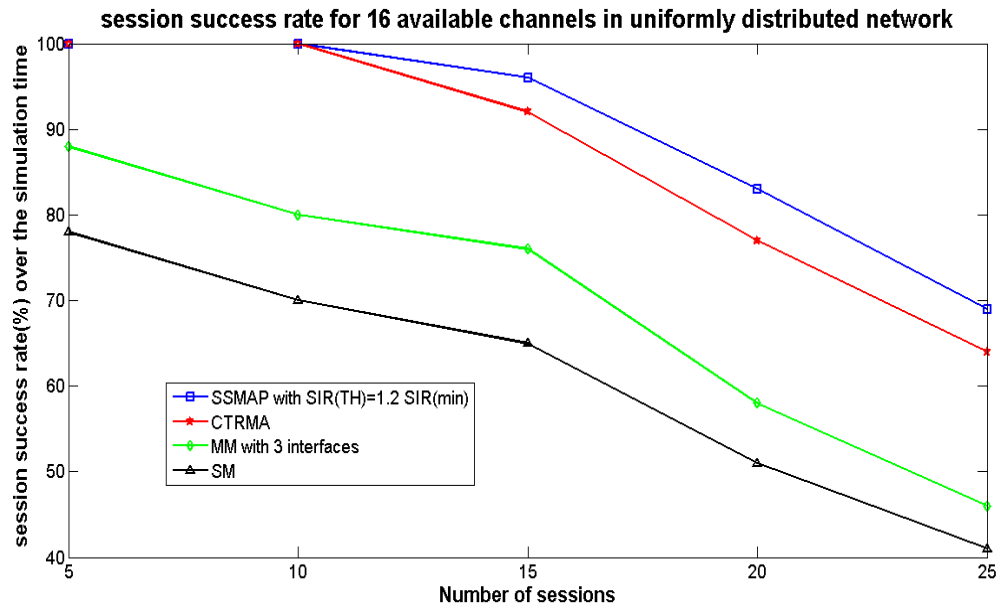**Figure 58 Throughput comparison in uniformly distributed network with 8 available channels**

Figure 59 Session success rate in uniformly distributed network with 16 channels

session success rate for 8 available channels in uniformly distributed network
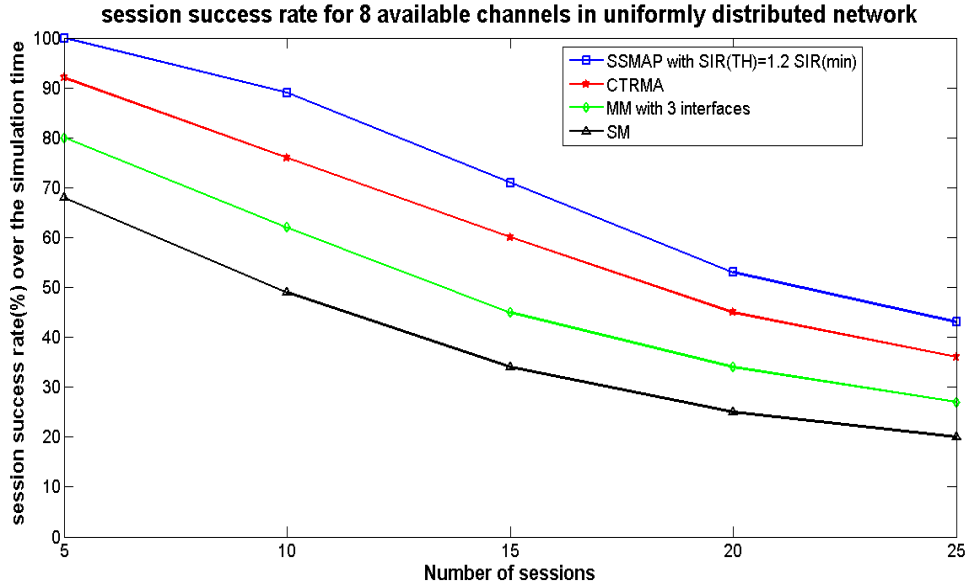
Figure 60 Session success rate in uniformly distributed network with 8 channels

Figure 58 shows the comparison of the system throughput when the number of available orthogonal channels for all schemes reduces to 8. From the simulation results it can be seen that the system performance for all schemes is reduced especially when the number of sessions is more than 15. However, with the proposed sub-channel allocation scheme, the system performance for SSMAP can still maintain reasonable level even when the number of sessions is more than 15.

Figure 59 shows the simulation results for the session success rate metric. The total available number of channels is set to 16. Because the path from the source node to the destination node has already been established when a session starts, the reason for a failed session can only be the lack of available channels along the path and collisions due to the co-channel interference. From the simulation results it can be seen that because of their sub-channel allocation schemes the SSMAP and the CTRMA have a higher session success rate than SM and MM. When the number of session increases,

the session success rate is reduced for SM and MM due to lack of the available channels. Although the session success rates for both SSMAP and CTRMA are more than 80 percent while the number of sessions is small, when the number of sessions is more than 15, the session success rate for SSMAP is much greater than CTRMA. Figure 60 shows the session success rate comparison for only 8 available channels for each scheme. When the number of available channel is reduced to 8, the session success rate for all schemes is degraded. However, the degradation for SM and MM is much more severe than SSMAP and CTRAM.
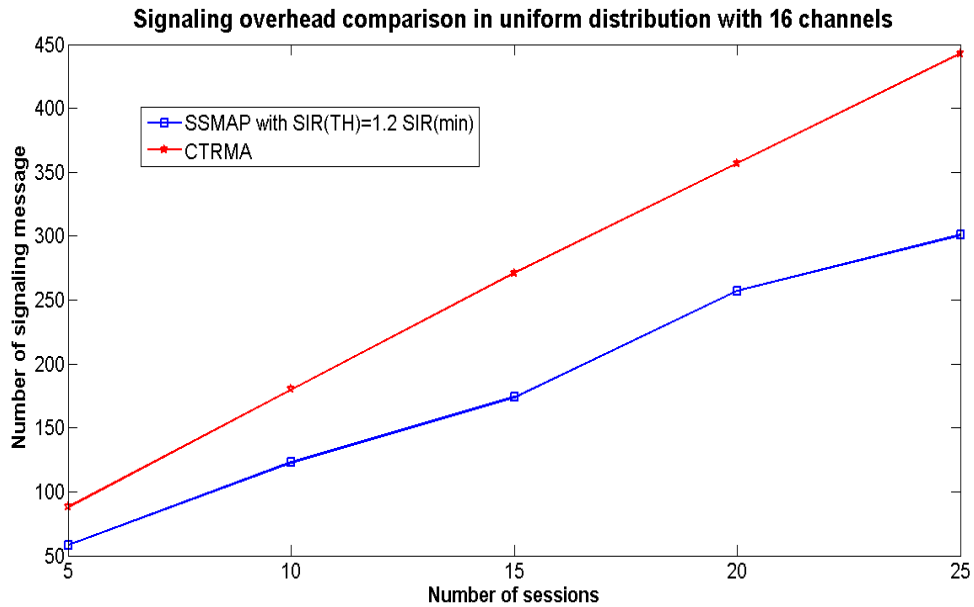
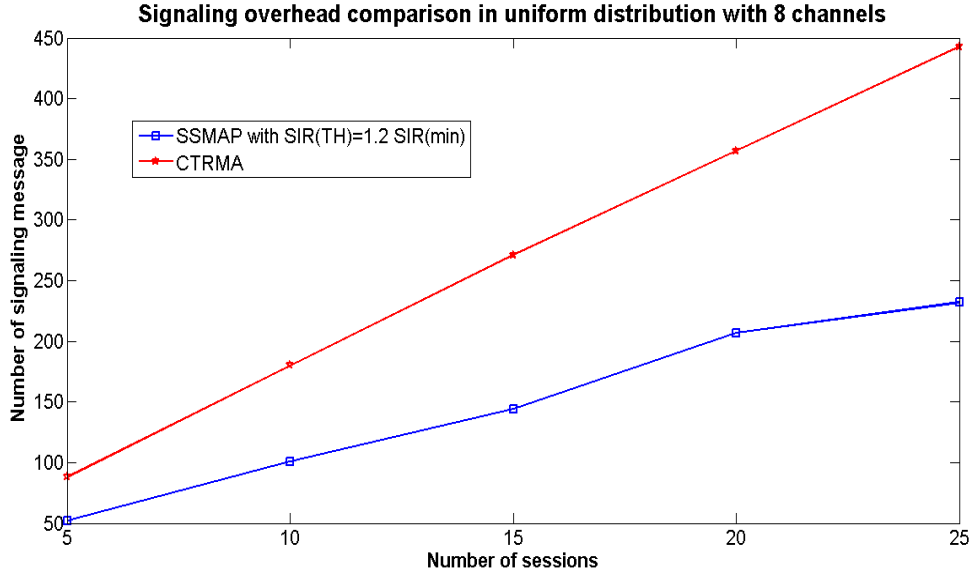**Figure 61 Signalling overhead comparison in uniform distributed network with 16 channels**

Figure 62 Signalling overhead comparison in uniform distributed network with 8 channels

Finally, the system signalling overhead is investigated. Figures 61 and 62 show the simulation results for the comparison of SSMAP and CTRMA in terms of overall signalling overhead for number of available channels equal to 16 and 8 respectively. The simulation results show a significant improvement for SSMAP over CTRMA in terms of signalling overhead. This is because SSMAP is an on-demand sub-channel allocation scheme. Each node does not need to periodically exchange local information with its neighbours which is a necessary procedure in CTRMA. Moreover, once the route is established, the active nodes do not need to process the *RTS/CTS* mechanism.

7.2.3 Comparison of SSMAP with other multi-band MAC protocols in a random node distribution topology

In the previous section the proposed SSMAP was compared with other relative schemes in an initial uniformly distributed network topology. In this scenario, the system

performance will be investigated for the case where the nodes are initially randomly placed in the network. 30 nodes are randomly distributed in an area of 1000 by 1000 meters. The other simulation settings are the same as the ones in section 7.2.2. The simulation time is set to 10 minutes. The number of sessions requested by the nodes varies from 5 to 25. Each session randomly starts from 0 to 8 minutes and lasts 2 minutes. The simulation results are averaged over ten different simulations with different seeds.
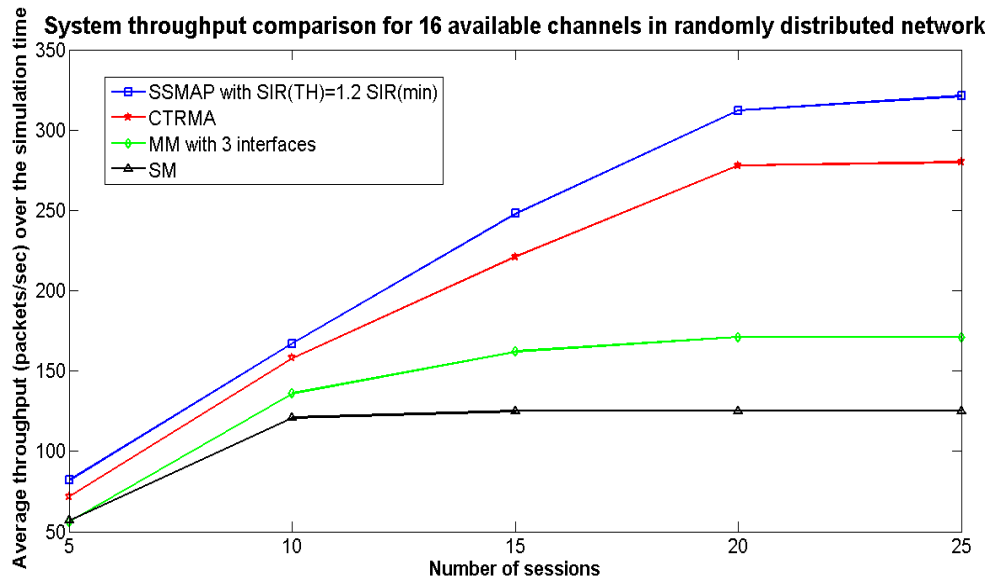

Figure 63 Throughput comparison in randomly distributed network with 16 channels

Figure 63 shows the achievable system throughput for SM, MM, CTRMA and SSMAP for 16 available channels. Compared to Figure 57, the system throughput is largely reduced for all schemes. This is because in the random node distribution topology, there are some isolated nodes whose transmission coverage does not reach any other node. Any session started or ended in those nodes will fail. Moreover, there are some areas where many nodes are close to each other increasing the congestion and subsequently

increasing the interference and reducing the availability of sub-channels. Consequently, the packet dropping rate will increase. This is even worse when the number of available channels is reduced to 8 (see Figure 64). However, in any of the cases, SSMAP always has the highest system throughput.
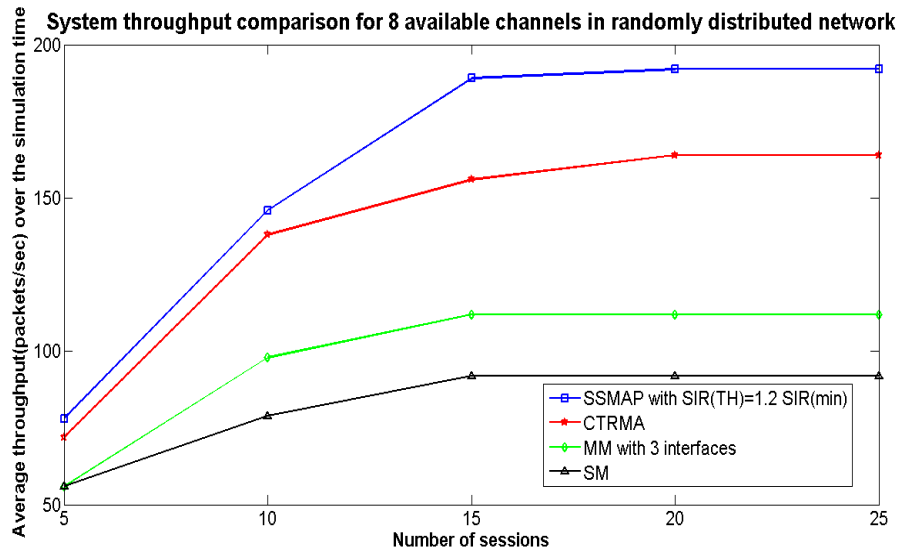

Figure 64 Throughput comparison in randomly distributed network with 8 channels
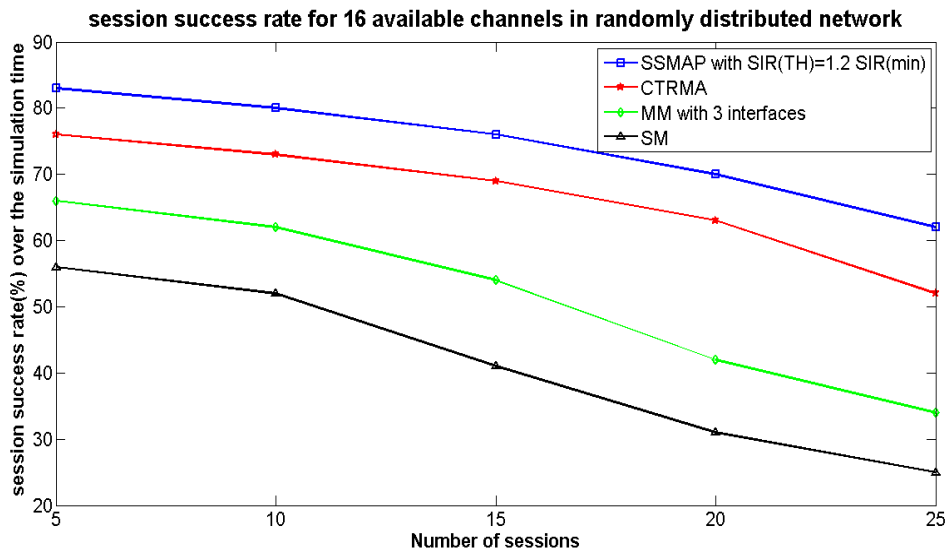

Figure 65 Session success rate comparison in randomly distributed network with 16 channels

Figures 65 and 66 show the simulation results for the session success rate when the number of available channel is 16 and 8 respectively. Similar to the system throughput performance, because of the isolated nodes and high congestion area, the session success rate for all schemes is largely reduced compared to the ones in uniform distribution scenario.

Figures 67 and 68 show the simulation results for the signalling overhead comparison between CTRMA and SSMAP in a random node distribution scenario. Compared to Figures 61 and 62, the signalling overhead for SSMAP is even lower than the ones in uniform distribution scenario in both cases. This is because SSMAP is an on-demand sub-channel allocation scheme and if one session is failed, the sub-channel allocation mechanism will not be triggered. By contrast, the signalling overheads for CTRMA in both scenarios maintain a high level, because every node in CTRMA has to periodically broadcast update information with its neighbours when the network is established.
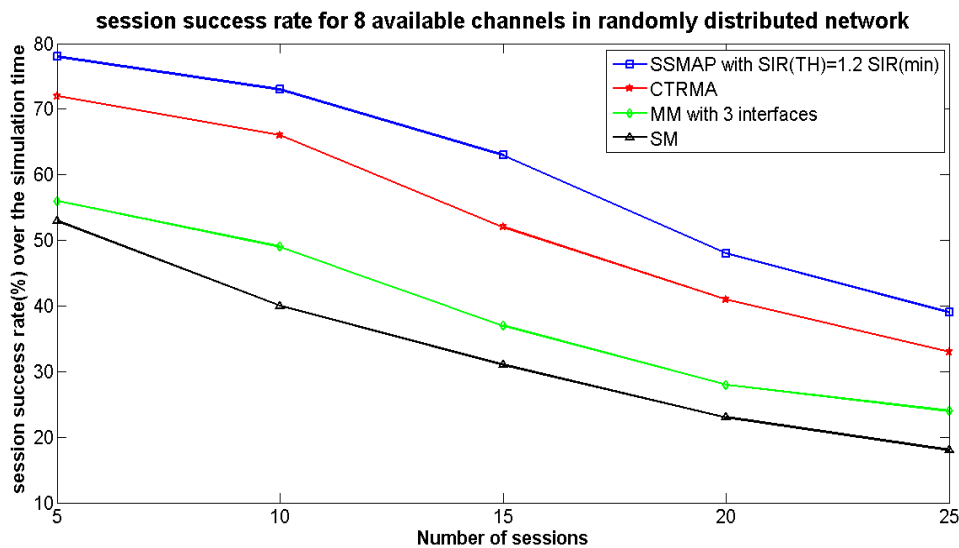

Figure 66 Session success rate in randomly distributed network with 8 channels
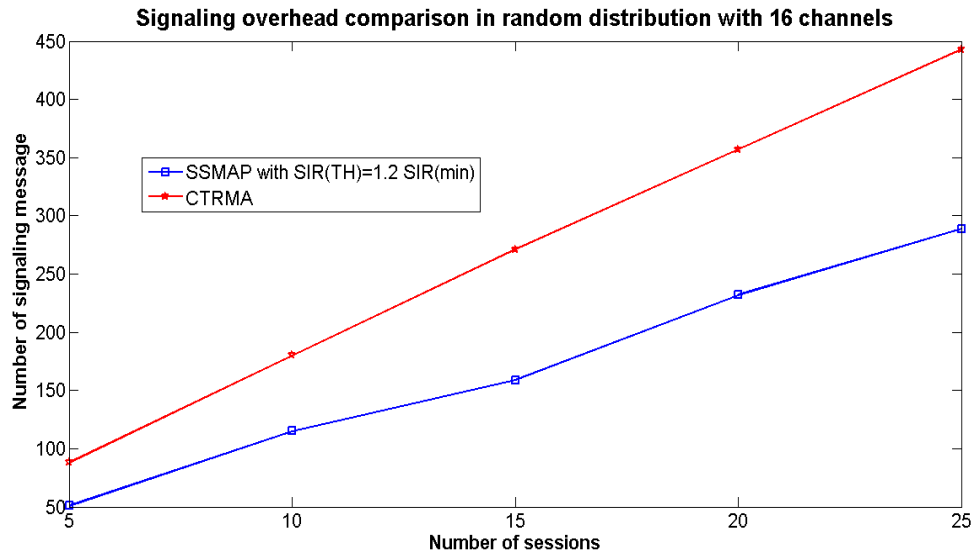
Figure 67 signalling overhead comparison in randomly distributed network with 16 channels
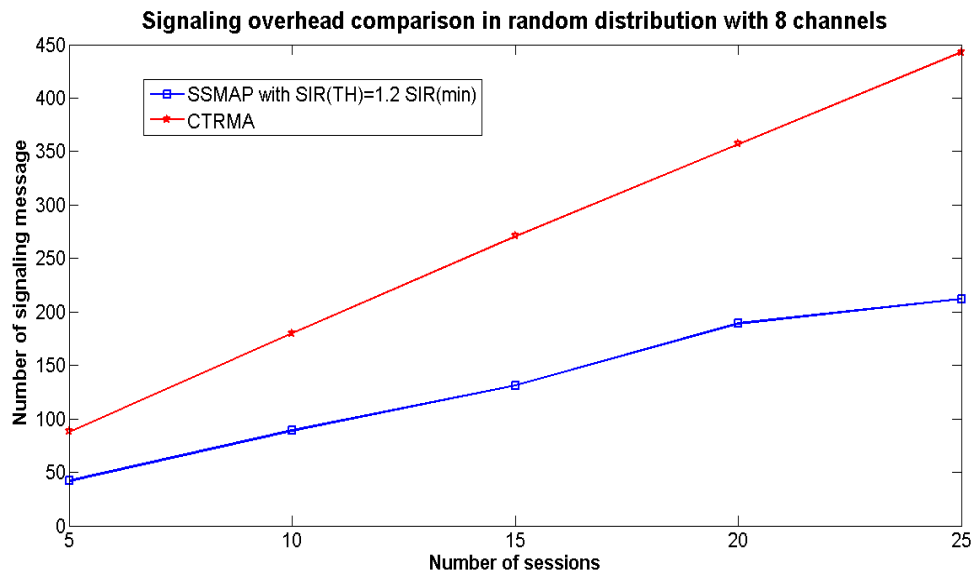

Figure 68 Signalling overhead comparison in randomly distributed network with 8 channels

7.3 Simulations and results of the proposed signal strength based QoS routing protocol

The performance of the proposed SSMAP based QoS routing protocol is evaluated with the simulation results in this section. The system model has been described in the previous chapter. Each mobile node is configured according to the node model described in 6.1.2. This chapter will first investigate the performance of the QoS routing protocol with other alternative proposals in static environment where every node in the network is fixed. Then the performance of the proposed routing protocol in a mobile environment will be analysed through simulations.

7.3.1 QoS performance analysis for the proposed routing protocol

Firstly, the QoS performance for the proposed scheme is evaluated and analysed. Once a session is generated at one node, the application process will assign an end-to-end delay requirement for that session by setting a maximum delay constraint in the route to the destination node the session can tolerant. There are 30 nodes randomly distributed in a 1000*1000 m^2 ad hoc network. Each node in the network starts a session at a random time from 0 to 10 seconds and the session lasts until the end of the simulation. The simulation time is set to 2 minutes. When a session is triggered in one node, the application process will set up the bandwidth and end-to-end delay requirement for this session and encapsulate this information into every packet of this session. Simulation results are shown in Figure 69.
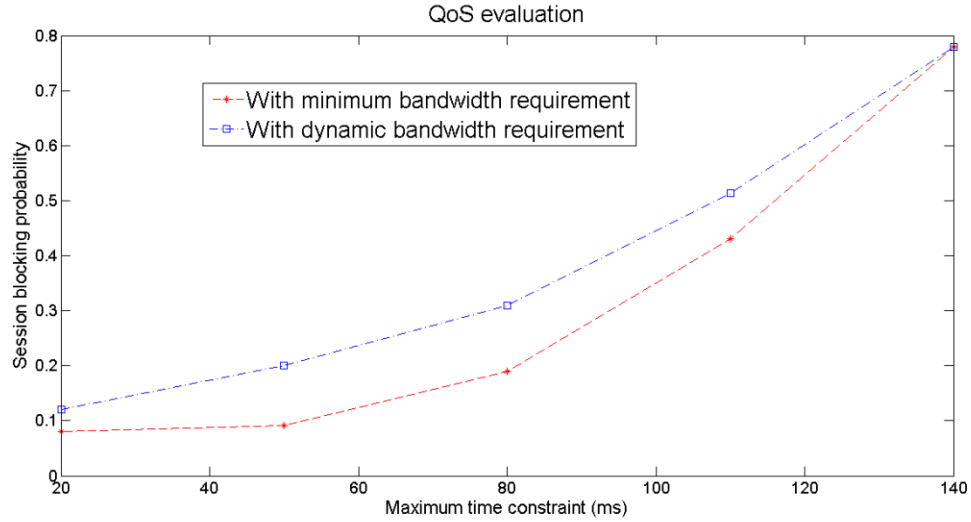
Figure 69 QoS evaluation for the bandwidth and end-to-end constraint

Two groups of simulations are performed. Each group runs five simulations with different end-to-end delay requirements from 20ms to 140ms. In the first group, the bandwidth requirement for each session is set to the minimum value where each session only needs one sub-channel. For the second group of simulations, each node assigns dynamic bandwidth requirement which the number of sub-channel needed is randomly chosen from 1 to 4 when the session is generated. From the simulation results in Figure 69 it can be seen that, for the case where the maximum time constraint is small (such as 20ms), both two scenarios have high blocking rate. This is because the destination which is far away from the source node will never receive packets from the source. As the maximum time constraint is increased, the blocking rate decreases significantly for both cases.

The simulation results show that the proposed routing protocol can successfully support both bandwidth and end-to-end delay requirements. However, if the QoS requirement increases, the success rate to find a proper route will decrease accordingly.

217

7.3.2 Comparison of the proposed QoS routing protocol with other alternative proposals in uniform distribution topology

In this experiment, 30 nodes are uniformly distributed in a 1000*1000 m^2 ad hoc network. Each active node in the network starts a session at a random time from 0 to 10 seconds and the session lasts until the end of the simulation. The simulation time is set to 2 minutes. The proposed SSMAP based routing protocol will be compared with SM, MM and CTRMA using the AODV routing protocol on top, since AODV is also an on-demand routing protocol and its functionality is close to the proposed QoS routing protocol. Figure 70 shows the comparison of signalling overhead between SSMAP based routing scheme with the other MAC layer protocols (SM, MM, CTRMA) that are using the AODV routing protocol on top. In this experiment, the number of sessions varies from 5 to 25 and the number of available channels for each scheme is set to 7. Each active node will randomly select a destination node and trigger the route discovery mechanism.
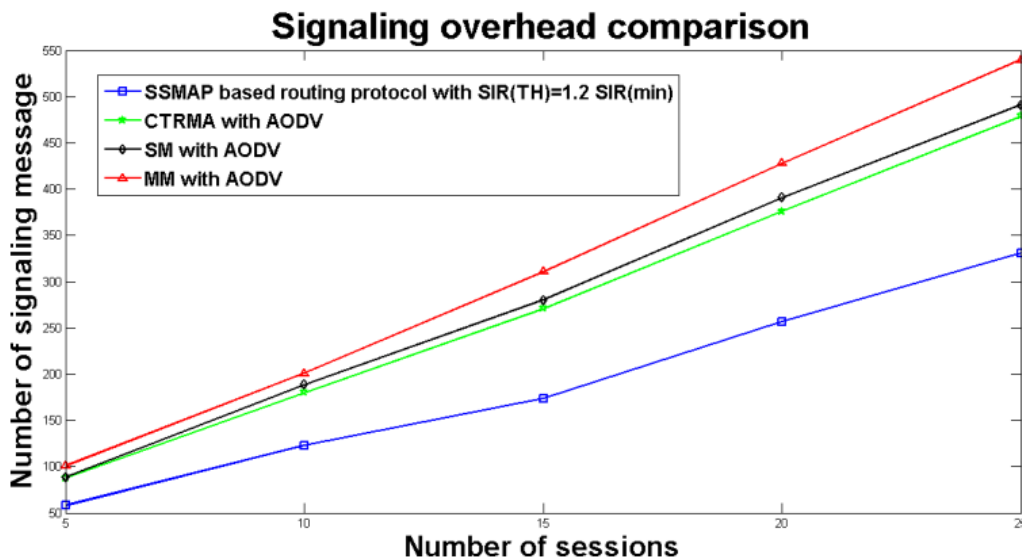

Figure 70 Signalling overhead comparison in uniform distribution topology

From the simulation results it can be seen that the signalling overhead of the proposed SSMAP based routing protocol is considerably lower than the other solutions. This is because in the case of the proposed routing scheme, each node does not need to exchange local information with its neighbours for sub-channel allocation while in any of the other schemes, periodic updates are exchanged among nodes.
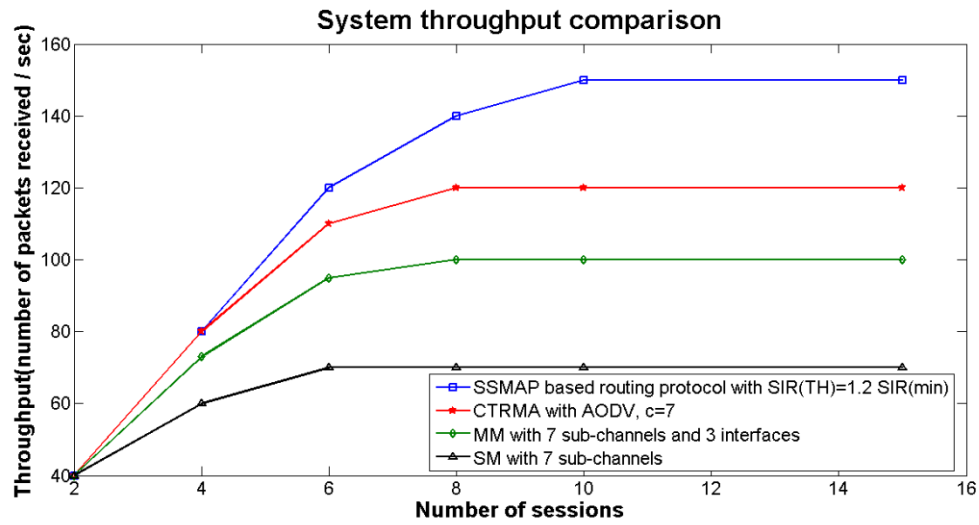

Figure 71 Comparison in terms of throughput in uniform distribution network

Next, the proposed SSMAP based routing scheme is compared with other proposals in terms of overall system throughput. Figure 71 shows the achievable throughput for the different schemes using different MAC layer protocols. In this experiment, the number of sessions varies from 2 to 16 and the number of available channels for each scheme is set to 7. From Figure 71, it can be seen that for any number of concurrent sessions, all the other schemes cannot match the performance of the SSMAP based routing protocol. Without proper sub-channel allocation strategies, when the number of concurrent sessions increases, the achievable throughput of SM and MM is significantly reduced

due to the lack of available sub-channels and increased co-channel interference and collisions.

7.3.3 Comparison of the proposed QoS routing protocol with other alternative proposals in a random node distribution topology

In this scenario, the SSMAP based routing protocol is compared with SM, MM, CTRMA using the AODV routing protocol on top. 30 nodes are randomly distributed in an area of 1000 by 1000 meters. The simulation parameters are the same as in the uniform distribution scenario. To compare the simulation results with the ones in the uniform distribution topology, the system throughput and the signalling overhead are also used as the criteria to evaluate the system performance. Figure 72 shows the simulation results for the signalling overhead of each compared scheme in the random node distribution scenario. Figure 72 shows that the total signalling overhead in the random node topology is higher than the one where the nodes were uniformly distributed for all of the proposed schemes. This is because in the random node distribution scenario, a route from source probably needs more hops to reach the destination node, consequently the route discovery signalling messages increase. However, the simulation results shows that in the random node distribution scenario the proposed SSMAP based routing scheme still has the least signalling overhead.

Figure 73 compares SSMAP with the other solutions in terms of overall system throughput in the randomly distributed node topology. Similarly, due to the isolation of some nodes and the high congestion in some areas, the system throughput for all the schemes is smaller than their own results in the uniform node distributed network. This effect is significantly worse for the SM and the MM based routing solutions. The reason

for that is because both SM and MM do not have a sub-channel allocation scheme. Moreover, in highly congested areas, large numbers of packets are dropped.
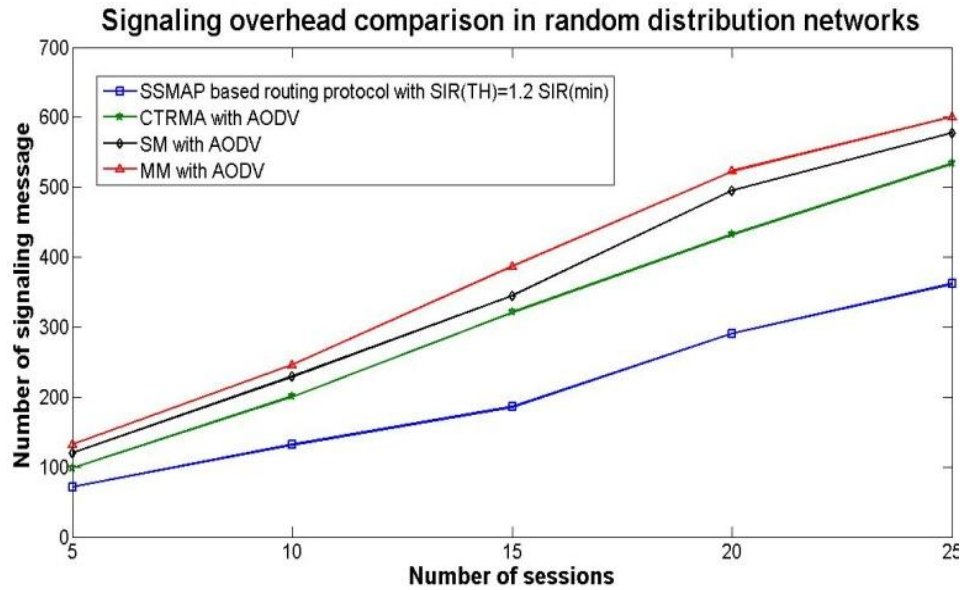

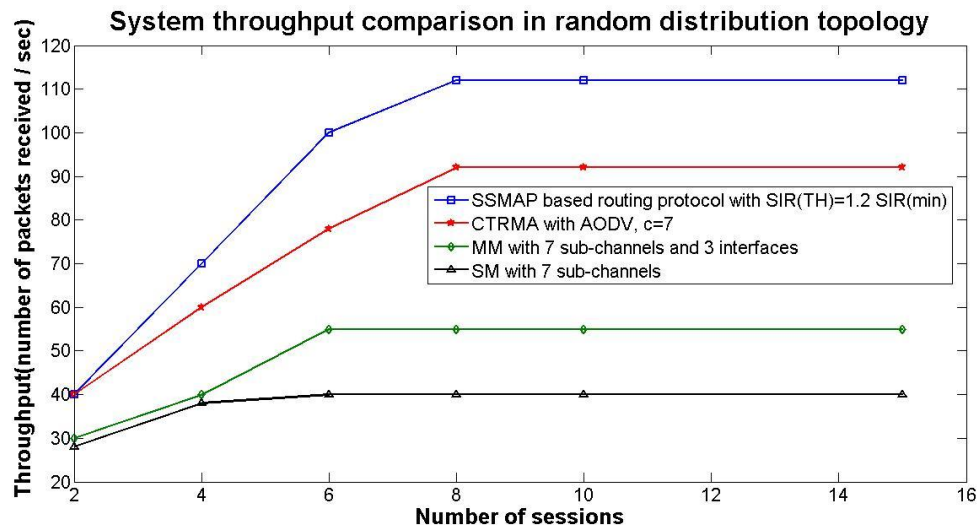**Figure 72 Signalling overhead in the random node distribution scenario**

Figure 73 System throughput for different routing schemes in the random node distribution

scenario

7.3.4 Performance analysis of the proposed routing protocol in a mobile environment in uniform distribution topology

The previous simulations are taking place in a static environment. In this section, the system performance will be evaluated in a mobile environment to evaluate the proposed routing maintenance scheme. A hundred nodes are distributed in an area of $2000 \times 2000 \, m^2$ ad hoc network. The location of a node is generated using a uniform distribution. Then transmission range for each node is 250m. Each active source node randomly selects a destination node and starts a session at a random time from 0 to 8 minutes. Each session lasts 2 minutes. The simulation time is set to 10 minutes. Every node in the network can move arbitrarily inside the network. The mobility model is described as follow. Each node will randomly generate a destination position within the simulation area. It will move to the destination with a pre-defined speed and waits there for a random period of time (0 to 5 seconds) and move to the next randomly generated destination position in a pre-defined speed.
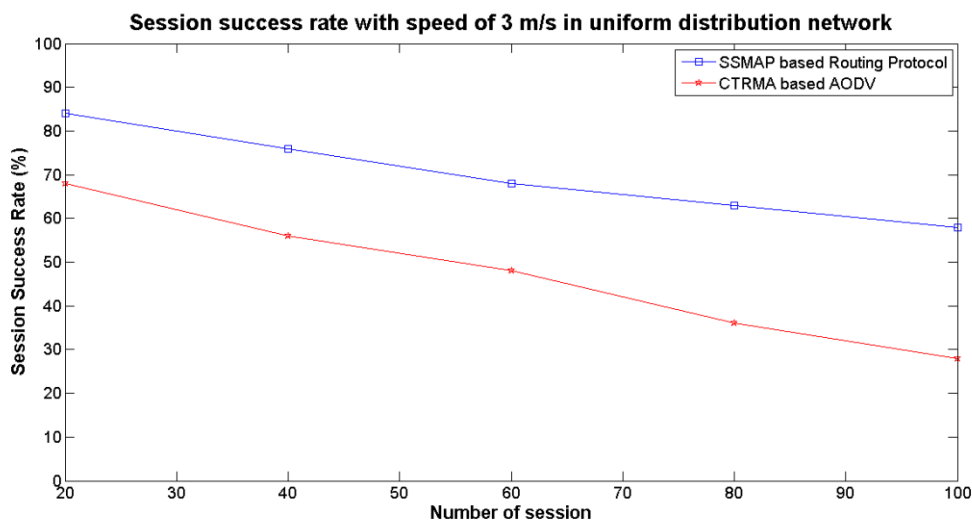
Figure 74 Session success rate with speed of 3m/s
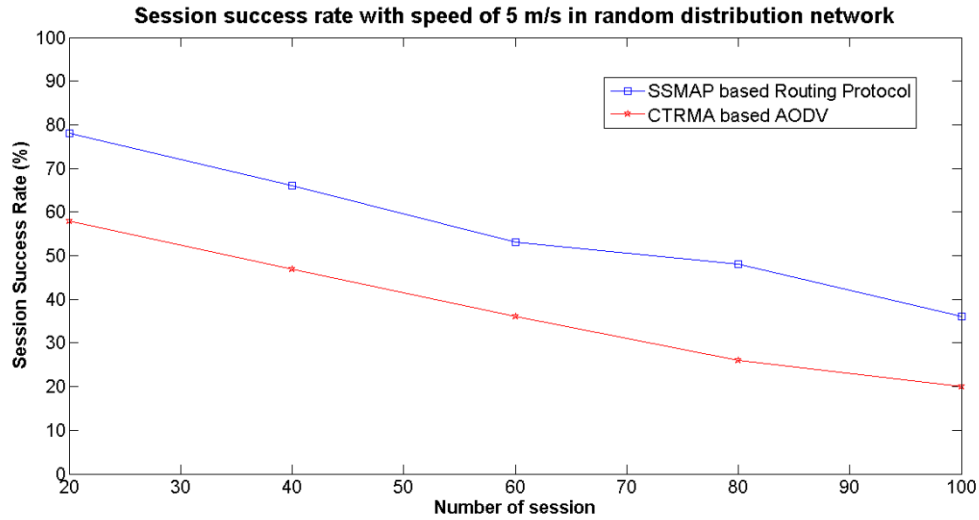
Figure 75 Session success rate with speed of 5m/s

Figure 76 Session success rate with speed of 10m/s

The proposed QoS routing protocol is compared with CTRMA using the AODV routing protocol with normal AODV routing maintenance scheme. Three groups of simulations are set up with node's moving speed from 3 m/s to 10 m/s. The number of sessions varies from 20 to 100. The session is only considered successful if 90% of its packets are

received successfully in the receiver. The simulation results are illustrated in Figures 74, 75, 76 for different moving speeds. The number of sub-channels for two schemes is 16. From the simulation results it can be seen that in any case, especially when the node's moving speed is high, the proposed route maintenance scheme can significantly increase the session success rate in a reasonably mobile scenario due to the pro-active procedures taken before any link breakage compared to the reactive manner used by AODV. However, when the speed of the nodes increases the packet dropping rate significantly increases for both schemes. This is because when the speed of the nodes reaches a certain value, the link breakage rate increases and the link recovery fails more often because of the lack of the available nodes for the new needed links.

7.3.5 Performance analysis of the proposed routing protocol in a mobile environment in random distribution topology

In this scenario, a hundred nodes are randomly distributed in an area of $2000 \times 2000 \ m^2$ network. The simulation parameters are set to the same as in uniform distribution scenario. In this simulation, the proposed routing maintenance scheme is also compared with CTRMA based AODV routing maintenance scheme in terms of session success rate for different mobile node speeds. The speed also varies from 0 m/s to 10 m/s. The number of sessions also varies from 20 to 100. The simulation results are averaged over ten experiments with different seeds.

Figures 77, 78, 79 show the simulation results with different velocities. From the simulation results it can be seen that in the random node distribution scenario, the system performance of the proposed scheme does not reduce significantly compared to

the uniform node distribution scenario. This is because even though there are some

isolated nodes, they might move away from the isolated area.

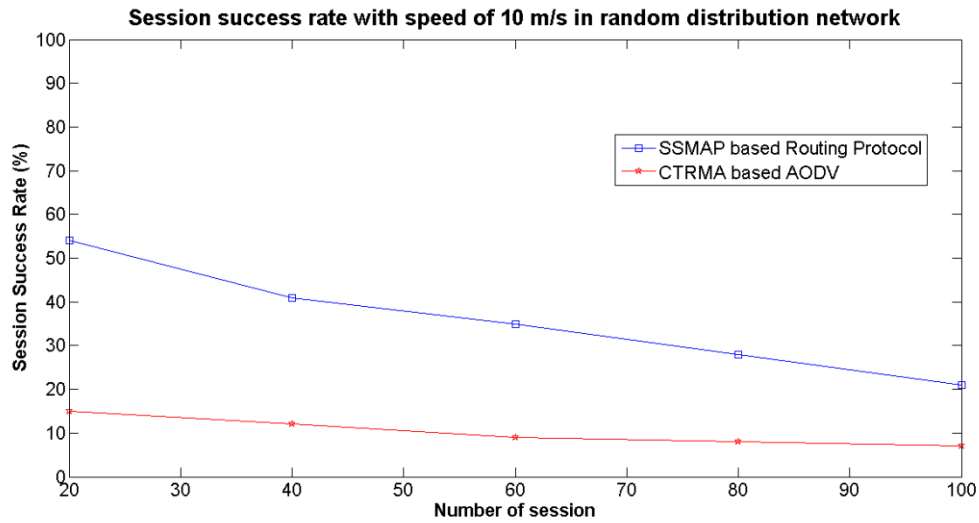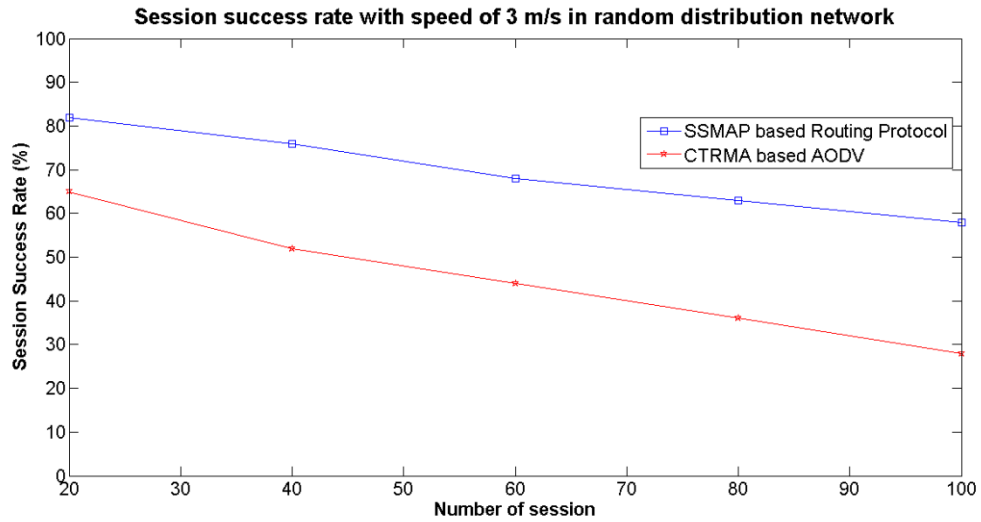**Figure 77 Session success rate with velocity 3m/s in random distribution network**
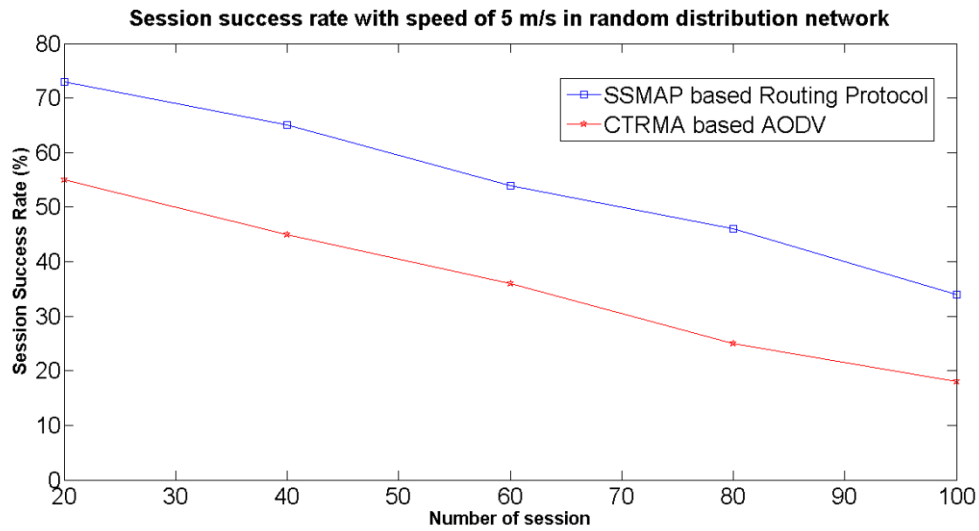

Figure 78 Session success rate with speed of 5m/s in random distribution network
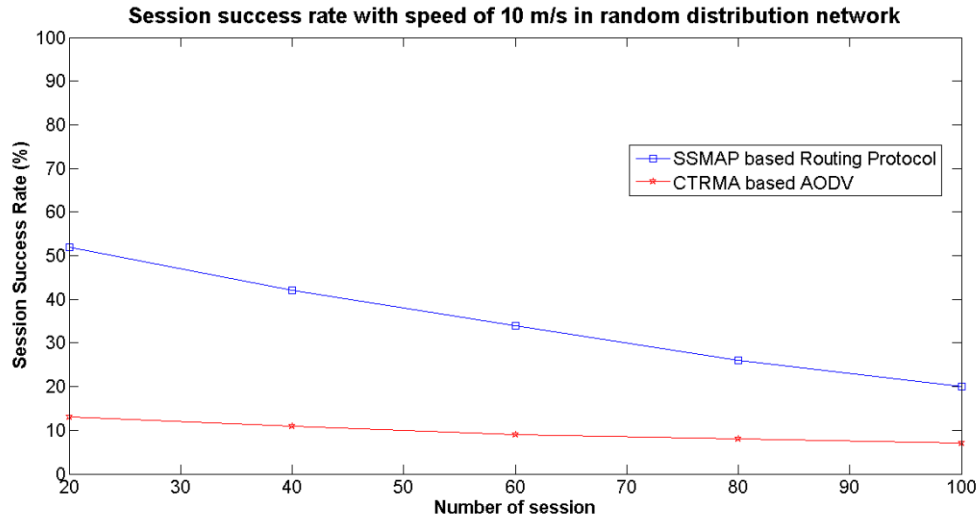
Session success rate with speed of 10 m/s in random distribution network

7.3.6 System performance evaluation in a large scale network

In the previous simulations, it shows that the proposed SSMAP based routing protocol has a good performance compared with other alternatives in a small ad hoc network with 30 nodes. In order to fully investigate the proposed solution in various situations, in this section, SSMAP will be evaluated in a large scale network. In this scenario, 1000 nodes are uniformly distributed in an 8000 by 8000 meters area. The transmission range for each node is 250 meters. The total number of sub-channels is 16. The number of active sessions varies from 10 to 50. This experiment only evaluates the SSMAP performance in a large scale network. Therefore, each active node will find a destination node and choose the shortest path from the source node to the destination node. SSMAP will perform the resource allocation alone the path. Once the sub-channels have been successfully allocated along the path, the source node will start to forward the data packets. Otherwise if there are not enough sub-channels, this session will fail. Each session will be triggered from a random time between 0 to 30 minutes and last 10

226

minutes. The simulation time is 60 minutes. Two sets of simulations are taking place in this experiment. In the first experiment, each active node will randomly select a destination node and in the second experiment, each active node will only select a destination node which is within 4 hops away from it.
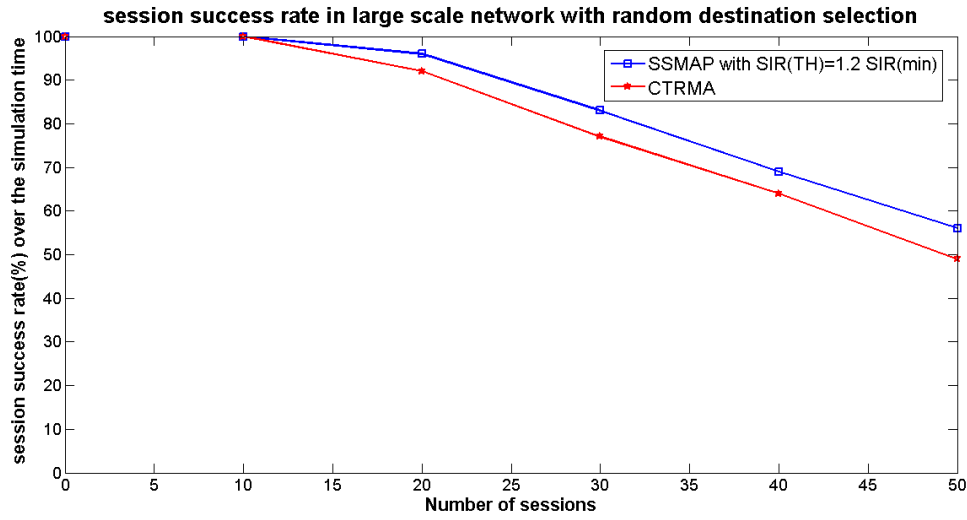
Figure 80 Session success rate in large scale network with random destination node selection
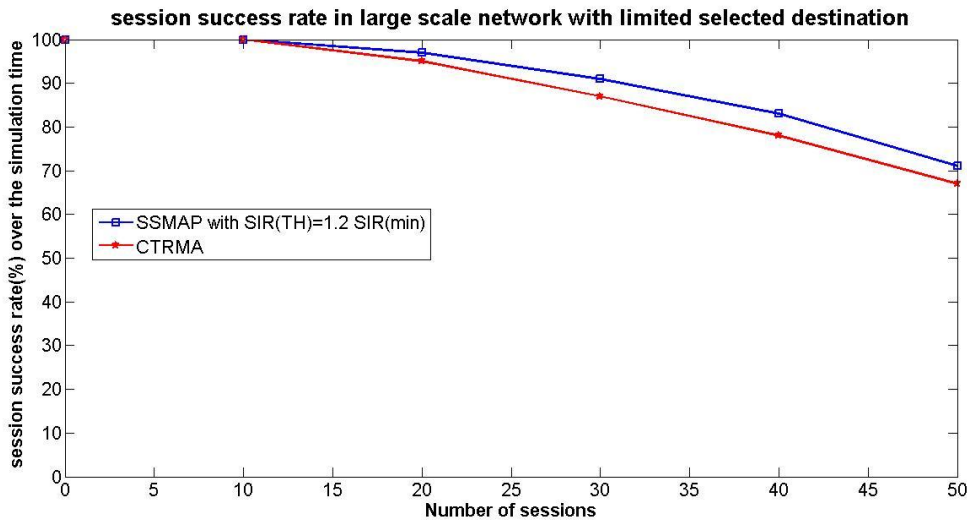
Figure 81 Session success rate in large scale network with limited destination node selection

Figures 80 and 81 show the simulation results for these two experiments respectively in terms of session success rate. From the results it can be seen that, in a large scale network, when the number of sessions is small, both schemes can have good performance. However, when the number of sessions increases, the collisions occur and the session success rate reduces in both cases. Moreover, for both schemes, the session success rates are higher in experiment two where the destination node is within four hops away from the source node. The reason for that is if the destination node is far away from the source node, this session might occupy more network resources which leaves less resources for others.
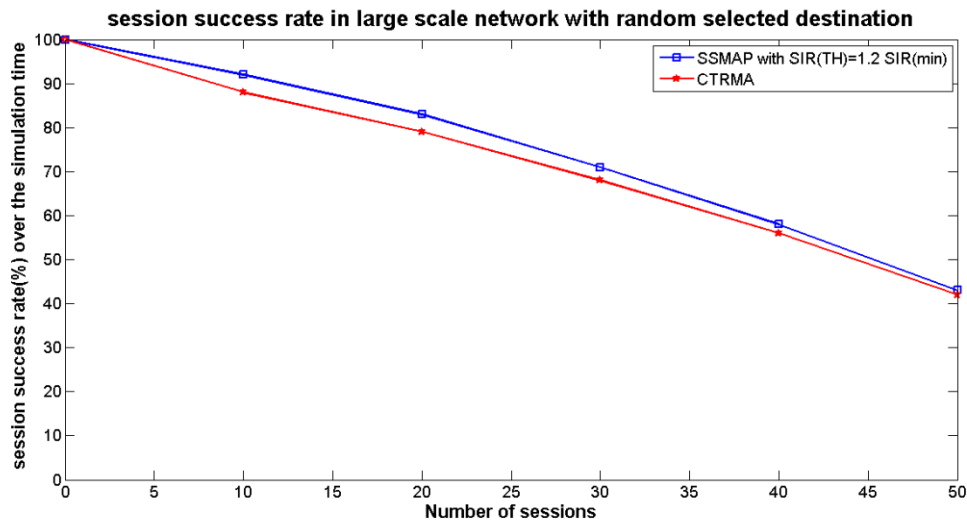
Figure 82 Session success rate with 8 total sub-channels for limited destination node selection
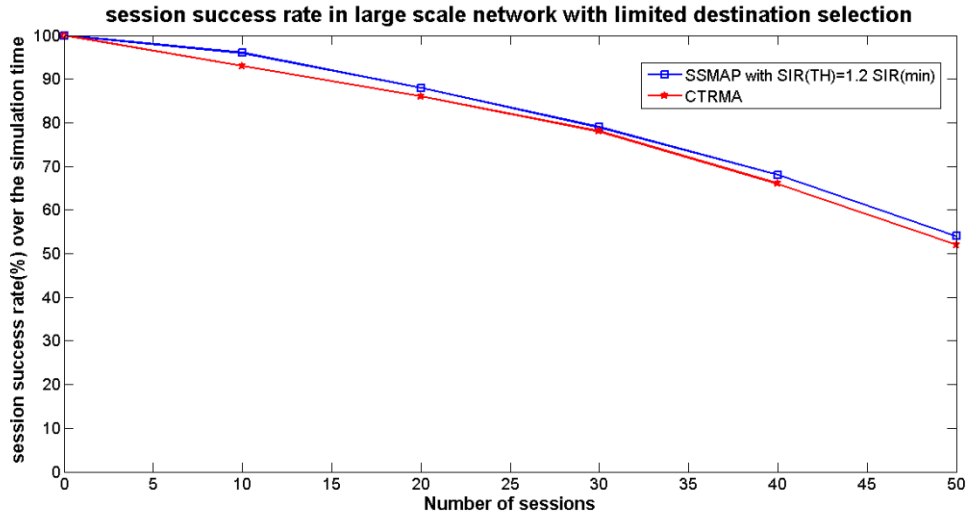
Figure 83 Session success rate with 8 total sub-channels for random destination node selection

Figures 82 and 83 are the simulation results when the total available sub-channels is reduced to 8. Compared to Figures 80 and 81, the session success rate is reduced in both cases. However, when the number of sessions is small (less than 20), the session success rate for the SSMAP with 8 sub-channals can still maintain a good level. However, as the number of sessions continuously increases, the session success rate is reducing rapidly due to the lack of available sub-channels to fulfil all the requirements.

7.4 Concluding remarks

In this chapter, the proposed SSMAP based routing protocol has been fully evaluated through various simulations in different scenarios. Section 7.1 presents the simulation parameters used in the simulations which is based on one of the options of the IEEE 802.11n standard.

In section 7.2, the signal strength based medium access protocol has been evaluated in both uniform and random node distribution scenarios. The proposed SSMAP is

compared with other multi-band MAC protocols and OFDMA based MAC protocol in terms of system throughput and session success rate. The simulation results show that the performance of the proposed SSMAP MAC protocol is better than other alternatives in all cases. Moreover, compared with another OFDMA based MAC protocol (CTRMA) which is one of the most recent proposals, the signalling overhead is much less in SSMAP than the one in CTRMA.

Section 7.3 presents the simulation results for the comparison of SSMAP based routing protocol with other alternative routing protocols. Sections 7.3.2 and 7.3.3 compare the SSMAP based routing protocol with other alternatives in uniform node distribution topology and random node distribution topology respectively. The simulation results show that SSMAP based routing protocol has the highest throughput compared to other schemes in both topologies, especially when the number of sessions increases. The simulation results also show that SSMAP based routing protocol has less signalling overhead than other schemes in either uniform distribution topology or random distribution scenario. In sections 7.3.4 and 7.3.5, the proposed route maintenance scheme is evaluated in a mobile environment. The simulation results show that the route maintenance scheme can have good performance when the nodes have low moving speed. However, when the nodes's velocity increases up to 5 m/s or higher, the packets are largely dropped because of the frequent link breakage and co-channel interference. This is one of the issues that needs to be improved in the future work. In the end of this chapter, the proposed routing scheme is also tested in a large scale network with 1000 nodes.

Discussion and conclusion are presented in the next chapter.

Chapter 8

Conclusions

8.1   Discussion

This thesis develops a cross layer QoS routing protocols for OFDMA based ad hoc networks. The major contributions of this research can be divided into three parts. First of all a signal-strength based medium access protocol (SSMAP) is proposed to allocate sub-channels among users in OFDMA based ad hoc networks in a fully distributed way. Secondly, a SSMAP based QoS routing protocol is developed including the route discovery and route maintenance scheme. Finally, the implementation of OFDMA in ad hoc network needs more stringent time and frequency synchronization. Therefore, this thesis proposed a novel time and frequency synchronization scheme to realize concurrent transmissions in ad hoc networks by using OFDMA.

OFDMA has been widely used in several wireless standards. In recent years, more and more researchers consider implementing OFDMA in wireless ad hoc networks. One of the most important issues to use OFDMA in ad hoc networks is how to distribute the

sub-channels among the mobile nodes. Many publications propose different sub-channel allocation schemes, such as CTRMA [17]. However, these proposals need extra signalling messages periodically exchanging among nodes which will largely increase the system overall signalling overhead. SSMAP proposes an on-demand sub-channel allocation scheme which assigns sub-channels according to the current interference level of each sub-channel. The interference level for each sub-channel is recorded in a local *Free_channel* table. Each sub-channel has different priority. The sub-channel that has the least receiving interference level has the highest priority. By always assigning the sub-channels that have the least interference level the system throughput can largely increase. Moreover, each node in the network does not need to periodically exchange updating information with its neighbours which can largely reduce the signalling overhead.

SSMAP based QoS routing protocol has been proposed for OFDMA based ad hoc networks (section 4.4). The routing protocol consists of a route discovery process and a route maintenance scheme. Sub-channel allocation is encapsulated in the route discovery process. One separate sub-channel is used only for signalling through a separate transceiver. Once a node receives a route discovery message, it will check its local *Free_channel* table and assign the receiving sub-channels for the previous hop according to the interference level of each sub-channel. It will inform the transmitter the allocated transmission sub-channel through the route reply message (*RRES*).

Bandwidth and end to end delay requirements are considered in the route discovery process. The bandwidth aware routing protocol always calculates the bandwidth availability on each hop with a route request packet broadcast at the route discovery

stage. The discovered route must satisfy the bandwidth requirement. For the delay aware routing protocol, each node that receives the route discovery message will calculate the cumulative delay for the previous hops. It only forwards the route discovery message if the cumulative delay so far does not exceed the QoS delay requirement.

The route maintenance scheme is designed aiming to reduce the packet drop rate and increase the system throughput. Two signal strength thresholds are used in the route maintenance scheme to detect the link breakage and co-channel collisions: *low_threshold_trigger* and *high_interference_trigger*. If the signal strength of the receiving sub-channel used by a neighbouring node reduces to lower than the *low_threshold_trigger* and lasts for a pre-defined time span τ_1, a forthcoming route breakage is expected. A new route will be built before the real breakage of the current route to reduce the packet drop rate. The link can be repaired locally by *SREA/SRRP* messages when co-channel collision happens. The receiver will reallocate the transmitting sub-channel for the previous hop when it detects an inteferring node approaching and informs the previous hop node about the new allocated sub-channel through a *SREA* message. The simulation results show that the route maintenance scheme can have good performance in walking speed. However, when the speed dramatically increases, the current link may break before the new route is established which will result in many packets being dropped. This is even worse when the nodes' moving speed is more than 10 m/s. Therefore, future research is required to further improve the route maintenance scheme.

Implementing OFDMA in ad hoc networks needs restricted time and frequency

synchronization. Although there are many time and frequency synchronization schemes proposed in the literature for both OFDM and OFDMA, there is no publication to fully address the OFDMA synchronization problem in ad hoc networks. This thesis fully addresses the OFDMA synchronization for ad hoc, in order to support the proposed OFDMA based routing protocol, a time and frequency synchronization scheme is proposed in chapter 5. With the proposed time synchronization scheme, global time synchronization is no longer needed. Each node only needs to synchronize with one-hop neighbours. The basic idea behind the frequency synchronization scheme is using null sub-carriers inserted in each sub-channel. The frequency offset can be compensated by detecting and compensating the energy falling in the null sub-carriers. Bandpass filters are used in each node to separate the OFDMA packets from different transmitters so that a separate synchronization algorithm can be applied for each transmitter. The effect of number of bandpass filter in a node on the system throughput is also investigated. The simulation results show that due to the limitation on the number of total available sub-channels two bandpass filters are enough for sessions transmissions in ad hoc networks.

The proposed SSMAP and the QoS routing protocol have been fully evaluated through simulations. The simulation model is built using a commercial simulation tool OPNET. The network model, node model and process model are designed and presented in details in section 6.1. In order to prove the SSMAP based routing protocol is correctly implemented in the simulation model and the simulation results are correct, a full validation is presented in section 6.2 for both route discovery process and route maintenance scheme. The time synchronization scheme is explained step by step in the

validation as well.

The performance of the SSMAP based routing protocol has been evaluated in both uniform and random node distribution topologies. Firstly, the simulation results obtained from the SSMAP were compared with those of other multi-band based MAC protocol and other OFDMA based sub-channel allocation schemes. The network performance characteristics used for comparisons were the system throughput, signalling overhead, and session success rate. The simulation results show that the OFDMA based MAC protocols have better system performance in system throughput than other multi-band based MAC layer protocol, because the OFDMA can support concurrent transmissions. Compared with other OFDMA based MAC protocols such as CTRMA, SSMAP has much lower signalling overhead while the system throughput for SSMAP is still better than the one in CTRMA. For the SSMAP based routing protocol, the system has better performance than other routing schemes in a static ad hoc network. In a low speed mobile ad hoc network, the route maintenance scheme can successfully predict the forthcoming link breakage and rebuild the route before the link's real breakage. Therefore, the session success rate can still maintain a good performance. However, in a fast mobile ad hoc network, the system throughput and session success rate drop significantly with the increase in node speed. A mobile ad hoc network with fast moving speed suffers more from link breakage and co-channel collisions. A longer route is more easily affected by these problems. Therefore, a further study is required to improve the route maintenance scheme to reduce the packet drop in a high mobile ad hoc network.

8.2   Conclusion

Wireless mobile ad hoc networks have received more and more attention in recent years and it will become more important in the future communication networks. Many new standards have been created which can support ad hoc networks such as the IEEE 802.11x, Bluetooth and WiMAX. QoS routing support poses a great challenge in the ad hoc wireless network design because of the vulnerable nature of the wireless link and the frequent changes of the network topology. In this thesis, a novel signal strength based medium access protocol (SSMAP) is proposed with sub-channel allocation scheme in OFDMA based mobile ad hoc networks to realize concurrent transmission and reception. Based on SSMAP, a cross layer QoS routing protocol is proposed which is aiming to reduce the signalling overhead and co-channel interference. The novel proposed sub-channel allocation scheme and QoS routing protocol are evaluated through simulations using OPNET simulation tool.

Moreover, one of the greatest challenges to implement OFDM/OFDMA in ad hoc netoworks is its sensitivity to time and frequency offsets. Therefore, a proper synchronization scheme needs to be proposed to compensate these timing and frequency errors. In mobile ad hoc networks, synchronization is much more complex than in other infrastructure based wireless systems. In order to support the proposed SSMAP based routing protocol, a new timing and frequency synchronization algorithm is proposed in this thesis specifically designed for mobile ad hoc networks. The performance of the proposed synchronization scheme is evaluated through the simulations. The simulation results show that the proposed synchronization scheme can

achieve good performance in estimating and compensating the time and frequency errors.

Overall, this thesis presents a systematic research approach to address the challenges in developing a cross layer solution in OFDMA based mobile ad hoc networks. The proposed solutions are feasible and successful in achieving the aims of this research.

8.3   Future work

In this thesis, the route maintenance scheme is proposed in section 4.4.4. Although the proposed route maintenance scheme can predict the forthcoming link breakage and rebuild the route before the real link breakage in walking speed, the ability to fast rebuild the route in a high mobile ad hoc network needs to be further improved to reduce the packet loss and increase the session success rate.

The proposed time synchronization scheme is using preambles to realise one-hop coarse time synchronization and leave the rest small time offsets to be compensated by equalizers. A future research should take into account these time offsets and further refine the synchronization scheme. Moreover, the frequency synchronization scheme is evaluated only in one scenario in this thesis. In the future work, the synchronization algorithm can be investigated in other scenarios.

In this thesis, the SSMAP based routing proposed is aiming to maximize the overall system throughput and reduce the interference. User fairness is not taken into account in the proposed routing protocol. There is a tradeoff between the system throughput and user fairness. In the future work, a further research should focus on find a balance

between the system throughput and the user fairness. Moreover, admission control and

service degradation could be also taken into account in the proposed routing protocol.

Publications so far:

1. Hongyi Xiong, Eliane Bodanese, "A Scheme to Support Concurrent Transmissions in OFDMA based Ad Hoc networks", *IEEE 76th Vehicular Technology Conference: VTC2012-Fall,* 2012

2. Hongyi Xiong, Eliane Bodanese, "A Signal Strength Based Medium Access Control for OFDMA Based Wireless Ad Hoc Networks", *IEEE 18th International Conference on Telecommunications (ICT 2011)*, 2011

3. Hongyi Xiong, Eliane Bodanese, "A Signal Strength Based Cross Layer QoS Routing Protocol for OFDMA Based Wireless Ad Hoc Networks", NEWCOM++ / COST 2100, Joint Workshop on Wireless Communications, JNCW 2011.

References

[1] Ramin Hekmat, "Ad-hoc Networks: Fundamental Properties and Network Topologies", *Springer, first edition*, ISBN-13: 978-1402051654, November 14, 2006

[2] *http://www.wi-fi.org/Wi-Fi_Direct.php*

[3] Subir Kumar Sarkar, TG Basavaraju, C. Puttamadappa. (October 26, 2007). *Ad Hoc Mobile Wireless Networks: Principles, Protocols and Applications*. Auerbach Publications. (978-1420062212)

[4] Hao Yang, Haiyun Luo, etc. "Security in Mobile Ad hoc Networks: Challenges and Solutions", *Wireless Communications, IEEE, vol. 11, pp. 38- 47, Issue: 1,* Feb 2004

[5] Hao Yang, Haiyun Luo, etc. "Security in Mobile Ad hoc Networks: Challenges and Solutions", *Wireless Communications, IEEE, vol. 11, pp. 38- 47, Issue: 1,* Feb 2004

[6] Youngsik Lim, "The limit of ad hoc networks to commercial success", position paper, 2004.

[7] Paul, T.K., Ogunfunmi, T, "Wireless LAN Comes of Age: Understanding the IEEE 802.11n Amendment", *IEEE JNL*, *Circuits and Systems Magazine, IEEE, Volume 8, Issue 1*, Page(s):28 – 54, First Quarter 2008

[8] T. B. Reddy, I. Karthigeyan, B. Manoj, and C. S. R. Murthy, "Quality of service provisioning in ad hoc wireless networks: a survey of issues and solutions", available online: http://www.sciencedirect.com, Apr 2004

[9] Hanzo-II, L and Tafazolli, R. "A survey of QoS solutions for mobile ad hoc network", *Communications Surveys & Tutorials, IEEE*, Volume: 9, Issue: 2, pp. 50-70, Jul. 2007

[10] *IEEE 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 revision). IEEE-SA. 12 June 2007

[11] *Digital Video Broadcasting (DVB); Interaction channel for Digital Terrestrial Television (RCT) incorporating Multiple Access OFDM (MAC) and Physical Layer (PHY) specifications*, ETSI Std., 2002

[12] *IEEE Standard for Local and metropolitan area networks - Part 16: Air Interface for Fixed Broadband Wireless Access Systems*, IEEE Std., 2002

[13] E. Dahlman, S. Parkvall, J. Skold, P. Beming, "3G Evolution: HSPA and LTE for Mobile Broadband", Academic Press, July 31, 2007

[14] G. Kulkarni and M. Srivastava, "Subcarrier and bit allocation strategies for ofdma based wireless ad hoc networks," *GLOBECOM '02. IEEE, vol. 1, pp. 92–96 vol.*1, 2002

[15] J. Vishwanath Venkataraman; Shynk, "Adaptive algorithms for ofdma wireless ad hoc networks with multiple antennas," *Signals, Systems and Computers, 2004, vol. 1, pp. 110–114 Vol.1, 7-10* Nov. 2004

[16] Veyseh, M, Garcia-Luna-Aceves, J.J, "Parallel Interaction Medium Access for Wireless Ad Hoc Networks", *Computer Communications and Networks*, ICCCN '08, 2008

[17] Veyseh, M, Garcia-Luna-Aceves, J.J, Sadjadpour, H.R, "OFDMA Based Multiparty Medium Access Control in Wireless Ad Hoc Networks", *Communications ICC* 09, 2009

[18] L. Litwin, M. Pugel, "The principles of OFDM", *RF Design,* pp. 30-48, Jan.2001

[19] Ye (Geoffrey) Li and Gordon L. Stuber, "Orthogonal Frequency Division Multiplexing for Wireless Communications (Signals and Communication Technology)", Addison-Wesley, February, 2006

[20] Hui Liu, Guoqing, Li, "OFDM-Based Broadband Wireless Networks: Design and Optimization", *Wiley-Interscience; 1st edition,* November 4, 2005

[21] Loutfi Nuaymi, "WiMAX: Technology for Broadband Wireless Access", *Wiley, John & Sons, Incorporated*, March 2007

[22] Ferdous, H.S., Murshed, M. "Ad Hoc Operations of Enhanced IEEE 802.11 with Multiuser Dynamic OFDMA under Saturation Load", *IEEE Wireless Communications and Networking Conference (WCNC), Page(s): 309 – 314,* 2011

[23] Pomportes, S., Busson, A., Tomasik, J., Veque, V. "Resource Allocation in Ad Hoc Networks with Two-Hop Interference Resolution", *IEEE Global Telecommunications Conference (GLOBECOM 2011), Page(s): 1 – 6,* 2011

[24] Veyseh, M., Garcia-Luna-Aceves, J.J., Sadjadpour, H.R. "OFDMA Based Multiparty Medium Access Control in Wireless Ad Hoc Networks", *Proceedings of the 2009 International Communications Conference (ICC 2009), June 14-18,* 2009

[25] Royer, E.M. and Chai-Keong Toh. "A review of current routing protocols for ad hoc mobile wireless networks" *Personal Communications, IEEE*, Vol.6, pp. 46 – 55, April 1999

[26] Perkins, Charles E "Ad hoc networking", Addison-Wesley, ISBN: 0201309769, 2001

[27] Andrew S. Tanenbaum, "Computer Networks", 3rd edition, Prentice Hall, 1996

[28] C. Hedrick, " Routing Information Protocol", *Network Working Group*, RFC 1058, June 1988

[29] J. Moy, RFC 2328: OSPF version 2. Network Working Group. April 1998

[30] C. E. Perkins and P. Bhagwat. "Highly dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for mobile computers", *ACM Computer Communication Review*, Vol. 24, No.4, (ACM SIGCOMM'94), pp.234-244 Oct. 1994

[31] Murthy, S. and J.J. Garcia-Luna-Aceves, "An Efficient Routing Protocol for Wireless Networks*", ACM Mobile Networks and App. J., Special Issue on Routing in Mobile Communication Networks*, pp. 183-97, Oct. 1996

[32]   D. Johnson, D. A. Maltz, "Dynamic source routing in ad hoc wireless networks", *Mobile Computing* (T. Imielinski and H. Korth, eds.), Kluwer Acad. Publ., vol.353, pp. 153-181, 1996

[33]   C. E. Perkins and E. M. Royer, "Ad Hoc On-Demand Distance Vector Routing," Proceedings of IEEE Workshop on Mobile Computing Systems and Applications 1999, pp.90-100, February 1999

[34]   Z. J. Haas. "The Zone Routing Protocol (ZRP) for ad hoc networks", Internet Draft, Nov. 1997

[35]   M. Abolhasan, T. Wysocki, E. Dutkiewicz, "A review of routing protocols for mobile ad hoc networks", *Ad Hoc Networks*, Vol. 2, No. 1., pp. 1-22. January 2004

[36]   J. N. Al-Karaki and A. E. Kamal, "Quality of service routing in mobile ad hoc networks: Current and future trends", in *Mobile Computing Handbook* (I. Mahgoub and M. IIays, eds.), CRC Publishers, 2004

[37]   S. Chakrabarti and A. Mishra, "QoS issues in ad hoc wireless networks", *IEEE Commun. Mag.,* vol. 39, pp. 142-148, Feb. 2001

[38]   T. B. Reddy, I. Karthigeyan, B. Manoj, and C. S. R. Murthy, "Quality of service provisioning in ad hoc wireless networks: a survey of issues and solutions", available online: http://www.sciencedirect.com, Apr 2004

[39]   T.-W. Chen, J. T. Tsai, and M. Gerta, "QoS routing performance in multi-hop, multimedia, wireless networks", *in Proc. IEEE 6th Int. Conf. Universal Personal Communications*, vol. 2, pp. 557-561, Oct 1997

[40]   L. Xiao, "CDMA Bus Lane: A Cross-layer Protocol for QoS routing in CDMA based Mobile Ad Hoc Networks", PhD thesis, Queen Mary University of London, 2007

[41] Ming Chen, Guo, J. "A mobility based mobile Ad-hoc Network routing protocol", *IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, Page(s): 935 – 938, 2012

[42] L. Kleinrock and F. Tobagi, "Packet switching in radio channels part II: The hidden terminal problem in carrier sense multipleaccess modes and the busy-tone solution", *IEEE Trans. Commun.*, vol. 23, no. 12, pp. 1417.1433, 1975

[43] D. Shukla, L. Chandran-Wadia, and S. Iyer, "Mitigating the exposed node problem in IEEE 802.11 ad hoc networks", *In Proc. 12th Int. Conf. Computer Communications and Networks*, pp. 157.162, Oct. 2003

[44] El-Bazzal, Z., et al. "A Cross Layered Routing Protocol for Ad hoc Networks", *IEEE 2012 International Conference on Information Technology and e-Services,* pp. 1-4, 2012

[45] H. Shen, B. Shi, L. zou, and H. Gong, "A distributed entropybased long-life qos routing algorithm in ad hoc network". *in Proc. IEEE Canadian Conf. on Electrical and Computer Engineering*, vol. 3, pp. 1535.1538, May 2003

[46] T. Rappaport, "Wireless Communications: Principles and Practice," Prentice Hall, New Jersey, 1996

[47] Veyseh, M, Garcia-Luna-Aceves, J.J, "Parallel Interaction Medium Access for Wireless Ad Hoc Networks", *Computer Communications and Networks*, *ICCCN '08, Communications ICC 09, Page(s): 1 – 6,* 2008

[48] Veyseh, M, Garcia-Luna-Aceves, J.J, Sadjadpour, H.R, "OFDMA Based Multiparty Medium Access Control in Wireless Ad Hoc Networks", *Communications ICC 09, Page(s): 1 – 6*, 2009

[49]   Kaixin Xu, Gerla, M., Sang Bae, "How effective is the IEEE 802.11 RTS/CTS handshake in ad hoc networks", *Global Telecommunications Conference, 2002.* GLOBECOM '02. IEEE, 2002

[50]   L.J. Cimini, "Analysis and simulation of A Digital Mobile Channel Using Orthogonal Frequency Division Multiplexing," *IEEE Trans. Commun., Vol. COM-33,* pp. 665-675, July 1985

[51]   J.Víctor,A.Ana,"Multi-user Synchronisation in ad hoc OFDM-based Wireless Personal Area Networks", *Wireless Personal Communications, Volume 40, Number 3, 387-399*, DOI: 10.1007/s11277-006-9198-3, 2007

[52]   P. H. Moose, "A technique for orthogonal frequency division multiplexing frequency offset correction," IEEE Trans. Commun., vol. 42, no. 10, pp. 2908–2914, Oct. 1994

[53]   T. M. Schmidl and D. C. Cox, "Robust frequency and timing synchronization for OFDM," *IEEE Trans. Commun., vol. 45, no. 12, pp. 1613–1621*, Dec. 1997

[54]   U. Lambrette, M. Speth, and H. Meyr, "OFDM burst frequency synchronization by single carrier training data," *IEEE Commun. Lett., vol. 1, no. 2, pp. 46–48*, Mar. 1997

[55]   F. Tufvesson, O. Edfors, and M. Faulkner, "Time and frequency synchronization for OFDM using PN-sequence preamble," *in Proc. IEEE Vehicular Technology Conf.*, Sep. 1999, vol. 4, pp. 2202–2207

[56]   M. Morelli and U. Mengali, "An improved frequency offset estimator for OFDM applications," *IEEE Commun. Lett., vol. 3, no. 3, pp. 75–77*, Mar. 1999

[57]   B. Yang, K. B. Letaief, R. S. Cheng, and Z. Cao, "Timing recovery for OFDM transmission," *IEEE J. Select. Areas Commun., vol. 18, no. 11, pp. 2278–2291*, Nov. 2000

[58] H. K. Song, Y. H. You, J. H. Park, and Y. S. Cho, "Frequency-offset synchronization and channel estimation for OFDM-based transmission," *IEEE Commun. Lett., vol. 4, no. 3, pp. 95–97,* Mar. 2000

[59] A. J. Coulson, "Maximum likelihood synchronization for OFDM using a pilot symbol: Algorithms," *IEEE J. Select. Areas Commun., vol. 19, no. 12, pp. 2486–2494,* Dec. 2001

[60] A. J. Coulson, "Maximum likelihood synchronization for OFDM using a pilot symbol: Analysis," *IEEE J. Select. Areas Commun., vol. 19, no. 12, pp. 2495–2503*, Dec. 2001

[61] T. Keller, L. Piazzo, P. Mandarini, and L. Hanzo, "Orthogonal frequency division multiplex synchronization techniques for frequency-selective fading channels," *IEEE J. Select. Areas Commun., vol. 19, no. 6, pp. 999–1008*, Jun. 2001

[62] H. Minn, V. K. Bhargava, and K. B. Letaief, "A robust timing and frequency synchronization for OFDM systems," *IEEE Trans. Wireless Commun., vol. 2, no. 4, pp. 822–839*, Jul. 2003

[63] K. Shi and E. Serpedin, "Coarse frame and carrier synchronization of OFDM systems: A new metric and comparison," *IEEE Trans. Wireless Commun., vol. 3, no. 4, pp. 1271–1284,* Jul. 2004

[64] F. Classen and H. Meyr, "Frequency synchronization algorithms for OFDM systems suitable for communication over frequency selective fading channels," *in Proc. IEEE Vehicular Technology Conf.*, vol. 3, pp. 1655–1659, Aug. 1994

[65] F. Daffara and O. Adami, "A novel carrier recovery technique for orthogonal multicarrier systems," *Eur. Trans. Telecommun., vol. 7, pp. 323–334*, Jul./Aug. 1996

[66] J. J. van de Beek, M. Sandell, and P. O. Borjesson, "ML estimation of timing and frequency offset in OFDM systems," *IEEE Trans. Signal Proc., vol." 45, no. 7, pp. 1800–1805*, Jul. 1997

[67] N. Lashkarian and S. Kiaei, "Class of cyclic-based estimators for frequency-offset estimation of OFDM systems," *IEEE Trans. Commun., vol. 48, no. 12, pp. 2139–2149*, Dec. 2000

[68] J. Lei and T. Ng, "A consistent OFDM carrier frequency offset estimator based on distinctively spaced pilot tones," *IEEE Trans. Wireless Commun., vol. 3, no. 2, pp. 588–599,* Mar. 2004

[69] F. Daffara and A. Chouly, "Maximum likelihood frequency detectors for orthogonal multicarrier systems," *in Proc. ICC'93, pp. 766–771,* Jun. 1993

[70] M. Morelli, A. N. D'Andrea, and U. Mengali, "Feedback frequency synchronization for OFDM applications," *IEEE Commun. Lett., vol. 5, no. 1, pp. 28–30*, Jan. 2001

[71] T. M. Schmidl and D. C. Cox, "Blind synchronization for OFDM," *Elect. Lett., vol. 33, no. 2*, pp. 113–114, Feb. 1997

[72] H. Liu and U. Tureli, "A high-efficiency carrier estimator for OFDM communications," *IEEE Commun. Lett., vol. 2, no. 4, pp. 104–106*, Apr. 1998

[73] U. Tureli, H. Liu, and M. D. Zoltowski, "OFDM blind carrier offset estimation: ESPRIT," *IEEE Trans. Commun., vol. 48, no. 9, pp. 1459–1461,* Sep. 2000

[74] G. Santella, "A frequency and symbol synchronization system for OFDM signals: Architecture and simulation results," *IEEE Trans. Veh. Technol., vol. 49, no. 1, pp. 254–275,* Jan. 2000

[75] X. Ma, C. Tepedelenlioglu, G. B. Giannakis, and S. Barbarossa, "Non-data-aided carrier offset estimator for OFDM with null subcarriers," *IEEE J. Select. Areas Commun., vol. 19, no. 12, pp. 2504–2515,* Dec. 2001

[76]   M. Ghogho, A. Swami, and G. B. Giannakis, "Optimized null-subcarrier selection for CFO estimation in OFDM over frequency selective fading channels," *in Proc. IEEE Globecom* 2001, Nov. 2001, vol. 48, pp. 202–206

[77]   H. Bolcskei, "Blind estimation of symbol timing and carrier frequency offset in wireless OFDM systems," *IEEE Trans. Commun., vol. 48, no. 6, pp. 988–999*, Jun. 2001

[78]   U. Tureli, D. Kivanc, and H. Liu, "Experimental and analytical studies on high-resolution OFDM carrier frequency offset estimator," *IEEE Trans. Veh. Technol., vol. 50, pp. 629–643*, Mar. 2001

[79]   M. Ghogho and A. Swami, "Semi-blind frequency offset synchronization for OFDM," in *Proc. ICASSP 2002*, vol. 3, pp. 2333–2336, May 2002

[80]   M. Ghogho and A. Swami, "Blind frequency-offset estimator for OFDM systems transmitting constant-modulus symbols," *IEEE Commun. Lett., vol. 6, no. 8, pp. 343–345,* Aug. 2002

[81]   F. Yang, K. H. Li, and K. C. Teh, "A carrier frequency offset estimator with minimum output variance for OFDM systems," *IEEE Commun. Lett., vol. 8, no. 11, pp. 677–679*, Nov. 2004

[82]   H. Nogami and T. Nagashima, "A frequency and timing period acquisition technique for OFDM systems," *in Proc. Personal, Indoor and Mobile Radio Communications (PIMRC)*, vol. 3, pp. 1010–1015. Sep. 1995

[83]   Morelli. M, Kuo. C.-C.J, Pun. M.-O, "Synchronization Techniques for Orthogonal Frequency Division Multiple Access (OFDMA): A Tutorial Review", *Proceedings of the IEEE Volume: 95 , Issue: 7 , Page(s): 1394 – 1427*, 2007

[84]   J. J. van de Beek, P. O. Bo ̈rjesson, M. L. Boucheret, D. Landstro ̈m, J. M. Arenas, O. O ̈dling, C. O ̈stberg, M. Wahlqvist, and S. K. Wilson, "A time and frequency

synchronization scheme for multiuser OFDM," *IEEE J. Select. Areas Commun., vol. 17, no. 11, pp. 1900–1914*, Nov. 1999

[85]   S. Barbarossa, M. Pompili, and G. B. Giannakis, "Channel-independent synchronization of orthogonal frequency division multiple access systems," *IEEE J. Select. Areas Commun., vol. 20, no. 2, pp. 474–486*, Feb. 2002

[86]   Gul, Malik Muhammad Usman; Lee, Sungeun; Ma, Xiaoli; "Null sub-carrier based carrier frequency offset estimation for OFDMA uplink systems", *Digital Signal Processing Workshop and IEEE Signal Processing Education Workshop (DSP/SPE)*, 2011

[87]   Pengfei Sun; Morelli, M.; Li Zhang, "Carrier Frequency Offset Tracking in the IEEE 802.16e OFDMA Uplink", *Wireless Communications, IEEE Transactions on Volume: 9*, 2010

[88]   M. Morelli, "Timing and frequency synchronization for the uplink of an OFDMA system," *IEEE Trans. Commun., vol. 52, no. 2, pp. 296–306*, Feb. 2004

[89]   M. O. Pun, M. Morelli, and C.-C. J. Kuo, "Maximum-likelihood synchronization and channel estimation for OFDMA uplink transmissions," *IEEE Trans. Commun., vol. 54, no. 4, pp. 726–736*, Apr. 2006

[90]   M. O. Pun, M. Morelli, and C.-C. J. Kuo, "Iterative detection and frequency synchronization for OFDMA uplink transmissions," *IEEE Trans. Wireless Commun., vol. 6, no. 2, pp. 629–639*, Feb. 2007

[91]   Y. Na, H. Minn, "Line search based iterative joint estimation of channels and frequency offsts for uplink OFDMA systems," *IEEE Trans. Wireless Commun., nol. 6, no. 12, pp. 4374-4382*, Dec. 2007

[92] Sanguinetti. L, Morelli. M, "A Low-Complexity Scheme for Frequency Estimation in Uplink OFDMA Systems", *Wireless Communications, IEEE Transactions on Volume: 9,* Issue: 8, 2010

[93] J. Choi, C. Lee, H. W. Jung, and Y. H. Lee, "Carrier frequency offset compensation for uplink of OFDM-FDMA systems," *IEEE Communication Letter*, vol. 4, no. 12, pp. 414–416, Dec. 2000

[94] D. Huang and K. B. Letaief, "An interferencecancellation scheme for carrier frequency offsets correction in OFDMA systems," *IEEE Trans. Commun., vol. 53, no. 7, pp. 1155–1165,* Jul. 2005

[95] S. Verdu, "Multiuser Detection", Cambridge, U.K, Cambridge Univ. Press, 1998

[96] P. Moose. "A technique for orthogonal frequency-division multiplexing frequency offset correction", *IEEE Trans. Commun*, 42(10): 2908-2914, Oct. 1994

[97] Martha Suarez, Martine Villegas and Genevieve Baudoin (2010). "RF and Microwave Band-Pass Passive Filters for Mobile Transceivers with a Focus on BAW" Technology, Advanced Microwave and Millimeter Wave Technologies Semiconductor Devices Circuits and Systems, Moumita Mukherjee (Ed.), ISBN: 978-953-307-031-5, InTech, Available from: http://www.intechopen.com/articles/show/title/rf-and-microwave-band-pass-passive-filters-for-mobile-transceivers-with-a-focus-on-baw-technology

[98] Pitschi, M.; Kiwitt, J. **"**Design of Duplexer "Inserts" for Mobile Phone Module Application", *Microwave Symposium Digest IEEE MTT-S International, Page(s): 1277 – 1280*, 2006

[99] Andrea Goldsmith, "Wireless Communications", Cambridge University Press, August 8, 2005

[100] K. Shi and E. Serpedin, "Coarse time and carrier synchronization of OFDM systems: A new metric and comparison," *IEEE Trans. Wireless Commun.*, vol. 3, no. 4, pp. 1271–1284, Jul. 2004

[101] Hiroshi Nogami, Toshio Nagashima, "A Requency and Timing Period Acquisition Technique for OFDM Systems", *IEICE Trans. Commun.*, Vol. E79-B, No. 8, pp. 1135-1146, Aug. 1996

[102] J. So and N. H. Vaidya, "Multi-channel mac for ad hoc networks: handling multichannel hidden terminals using a single transceiver," *in MobiHoc '04, 2004*, pp. 222–233

[103] E. Shim, S. Baek, J. Kim, and D. Kim, "Multi-channel multi-interface mac protocol in wireless ad hoc networks," *Communications, 2008. ICC '08. IEEE International Conference on, pp. 2448–2453*, May 2008

[104] Karthikeyan Bhargavan, Carl A. Gunter, Moonjoo Kim, Insup Lee, Davor Obradovic, Oleg Sokolsky, and Mahesh Viswanathan, "Verisim: Formal Analysis of Network Simulations", *IEEE Transaction on Software Engineering, Vol.28, pp. 129-145,* Feb 2002

[105] T. Clausen, P. Jacquet, A. Laouiti, P. Muhlethaler, a. Qayyum et L. Viennot, "Optimized Link State Routing Protocol", *IEEE INMIC Pakistan, pp.62-68,* 2001

[106] Jeff Feigin, "Practical Costas loop design" , *RF Design: pp. 20–36.* January 1, 2002

[107] Schlesinger, et al. "Terminology for Model Credibility", *Simulation, pp. 103–104.* 1979