私立東海大學資訊工程與科學研究所

碩士論文

指導教授：朱延平

行動隨意網路下隱藏接收節點問題之研究
A Study on Hidden Receiver Problem in Mobile Ad Hoc Networks

研 究 生：羅群智

中華民國九十七年六月

誌 謝

本論文得以順利完成，首先要感謝的便是我的指導教授 朱延平教授。當初剛從南非回台灣讀書的我，在什麼都不懂的情形下，來到東海大學念研究所，老師不計背景的好壞，悉心的指導，讓我在學業上與研究上有許多收穫，並且在論文上老師給予相當多的建議與幫助，讓我能順利完成碩士學位，同時，也謝謝老師在生活上的關懷與經驗上的分享，使我於各方面皆有所成長，讓我受益良多。此外，也要感謝黃一泓學長與林冠成學長，指導我論文方向，並且在相關研究的問題探討上，給予我建議與協助。

另外，要感謝實驗室的學長、學姐、學弟與同學們，感謝各位陪我走過這段研究所的時光，特別是郭昭佑學長，在和他討論論文上，學到凡事要細心求證，並對我在做論文與申請博班期間，時時打電話麻煩他，感到歉意，在此由衷謝謝他。接著感謝黃卉娟學姐，我碩一剛進實驗室時，有許多地方得到她的照顧，非常謝謝她。再來要感謝二年來相處的同學－王祥明同學、邱益仁同學與學弟呂志銘，在我的研究過程中給我寶貴的意見，幫助我解決課業與生活上所遭遇到的問題，你們是實驗室的開心果，有你們使得實驗室處處充滿了歡樂。

最後，我要感謝的是我最愛的父母與姐姐們，他們辛苦的工作來栽培我，使我在物質生活上不虞匱乏，並在我最困難無助的時候，鼓勵我、陪伴我，讓我能順利的完成學業，他們功不可沒，感謝你們對我的付出與關心，雖然父親最近因事故過世, 但他對我從小到大的教導我會深深記住, 也很感謝從小到大不辭辛苦的把我送出國讀書與栽培，我會努力做到最好。

中文摘要

媒體存取協定（Medium Access Control, MAC）在 IEEE 802.11 的標準中，一直都是無線區域網路(Wireless Local Area Networks, WLAN）與無線行動隨意網路（Mobile Ad Hoc Networks, MANET）的標準。所謂的 Hidden Receiver 的問題是由於高度忽視隱藏節點的存在而讓潛在的傳送者（隱藏節點的鄰居）進入了一個不必要的退避過程(Backoff）和重傳循環，所以導致競爭者之間嚴重的不公平競爭並增加了封包碰撞(Collision)與干擾(Interference)的可能性。因此，大大的降低網路吞吐(Throughput)的表現，這也讓 Hidden Receiver 成為無線行動隨意網路最值得探討的議題之一。

在這一篇論文裡，我們探討了在無線網路環境下所存在的相關問題，並且提出了兩個可以應用在 IEEE 802.11 Distributed Coordination Function (DCF)與 Explicit Blocking Notification (EBN)的改善方法。

模擬結果顯示我們所提出的方法不但可以有效降低封包碰撞的機率，且可減少網路媒介的競爭以及有效的提升網路公平性，更可以大大的改善網路整體的吞吐量(System Throughput)與品質的服務(QoS)。

關鍵詞： 隨意網路（Ad Hoc Network), 隱藏節點問題（Hidden Terminal Problem), 二元指數倒退法（Binary exponential backoff）

Abstract

In IEEE 802.11 Medium Access Control (MAC) protocol has been the standard for the Wireless Local Area Networks (WLAN) and Mobile Ad Hoc Networks (MANET). However, the hidden receiver problem is one of the primary concerns in wireless ad hoc networks, due to the high rates of neglecting the state of the hidden terminal; the potential sender (neighbors of the hidden terminal) enters into an unnecessary backoff process and retransmission circle, therefore leading to serious unfairness between contention terminals.

Hidden receiver problem increases the probability of collision and interference, hence, degrading the performance of network throughput significantly. In this thesis, we review the related problems in wireless environments and proposed two improved schemes with an additional procedure which can be adapted to IEEE 802.11 Distributed Coordination Function (DCF) and Explicit Blocking Notification (EBN) schemes.

Our simulation results have shown that our proposed schemes not only achieve better fairness between competing traffics, reducing the probability of collisions and decreasing the contention on medium access but also greatly improve system throughput and the Quality of Service (QoS) of networks.

Keywords **：** *Ad Hoc Network, Hidden Terminal Problem, Binary exponential backoff*

LIST OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1
Introduction

1.1 Overview

IEEE 802.11 defines wireless LAN Medium Access Control (MAC) and physical layer (PHY) specifications [1]. The architecture of the MAC sub-layer includes the Distributed Coordination Function (DCF), the Point Coordination Function (PCF) and their coexistence. The fundamental access method of the IEEE 802.11 MAC is the DCF, also known as Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) and PCF is only usable on infrastructure network configurations thus PCF will not be discussed in this thesis.

Mobile Ad Hoc Wireless Networks (MANET) is a collection of wireless mobile nodes that does not need any centralized control. MANET also self-configures to form a network without the aid of any pre-existing infrastructure. Mobile nodes can communicate with each other directly or via other nodes. A mobile node can not receive data from two different sources simultaneously, which may cause collisions at the receiver. Since the nodes are mobile, the network topology may change rapidly and unpredictably at any time. Example applications of mobile ad hoc networks include emergency rescue, natural disaster relief, information exchange of the enterprises or military battlefield communication, etc.

The performance of a Wireless Local Area Network (WLAN) heavily depends on its medium access control scheme [11]. The IEEE 802.11

WLAN protocol uses a medium access control mechanism based on the Carrier Sense Multiple Access (CSMA) protocol [7]. In CSMA, a node is allowed to transmit only if it determines the medium to be idle. However, CSMA cannot prevent packet collisions when two senders are not in range of one another but both transmit data to a common receiver. In this case attempting to detect if the medium is free does not necessarily work because the two senders can not detect one another's transmissions. Thus the packets from the two senders will collide at the common receiver. We refer this as hidden terminal problem [13].

In order to solve hidden terminal problem and have fair medium accesses, there are several existing MAC protocol schemes that use the mechanism of channel sensing or packet sensing to avoid collision. The sensing mechanisms typically rely on the transmitter and receiver performing a handshake prior to the transmission of the data packet, such as the Medium Access Collision Avoidance (MACA) [5] and the Media Access Protocol for Wireless LAN's (MACAW) [2].

1.2 Motivation

In IEEE 802.11 Mac protocol has been the standard for the Wireless Local Area Networks and Mobile Ad Hoc Networks. However, hidden receiver problem [8] is the primary concern in wireless ad hoc networks due to the high rates of neglecting the state of the hidden terminal. The simulation results of [6] and [8] have clearly shown that hidden terminal and receiver has a significant impact on network performance, especially on maximum achievable throughput and delay at moderate loads, since the presence of hidden terminal and hidden receiver induces increased number of retransmissions due to collisions and proves that the hidden terminal and receiver problems strongly affects the channel utilization.

1.3 Contributions

In this thesis, we review the problems and drawbacks in the ad hoc networks environment. We explain the disadvantages of RTS/CTS solution and unfairness of binary exponential backoff algorithm, and proposed two improved scheme aiming to effectively avoid blindly backoff and unnecessary RTS retransmissions in hidden terminal and receiver problem.

In our proposed schemes, we introduced an additional procedure which can be adapted to IEEE 802.11 Distributed Coordination Function (DCF) and Explicit Blocking Notification (EBN) schemes to achieve the objectives of our thesis. We modify the contention mechanism in such way that the transmission probabilities are fair between competing traffics and reduce the collisions of RTS packets. These two schemes greatly improved the fairness between contention traffics and system throughput as well as effectively avoid blindly backoffs in ad hoc networks.

1.4 Thesis Organization

The rest of the thesis is organized as follows. In Chapter 2, we describe related work of IEEE 802.11 DCF (Basic Access Mechanism, Binary Exponential Backoff algorithm, RTS/CTS Mechanism) and hidden terminal problem along with its solution as well as the problem with RTS/CTS which causes the hidden receiver problem and also the EBN scheme briefly. In Chapter 3, we describe the design overview and the details of our proposed scheme. In Chapter 4, we carry out several simulations to study and validate our proposed scheme in different scenarios. Finally, we conclude the thesis in Chapter 5.

Chapter 2
Related Work

This chapter provides some background information on IEEE 802.11 DCF and discusses some of the important problems caused by RTS/CTS mechanism, as well as the possible solutions introduced by these problems.

2.1 IEEE 802.11 DCF

In the IEEE 802.11 protocol, the fundamental medium access mechanism is referred to as Distributed Coordination Function (DCF), which is a random access protocol based on carrier sense multiple access with collision avoidance (CSMA/CA). DCF specifies two access mechanisms, namely the default basic access mechanism and an optional RTS/CTS mechanism.

2.1.1 Basic Access Mechanism

The basic idea of CSMA is to reserve the channel for the source of a certain ongoing transmission by carrier sensing. Any station wishing to transmit must sense the medium first. If some other nodes are already transmitting, the node sets a random timer and then waits for this period of time to try again. On the other hand, if the medium is currently idle, the node begins its transmission. However, the simple CSMA mechanism is susceptible to the hidden terminal, especially in wireless ad hoc networks.

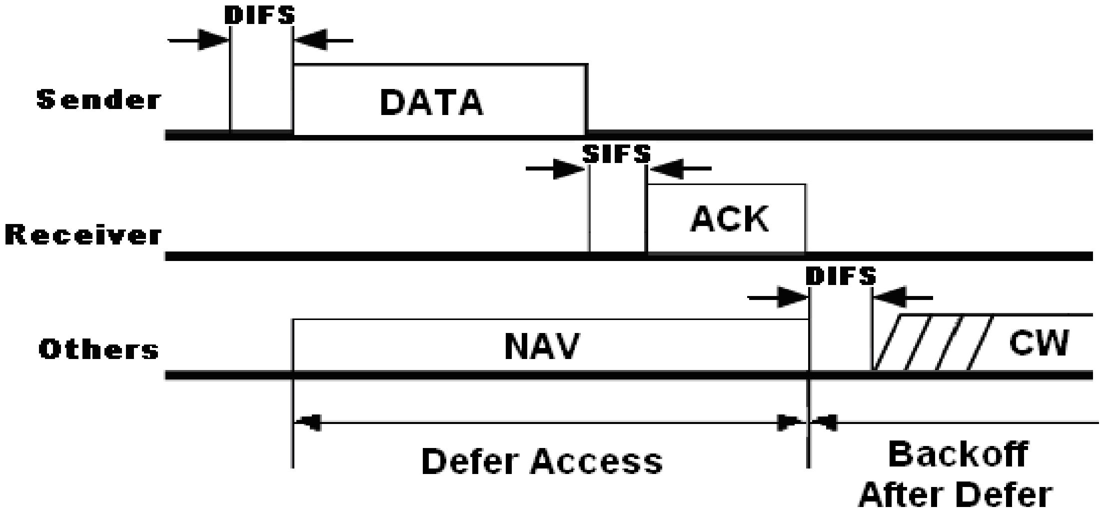


Figure 1: Basic Access Mechanism

In Figure 1, when sender wants to transmit data packet to receiver, it has to senses the medium first, if the medium is busy (i.e. some other terminal is busy transmitting) then the sender have to defer its transmission to a later time, if the medium is sensed idle then the sender is allowed to transmit. Each terminal contains a Network Allocation Vector (NAV) that predicts when the ongoing transmission on the medium will be completed. This is based upon on the information of the Duration/ID field of the received frames. The NAV is similar to a counter that decrements to zero. When the counter reaches to zero, then it indicates that the medium is idle.

These kinds of protocols are very effective when the medium is not heavily loaded, since it allows terminals to transmit when the delay is minimum, but there is always a chance of terminals transmitting at the same time therefore causing collisions. These collision situations must be identified, so the MAC layer can retransmit the packet by itself and not by upper layers, which would cause significant delay.

2.1.2 RTS/CTS Mechanism

The Request-to-Send and Clear-to-Send (RTS/CTS) mechanism is an optional handshaking procedure used by the IEEE 802.11 wireless ad hoc network to reduce the possibility of collision. The RTS/CTS mechanism is used to control terminals from accessing to the medium when collisions occur.

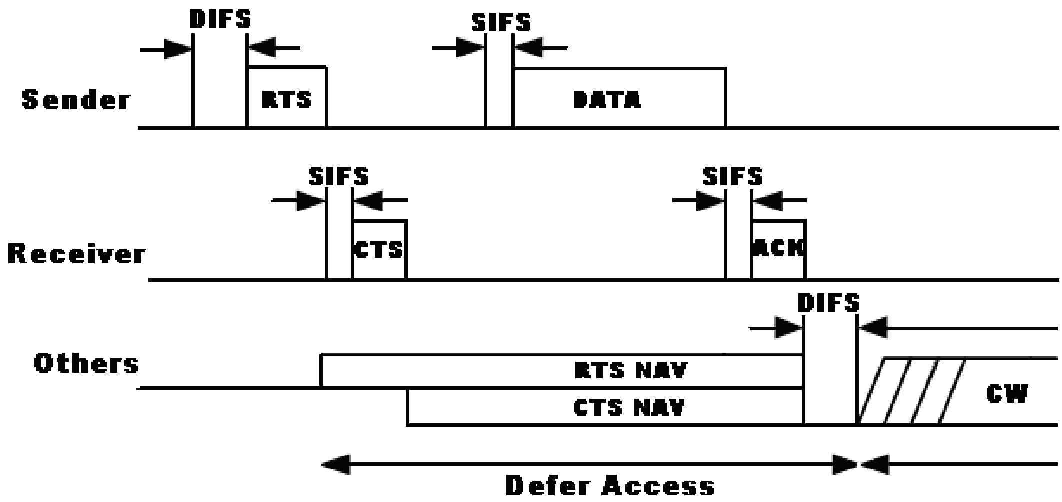


Figure 2: RTS/CTS Mechanism

In Figure 2, it shows how the four-way hand-shaking exchanges between a sender and receiver. It also indicates the NAV duration recorded in the RTS and CTS frames. The carrier sense mechanism combines the NAV state and the terminal's transmitter status with physical carrier sense to determine the busy/idle state of the medium. When the NAV duration expires, the virtual carrier sense mechanism indicates that the medium is idle; otherwise, it indicates that the medium is busy.

2.1.3 Binary Exponential Backoff algorithm

In IEEE 802.11 MAC protocol, a simple and effective random backoff algorithm is widely used to avoid collisions when more than one terminal tries to access the channel [9]. Only one of the terminals is granted access to the channel, while other contending terminals are suspended into a backoff state [3]. In particular, the binary exponential backoff algorithm adjusts the contention window size dynamically in react of collision intensity.

A terminal determines that the medium is idle through the use of the carrier sense function for a specified interval. It means that the terminal shall ensure that the medium keeps idle for an inter-frame space (IFS), and then it decides if the channel is really idle and if it has the right to send packets.

The IEEE 802.11 standard defines four types of IFS, which are Short IFS (SIFS), PCF IFS (PIFS), DCF IFS (DIFS), and Extended IFS (EIFS). The time relevance among those IFS are SIFS < PIFS < DIFS < EIFS.

- SIFS: It is the shortest out of all inter-frame spaces with the highest priority to access the communication medium. The SIFS is used for RTS, CTS and the ACK frames after SIFS interval seize the communication medium. The SIFS times includes processing delay and receive/transmit turnaround time.
- PIFS: PIFS is used by the PCF to gain access to the communication medium for transmitting data frames. This interval gives PCF a higher priority than DCF.

- DIFS: DIFS is used by the DCF to gain access to the communication medium for transmitting data frames. In the RTS/CTS access mechanism, the terminal should wait for DIFS before transmitting RTS frame.
- EIFS: EIFS is be used by the DCF whenever the last transmission is not successful.

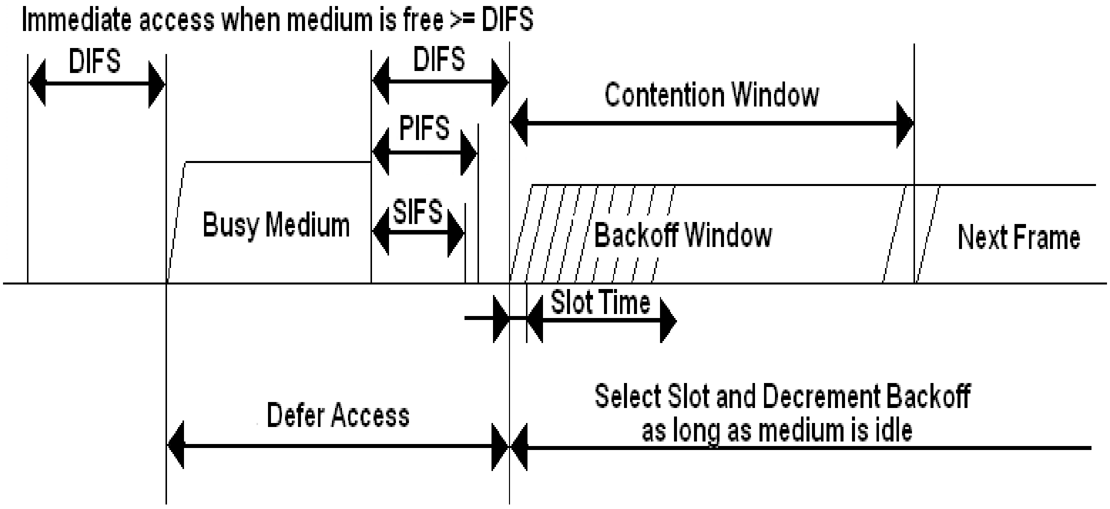


Figure 3: The Backoff Procedure

In Figure 3, when a node has a packet to send in the network, before attempting to transmit data, it first senses the channel using carrier sensing technique to determine whether it is idle and not being used by any other node. If the medium is sensed idle throughout a DIFS, then node is granted access to transmit. If no medium activity is indicated for the duration of a particular backoff slot, then the backoff procedure shall decrement its backoff time by a SlotTime. If the medium is determined to be busy at any time during a backoff slot, then the transmission wait for a SIFS and backoff procedure is suspended; that is the backoff time shall not be decremented for that slot. When the medium is idle again for the duration of a DIFS or EIFS, the backoff time will start counting down. Transmission shall commence whenever the backoff timer reaches zero. The effect of this procedure is that when multiple nodes are deferring and goes into a random backoff, the node selecting the smallest backoff time using the random function will win the contention.

A random backoff time will be uniformly chosen in the range of [0, CW - 1] and used to initialize the backoff timer, where CW is the current contention window size.

The following equation is used to calculate the backoff time:

$$BackoffTime = (Random(\ ) \times CW) \cdot aSlotTime \quad (1)$$

After calculating the backoff time, the backoff procedure will be preformed by deferring the node for that time period. Using carrier sense mechanism, the activity of the medium is sensed at every time slot. If the medium is found to be idle then the backoff time period is decremented by one time slot.

$$NewBackoffTime = OldBackoffTime - aSlotTime \quad (2)$$

The backoff timer keeps running as long as the channel is sensed idle, when the medium is busy during backoff time period, then the backoff timer is paused, and resumed when the channel is sensed idle again for more than DIFS. When the backoff timer expires, the node will attempt to transmit a data frame at the beginning of next slot. As illustrated in Figure 4, if the node successfully received the data packet, the receiver transmits an acknowledgment for that packet, and then the *CW* for this node is reset to the minimum. When the transfer fails, the node goes into another backoff.

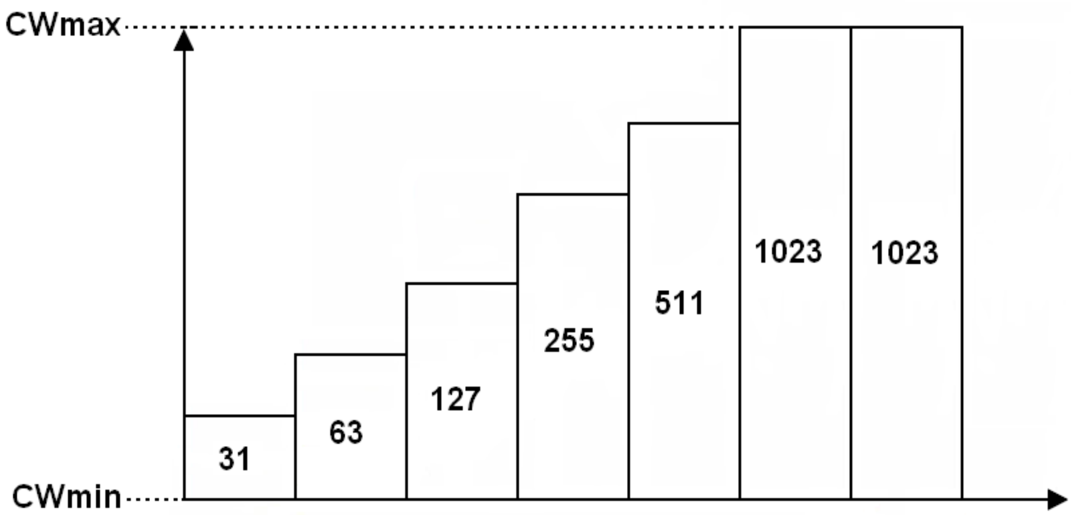


Figure 4: An Example of Exponential Increase of *CW*

2.2 Problems and Solutions

2.2.1 Hidden Terminal Problem

Hidden terminal occurs when a node is within the range of the intended receiver but out of range of the transmitter. Transmission ranges of each node are illustrated in Figure 5. Node B is within the communication range of node A and node C, but node A and node C is not in each other’s communication range which means that node C is a hidden terminal.

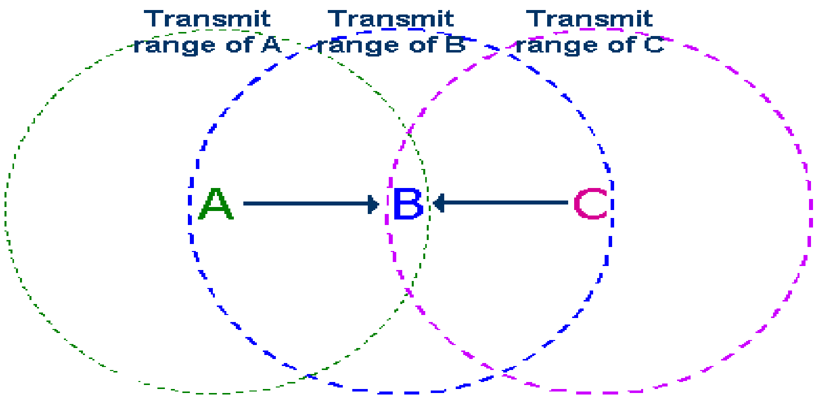


Figure 5: Hidden Terminal Problem

Now consider the case where node A is transmitting to node B and node C wants to transmit to node B at the same time, but node C being out of communication range of node A, so node C cannot detect that node A is busy transmitting to node B, therefore when node C sends data to node B, it may cause collisions at node B and data received by node B may be incorrect. The hidden node C therefore needs to defer its transmission while node A transmits a packet to node B.

This is referred to as the hidden terminal problem (hidden node is a sender), as node A and node C are hidden from each other. This can be resolved by RTS/CTS solution shown in Figure 6.

2.2.2 RTS/CTS Solution

MACA implements two short control packets called Request-to-Send (RTS) and Clear-to-Send (CTS) to achieve the medium reservation.

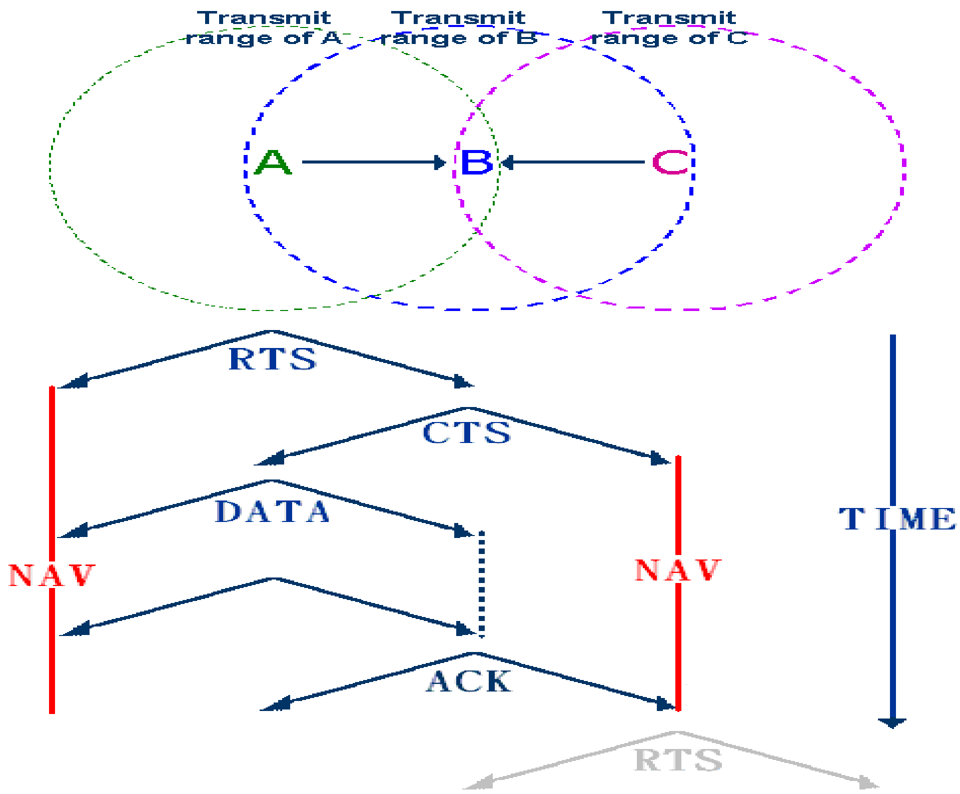


Figure 6: RTS/CTS Solution for Hidden Terminal Problem

As illustrated in Figure 6, when node A wants to transmit data to node B, node A initially sends a RTS packet to node B. Upon correctly receiving this RTS packet, node B responds with a CTS packet to indicate the acceptance of communication. After receiving this CTS packet, node A knows that the channel is reserved and then begins the transmission of data to node B (If any other node overhears the RTS or CTS packets at this period of time, it will defer its own transmission for the time period indicated by RTS/CTS packets). After the transmission of data, node A waits for a control packet called Acknowledgement (ACK) from the receiver to indicate it has correctly received the data.

This example clearly shows how the hidden terminal problem is eliminated by RTS/CTS solution, and thus collisions are avoided. Via this control packet exchanging process, all the hidden terminals will not transmit data during the period of data transmission, and the effect of hidden node problem is reduced to minimal.

2.2.3 Hidden Receiver Problem

RTS/CTS protocol was originally introduced to solve the hidden terminal problem. However, RTS/CTS does not always solve hidden terminal problem completely and sometimes can be considerably inefficient due to false blocking, which causes network resource wastage and unfairness between contention traffics.

In the RTS/CTS exchanging mechanism, any node receiving either an RTS or CTS packet will be blocked for a certain period of time to ensure not to interfere with ongoing transmissions. Since nodes in an ad hoc network share a single transmission channel, only one node is allowed to transmit at any time within the range of a receiver, and all the other nodes may be blocked. As for the neighbors of a blocked node, these nodes will not be aware of the fact that this node is blocked. Therefore, communication with the blocked node may still be initiated by its neighbors. In this situation, the sender sends out an RTS packet and waits for response. However, the blocked destination will not respond to this RTS packet. Since the sender does not get any response to its RTS packet, it enters into binary exponential backoff mode. Furthermore, this RTS packet forces every other node that receives it to inhibit any transmission even though the blocked destination does not respond. Without a CTS response, data transmission will not be ignited. It is a waste of medium that stations been inhibited from transmitting while no data transmission is actually takes place.

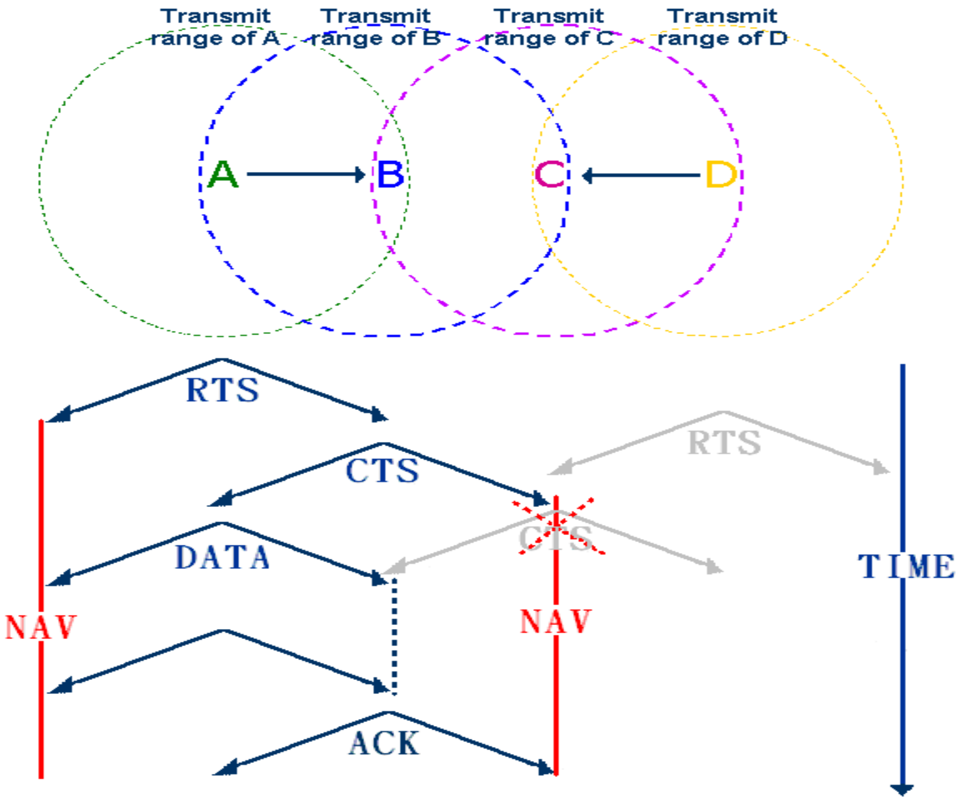


Figure 7: Hidden Receiver Problem

In Figure 7, clarifies such situation. Suppose that during an ongoing transmission from node A to node B, node D wants to initiate a transmission to node C. Upon receiving RTS packet from node D, node C will not be allowed to reply with a CTS packet because of the CTS packet that was previously heard from node B, although it is not exposed to any ongoing transmission in its vicinity. We refer to this consequence of the inhibitory nature of RTS/CTS as Hidden Receiver Problem (the hidden node acts as a receiver).

Node D doesn't know exactly what indeed caused the control packet exchange to fail, it has to enter into binary exponential backoff and reinitiates RTS/CTS handshake later. As long as the hidden node is still deferring, the handshake will not succeed, which results in consecutive unnecessary backoff and RTS retransmission.

Moreover, in each backoff, contention window size is doubled. Meanwhile, node A transmits the packet successfully and is not aware of the collision at node C, when transmitting the next packet, node A uses the minimum contention window size. The binary exponential backoff algorithm tend to favors the last succeeding node and the chance of collision recognized by node D can not be reduced even though it backoff before next retry. The hidden receiver problem increases the chance of multiple retries, making the wrong declaration of link failure and therefore rerouting instability more likely. Above mentioned problem will not only affect hidden terminal and receiver; it will also affect all other nodes that are in the range of last succeeding node.

2.2.4 Explicit Blocking Notification (EBN) scheme

Dong *et al.* [8] proposed a solution to address the hidden receiver problem, the EBN scheme. He introduces an additional phase in traditional RTS/CTS based scheme which is called Blocking Notification (BN).

The main purpose of EBN scheme is that the hidden node may explicitly notify its neighbors (potential senders) in advance regarding its current state of deference. Once the status of hidden node is obtained, then more effective decisions can be made by the potential senders to avoid blindly backoffs and retransmissions and his experiments results shown that EBN scheme achieves better fairness between competing traffics.

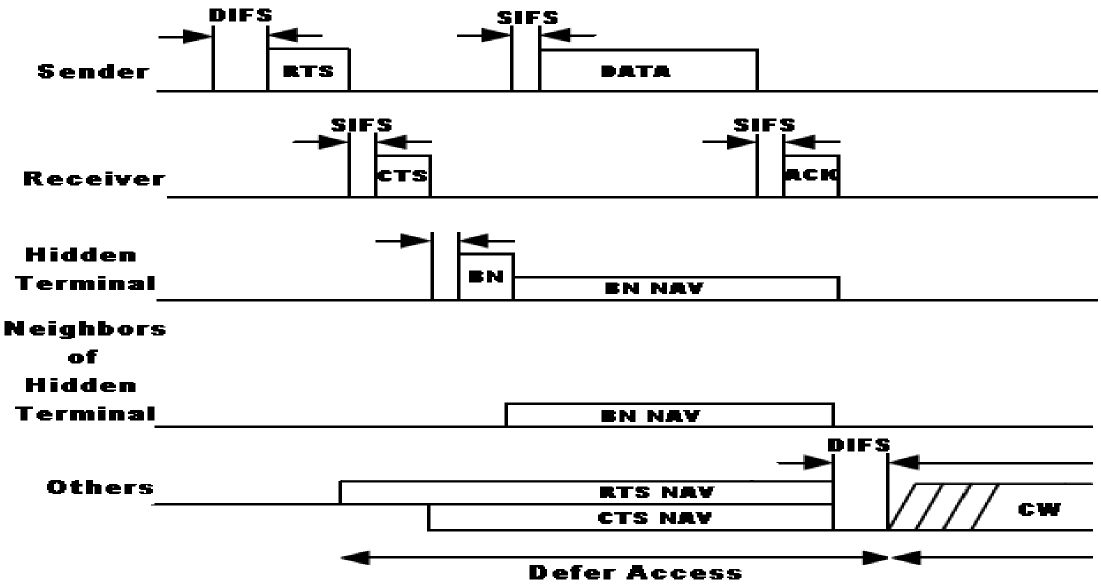


Figure 8: Access mode of EBN Scheme

In Figure 8, it shows the handshake sequence of EBN scheme (RTS/CTS/BN/DATA/ACK) between a sender and receiver. It also indicates the NAV duration recorded in the RTS, CTS and BN frames, and how BN frame explicitly notify its neighbors (potential senders) in advance regarding its current state of deference.

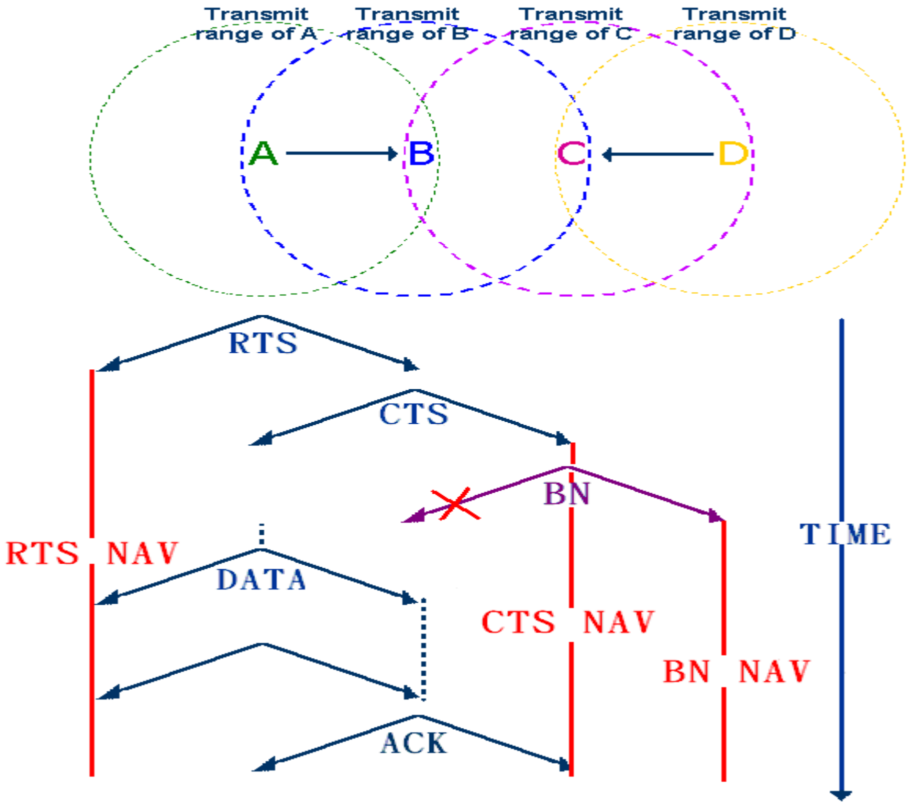


Figure 9: EBN Scheme

As illustrated in Figure 9, when sender sends RTS to its receiver, the receiver replies with a CTS packet, but when other node receives this CTS packet that is not destined for it, it then indicates that that node is a hidden terminal, and then it broadcasts a BN packet to the neighboring nodes, which may be potential senders, announcing the forthcoming data transmission. The neighboring nodes and the potential senders then extract the information from BN packet then know that the hidden terminal is currently deferring. (When the data receiver receives this BN packet it will just discards this BN packet silently and prepares for data reception.)

When a potential sender tries to communicate with a hidden neighbor at that period of time, it will try not to send RTS packet until the transmission process is completed. Every time when potential sender tries to send RTS, the *CW* of the potential sender will double. After successfully transmitting data, the receiver sends an ACK back to the sender. By receiving an ACK, the sender resets its *CW* size to *CWmin*.

Chapter 3
Proposed Schemes

This chapter provides the design overview, ideas and diagrams of our proposed schemes. Our proposed schemes can be applied to two different scenarios, namely DCF (Hidden terminal problem) and EBN (Hidden receiver problem). Both design to overcome the unfairness between competing traffics, reducing the probability of collisions, improve system throughput and the QoS of networks.

3.1 Design Overview

Xu *et al*. [12] point out that binary exponential backoff algorithm has a number of disadvantages. One major disadvantage is the problem with fairness because it always favors the last succeeding node. The reason for this is that binary exponential backoff algorithm tends to prefer the last contention winner and new contending nodes over other nodes when allocating channel access (when the sender receives an ACK from the receiver, the sender only resets its *CW* size to *CWmin*, but not others). This is done by choosing a random backoff interval from *CWmin* which has a smaller size for new contending nodes and contention winners.

Based on above mentioned problem, we proposed a scheme that can be applied on DCF and EBN scheme to improve the fairness and the performance of hidden terminal and receiver problems. As well as effectively avoid blindly backoffs and retransmissions in DCF and EBN scheme.

3.2 Proposed Scheme Applied on DCF (Scheme 1)

We introduce an additional backoff procedure in DCF as illustrated in Figure 10.

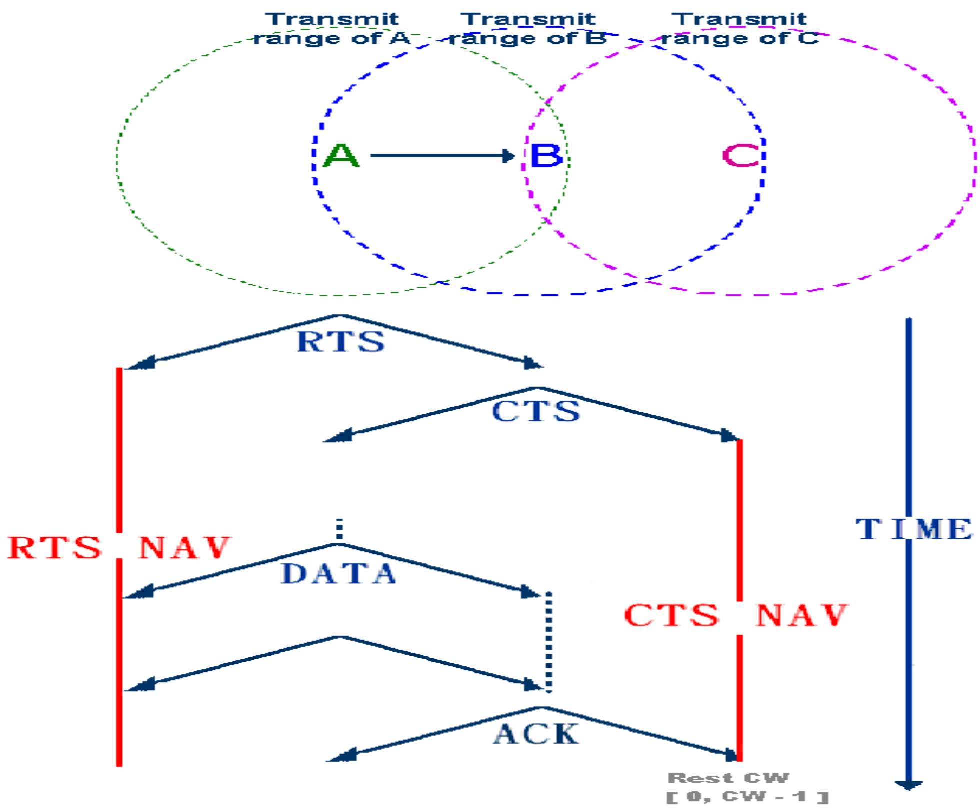


Figure 10: Proposed Scheme Applied on DCF (Scheme 1)

Let us consider the case where node A and node C are hidden from each other and node A want to transmit data to node B. Node A first sends a RTS packet to node B, then node B responds with a CTS packet. After receiving this CTS packet, node A begins to transmit data to node B. If any other node overhears the RTS or CTS packets they will defer its own transmission for a certain period of time indicated by RTS/CTS packets. After data has been successfully transmitted, node B sends an ACK back to node A. By receiving an ACK from node B, node A and node C

(Hidden terminal) resets their *CW* size back to *CWmin*.

By resetting the backoff timer of the node C after the data transmission has completed may reduce the difference in contention window of all senders so the chance of loosing the next contention is negligible thus improve the unnecessary backoffs, fairness and throughput between them.

3.3 Proposed Scheme Applied on EBN Scheme (Scheme 2)

We use the same idea and adapt it to EBN scheme, but this time we reset the *CW* of potential sender (node D) rather than reset the *CW* of ACK receiver (node C) illustrated in Figure 11.

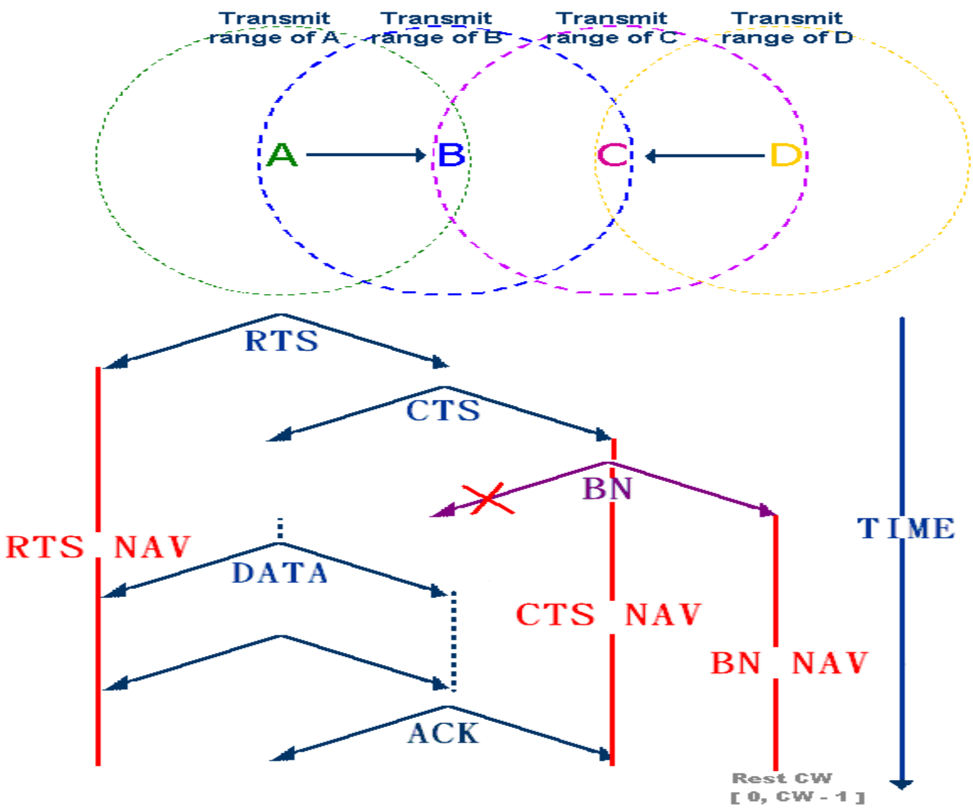


Figure 11: Proposed Scheme Applied on EBN Scheme (Scheme 2)

Let us consider the case where an ongoing transmission from node A to node B, node D wants to initiate a transmission to node C. Node A first sends a RTS packet to node B, then node B responds with a CTS packet. At this moment, node A and node C received this CTS packet at the same time then checks to see whether it is the intended receiver, node C realized that it is not the intended receiver, so it indicates that node C is a hidden terminal, then node C broadcast a BN packet to node D announcing the forthcoming data transmission and the deferring state of node C. If node D tries to communicate with node C, node C will not responds to it until the transmission process is complete. Every time when node D tries to send RTS to node C, the *CW* of node D will double. When node A successfully transmit its data, node B sends an ACK back to node A indicating it has correctly received the data. After node A received an ACK from node B, the Network Allocation Vector (NAV) on node D (set by BN packet) will end at the same time, therefore node A will reset its *CW* size back to *CWmin* and by the end of *BN_NAV*, node D will also reset its *CW* size back to *CWmin*. By resetting the backoff timer of the potential sender (neighbors of hidden terminal) after the data transmission has completed may reduce the difference in *CW* of all senders so the chance of losing the next contention is negligible thus improve the unnecessary backoffs, throughput and fairness between them.

Chapter 4
Simulations Results

This chapter introduces the simulation environments, the parameters and the scenarios of the simulation network, as well as the simulation results and discussions of these results.

4.1 Simulation Environments

4.1.1 Simulation Parameters

We evaluate the performance, throughput, fairness and unreplied RTS ratio of our proposed schemes by using NS-2 [10]. Simulations were carried out to exam our proposed scheme against DCF and EBN scheme in four different scenarios. Namely light traffics, heavy traffics, hidden terminal and hidden receiver problem.

Simulations performed on experiment one, two and three are based on Proposed Scheme Applied on DCF (Scheme 1) and experiment four is based on Proposed Scheme Applied on EBN Scheme (Scheme 2).

We used CBR (Constant Bit Rate) traffics and UDP packets only in all four experiments and we assume the packet length for experiment one, two and three are 1024 bytes with Destination Sequenced Distance Vector (DSDV) routing protocol, and we assume the packet length for experiment four is 547 bytes with Dynamic Source Routing (DSR) routing protocol.

In our experiments, we use the number of nodes to verify the traffic load of the network. In addition to the traffic model, the bandwidth model in our simulation is based on the fixed channel bandwidth model, where each channel has a fixed bandwidth. In order to simulate the saturation condition, each station is assumed to always have data packets to transmit.

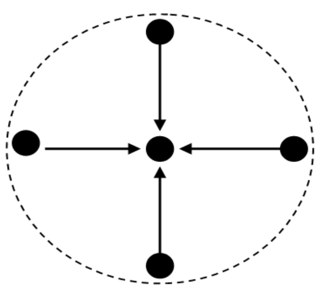
In our fairness calculations we will use Jain's fairness index [4], which is a well-known index of fairness and suitable for many situations.

$$Fairness\_Index = \frac{\left(\sum x_i\right)^2}{\left(n \cdot \sum x_i^2\right)} \tag{1}$$

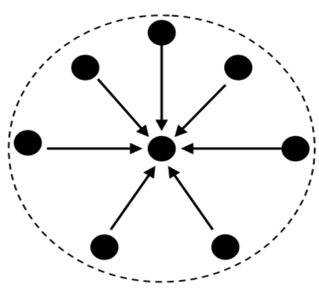
Here x_i is the throughput of the i th flow, e.g. the amount of data that has been successfully transferred from the sender to the receiver in each flow, n is the number of flow. The closer fairness index is to the value 1, the better (more equally) the bandwidth is utilized during the traffic flows.

4.1.2. Scenarios

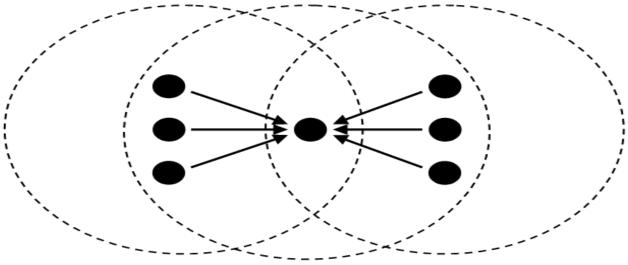
In order to evaluate the performance of the proposed scheme, four scenarios are considered, as illustrated in Figire 12 (a)-(d).



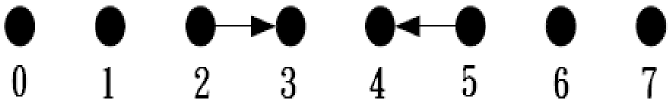
(a) Single-hop scenario with four senders



(b) Single-hop scenario with seven senders



(c) Hidden Terminal



(d) Hidden Receiver

Figure 12: Simulation Scenarios

4.2 Simulation Results and Discussions

4.2.1 Experiment One (Light Traffic Scenario)

For this simulation, we use the scenario illustrated in Figure 12 (a), which is a single-hop scenario, it has four nodes and a common receiver with long-lived data traffic from 0-50 seconds (each node always has packets to send to the receiver).

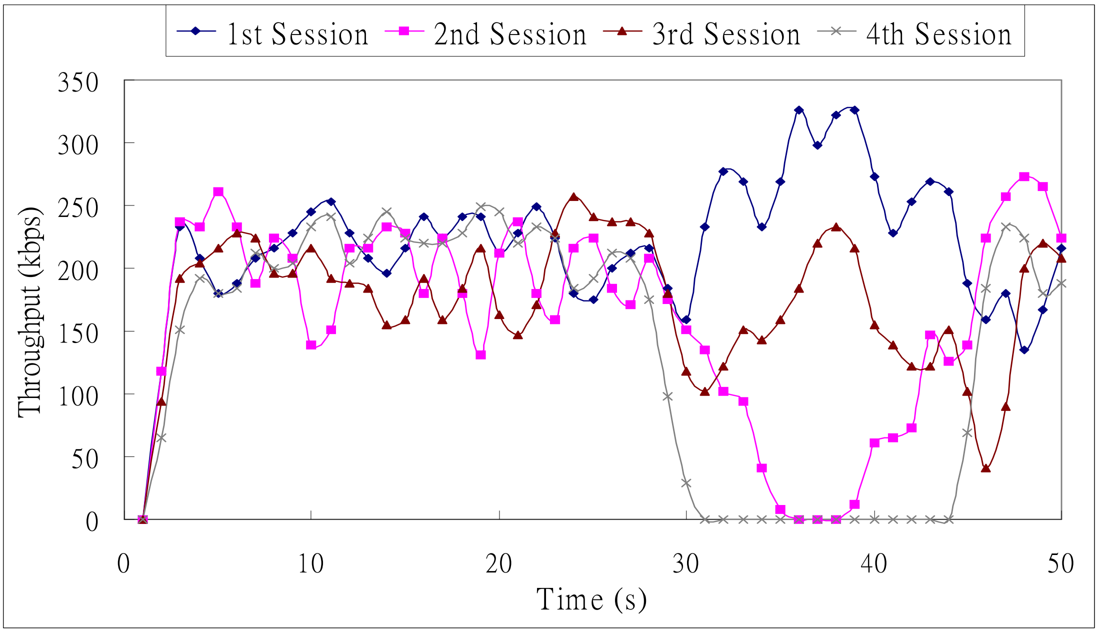


Figure 13: DCF (Experiment 1)

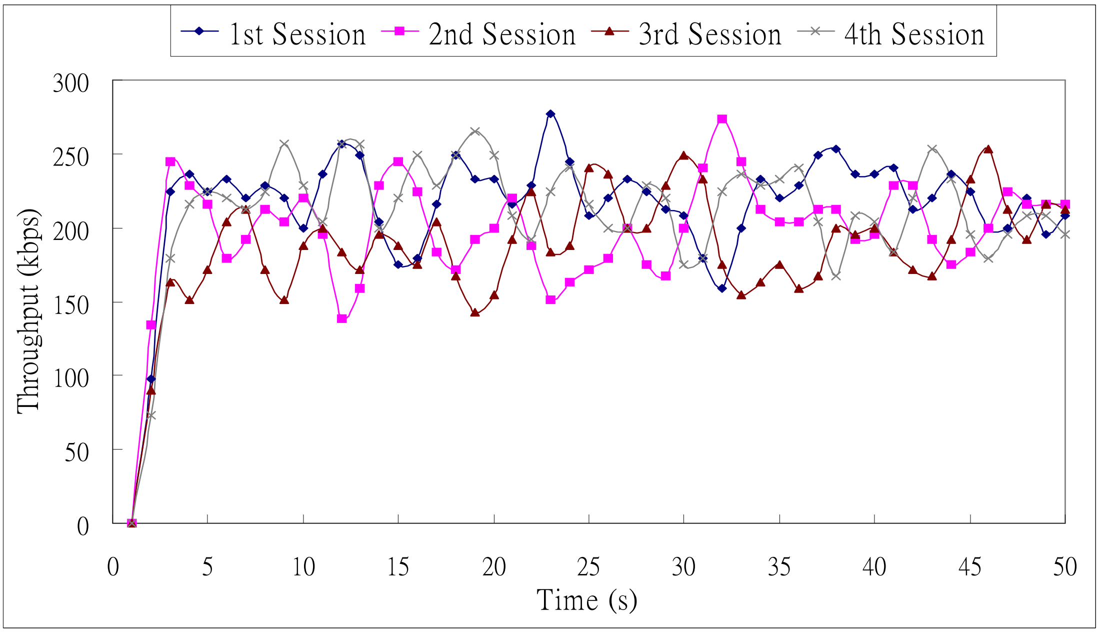


Figure 14: Proposed Scheme 1 (Experiment 1)

First, we compare the average throughput between DCF and proposed scheme with four sessions (senders), as illustrated in Figure 13 and Figure 14.

TABLE I
SIMULATION RESULTS ON FIGURE 12 (a)

Session #	DCF (kbps)	Proposed (kbps)
1	220	215
2	160	200
3	170	185
4	135	210
Total	685	810

As illustrated in Figure 13, Figure 14 and Table I, our proposed scheme achieved a far more stabilized average throughput than DCF. In DCF, the throughputs shared by these sessions are fluctuating from 0 kb to 330 kb (Figure 13) and proposed scheme only fluctuate from 140 kb to 280 kb (Figure 14). Therefore, channel resources of our proposed scheme are fairly shared between these sessions.

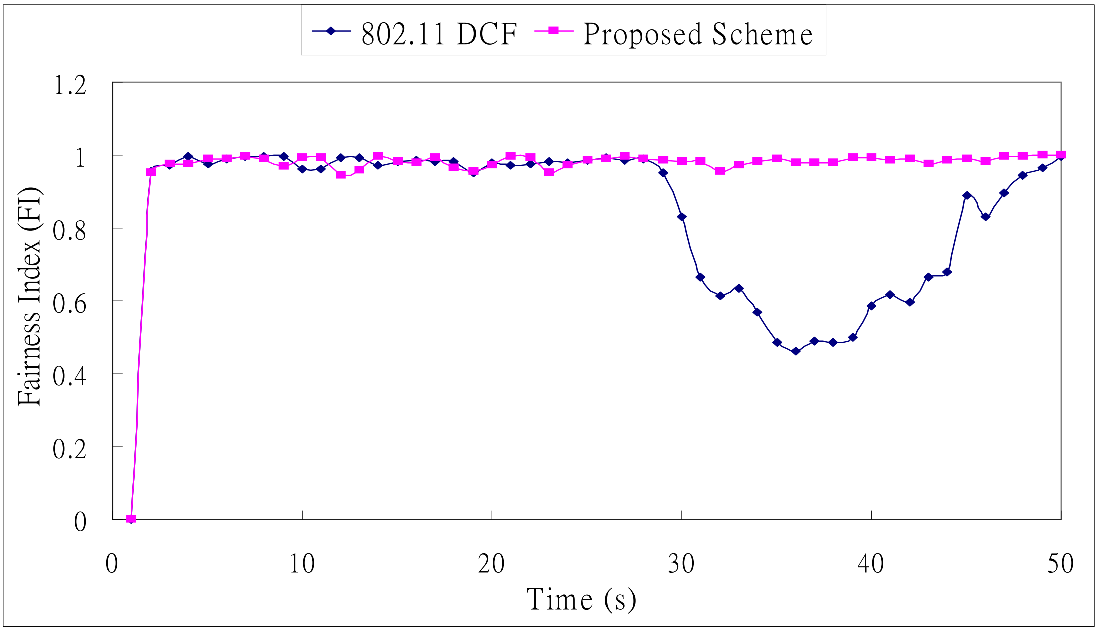


Figure 15: Fairness Index (Experiment 1)

We calculate the Fairness Index (FI) using Equation (1) to compare the fairness between DCF and proposed scheme. As we can see in Figure 15, the short-term FI of our proposed scheme outperformed DCF as well as long-term FI, because long-term FI of DCF is 0.81 and our proposed scheme achieves much higher long-term FI of 0.95. With this result, we concluded that our proposed scheme achieves far better stability and fairness than DCF in this scenario.

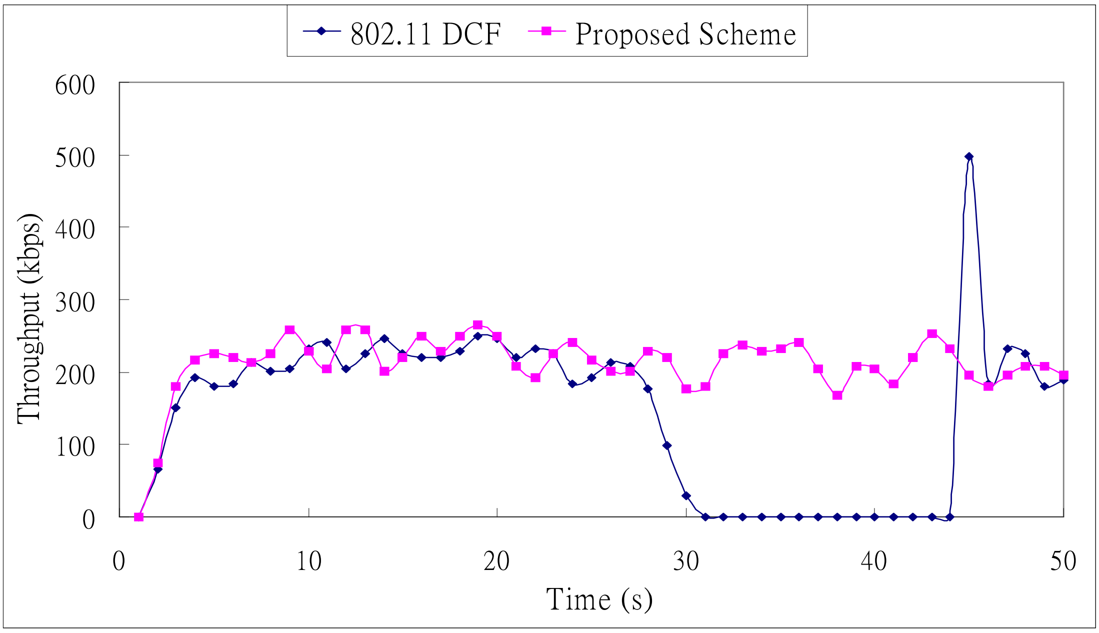


Figure 16: System Throughput (Experiment 1)

The result illustrated in Figure 16, shows that the system throughput of DCF change drastically and only achieved 34415 kb. In our proposed scheme, we achieved far better system throughput of 40400 kb and it remains stable with improvement of 17.3%.

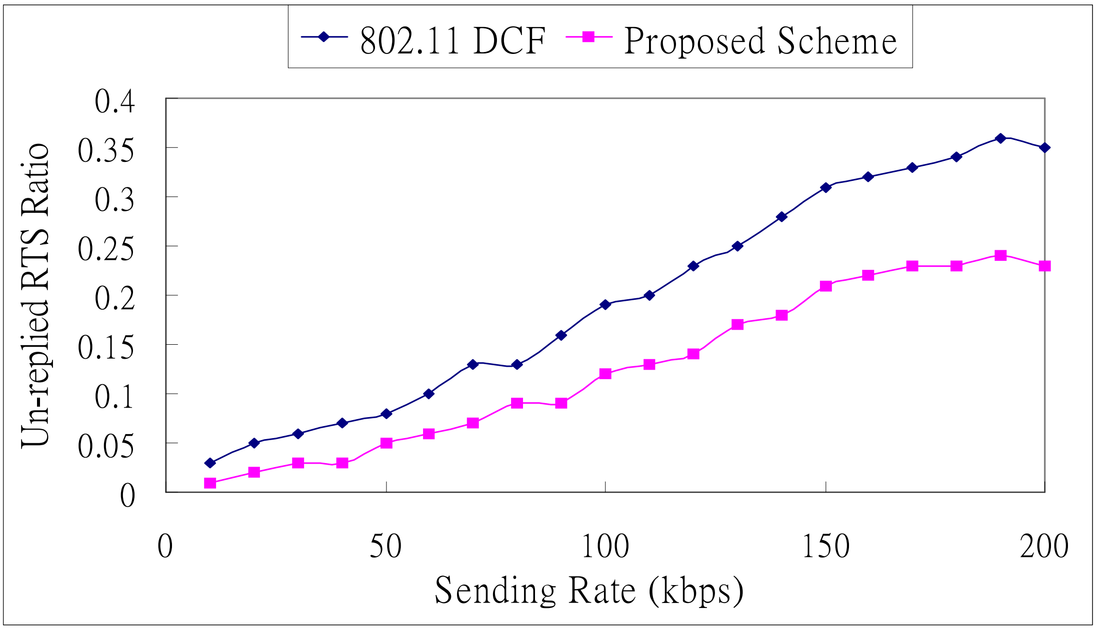


Figure 17: Unreplied RTS Ratio (Experiment 1)

Figure 17 shows the unreplied RTS ratio against data transmisson rate ranging from 10 kbps to 200 kbps and increment of 10 kbps. Smaller unreplied RTS ratio means more effective in unnecessary RTS transmission control and vice versa. With the increase of transmissions, the unreplied RTS ratio will also increase. Unreplied RTS ratio of proposed scheme is always lower than DCF with an average reduction of 35.78% in light traffic scenario, which indicates the efficiency of our proposed scheme in restricting unnecessary RTS propagation.

4.2.2 Experiment 2 (Heavy Traffic Scenario)

For this simulation, we use the scenario illustrated in Figure 12 (b), which it is a single-hop scenario, it has seven nodes and a common receiver with long-lived data traffic from 0-50 seconds.

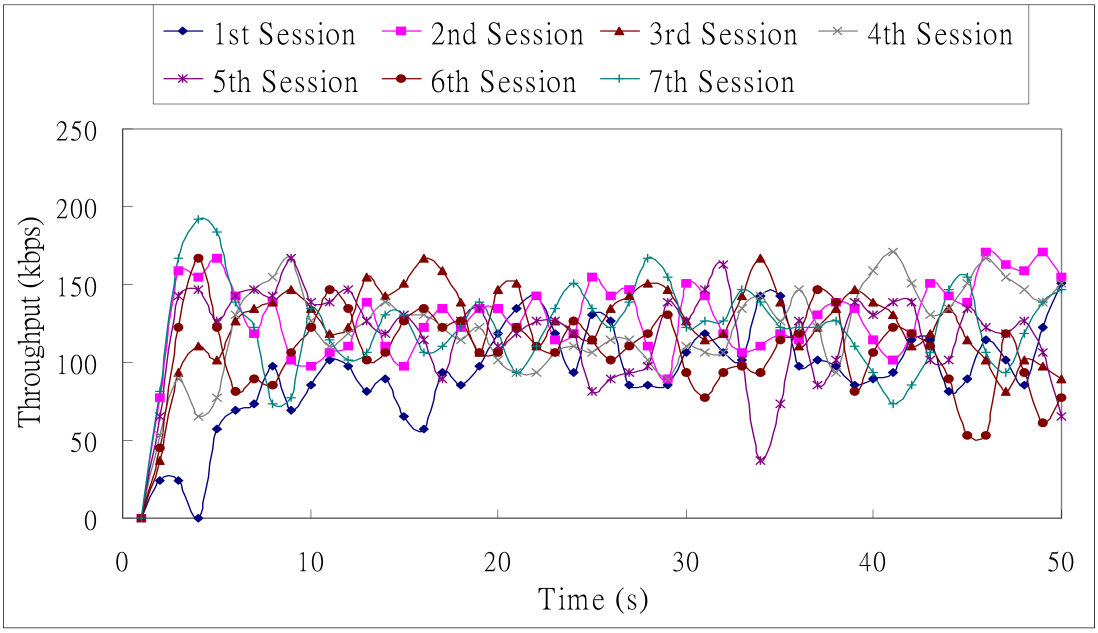


Figure 18: DCF (Experiment 2)

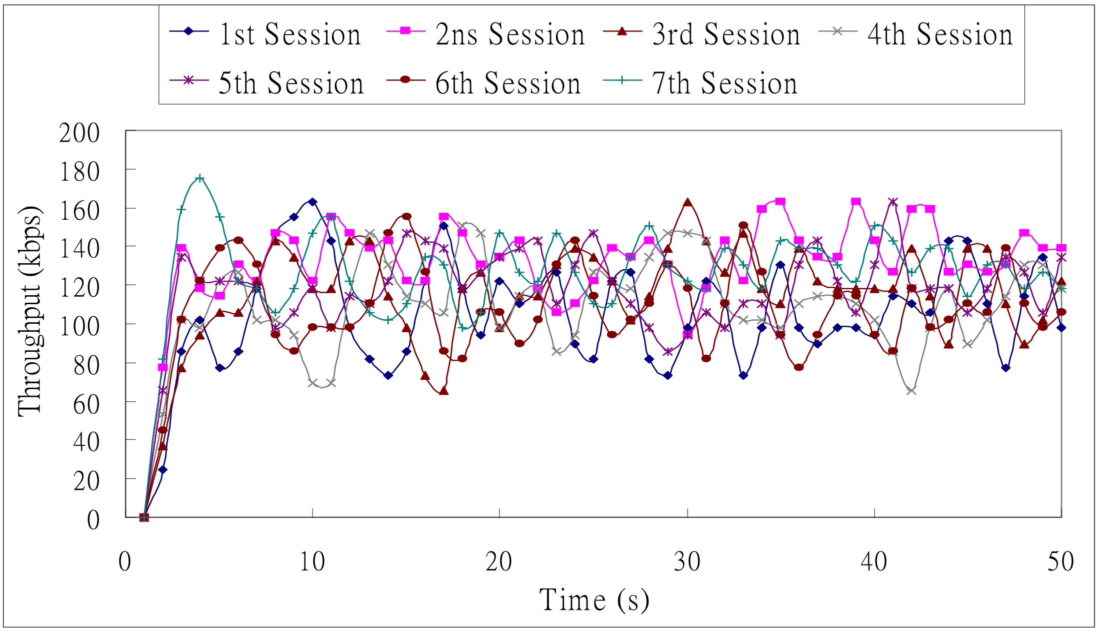


Figure 19: Proposed Scheme 1 (Experiment 2)

First, we compare the average throughput between DCF and proposed scheme with seven sessions, as illustrated in Figure 18 and Figure 19.

TABLE II

SIMULATION RESULTS ON FIGURE 12 (b)

Session #	DCF (kbps)	Proposed (kbps)
1	90	105
2	130	130
3	120	115
4	120	110
5	115	115
6	105	110
7	120	125
Total	800	810

As illustrated in Figure 18, Figure 19 and Table II, our proposed scheme achieved more stabilized average throughput than DCF. In DCF, the throughputs shared by these sessions are fluctuating from 35 kb to 190 kb (Figure 18) and proposed scheme only fluctuate from 65 kb to 175 kb (Figure 19). Therefore, channel resources of our proposed scheme are fairly shared between these sessions.

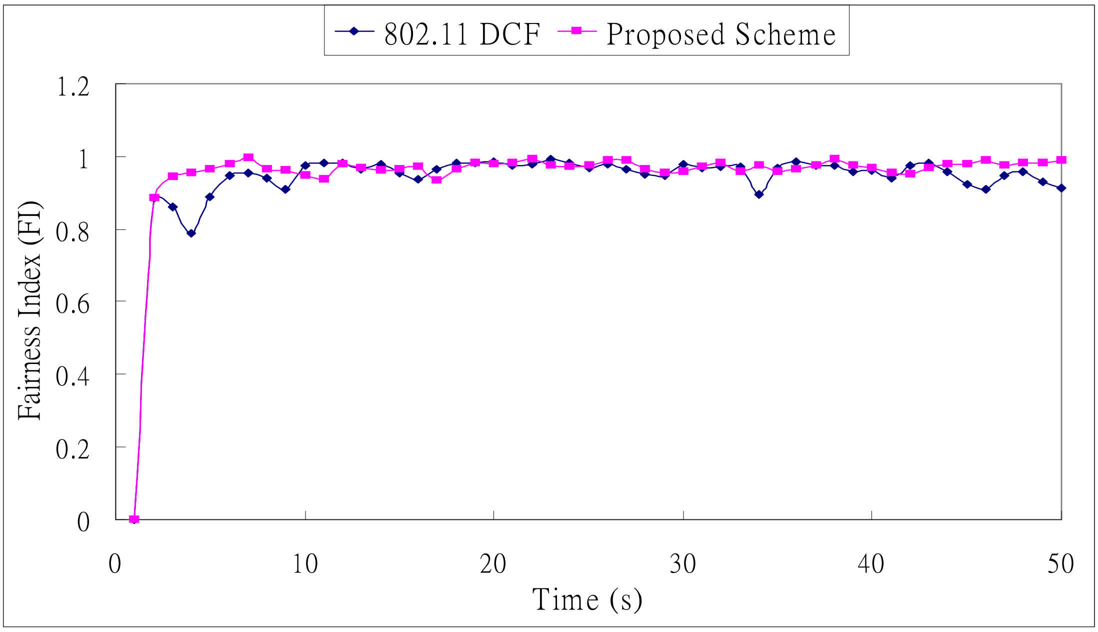


Figure 20: Fairness Index (Experiment 2)

As we can see in Figure 20, the short-term FI of our proposed scheme outperformed DCF as well as long-term FI, because long-term FI of DCF is 0.91 and our proposed scheme achieves higher long-term FI of 0.94. With this result, we can conclude that our proposed scheme perform well in light traffics as well as in heavy traffics too.

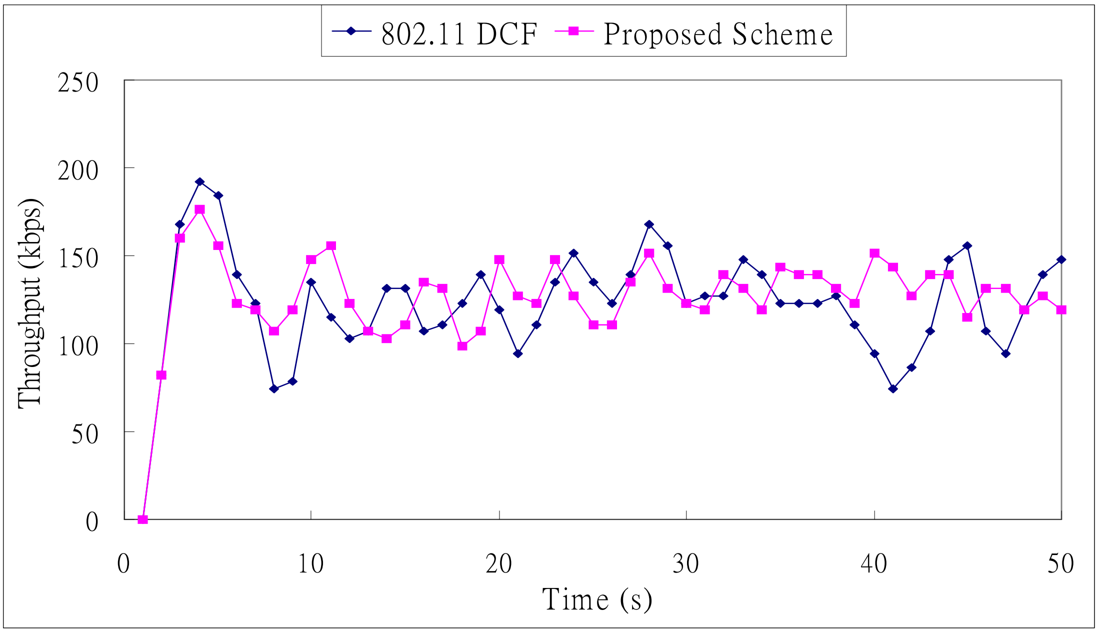


Figure 21: System Throughput (Experiment 2)

The result illustrated in Figure 21, shows that DCF achieved 40400 kb in system throughput and our proposed scheme with a slight increase of 40550 kb. It seems like DCF and proposed scheme process similar amount of data packets in heavy traffics but not as fair as our proposed scheme.

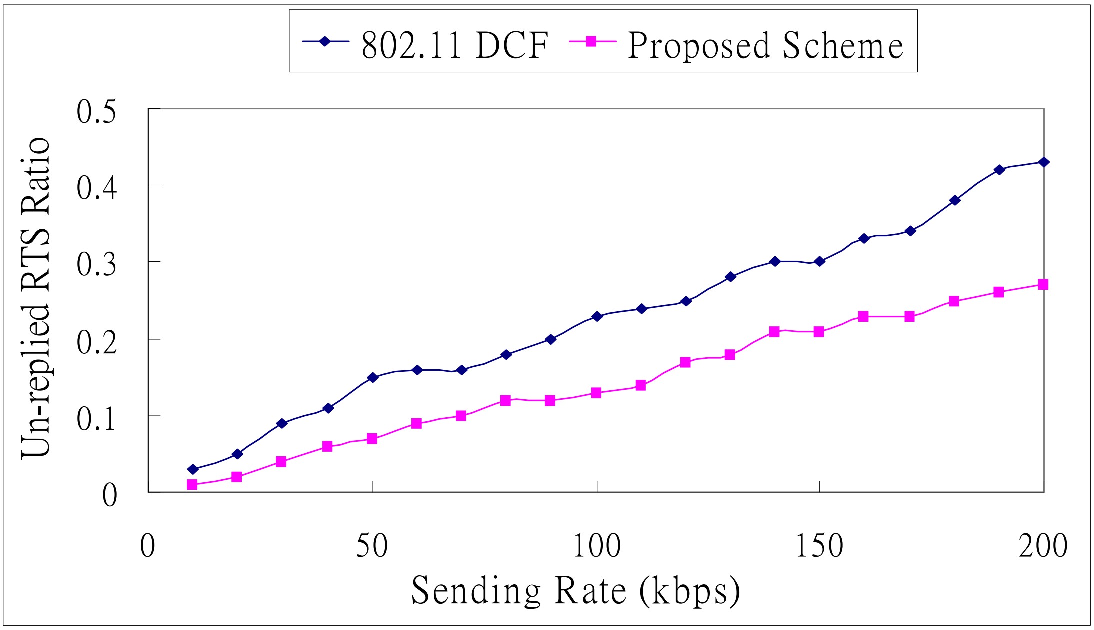


Figure 22: Unreplied RTS Ratio (Experiment 2)

Figure 22 shows the unreplied RTS ratio of proposed scheme is always lower than DCF with an average reduction of 31.14% in heavy traffic scenario, which indicates the efficiency of our proposed scheme in restricting unnecessary RTS propagation.

4.2.3 Experiment 3 (Hidden Terminal)

For this simulation, we use the scenario illustrated in Figure 12 (c), this is a hidden terminal scenario; it has two independent groups (which cannot hear each other) with three nodes each transmitting to a common receiver with long-lived data traffic from 0-50 seconds.

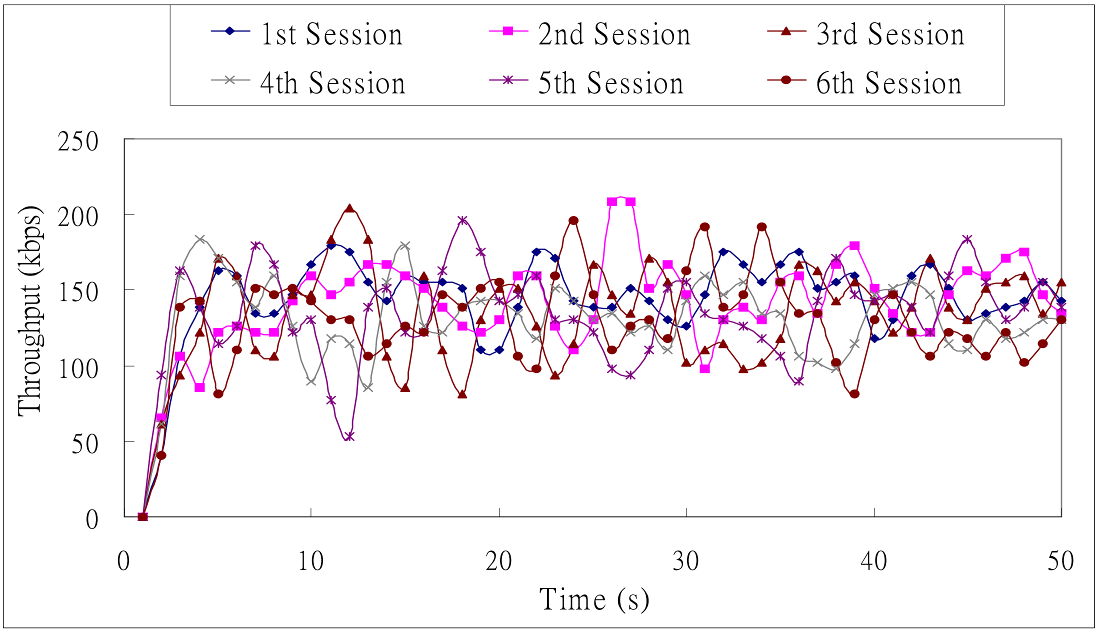


Figure 23: DCF (Experiment 3)

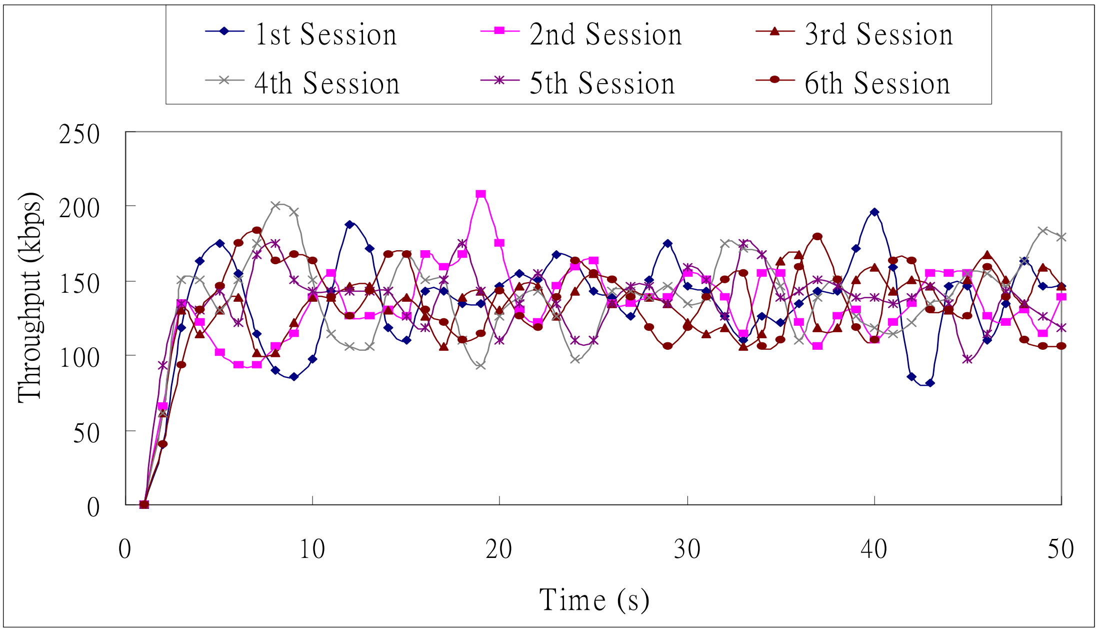


Figure 24: Proposed Scheme 1 (Experiment 3)

First, we compare the average throughput between DCF and proposed scheme with six sessions, as illustrated in Figure 23 and Figure 24.

TABLE III
SIMULATION RESULTS ON FIGURE 12 (c)

Session #	DCF (kbps)	Proposed (kbps)
1	145	135
2	140	135
3	135	130
4	130	140
5	130	135
6	125	135
Total	805	810

As illustrated in Figure 23, Figure 24 and Table III, our proposed scheme achieved little more stabilized average throughput than DCF. In DCF, the throughputs shared by these sessions are fluctuating from 50 kb to 210 kb (Figure 23) and proposed scheme fluctuate from 80 kb to 205 kb (Figure 24). Therefore, channel resources of DCF and proposed scheme are fairly shared between these sessions in this scenario.

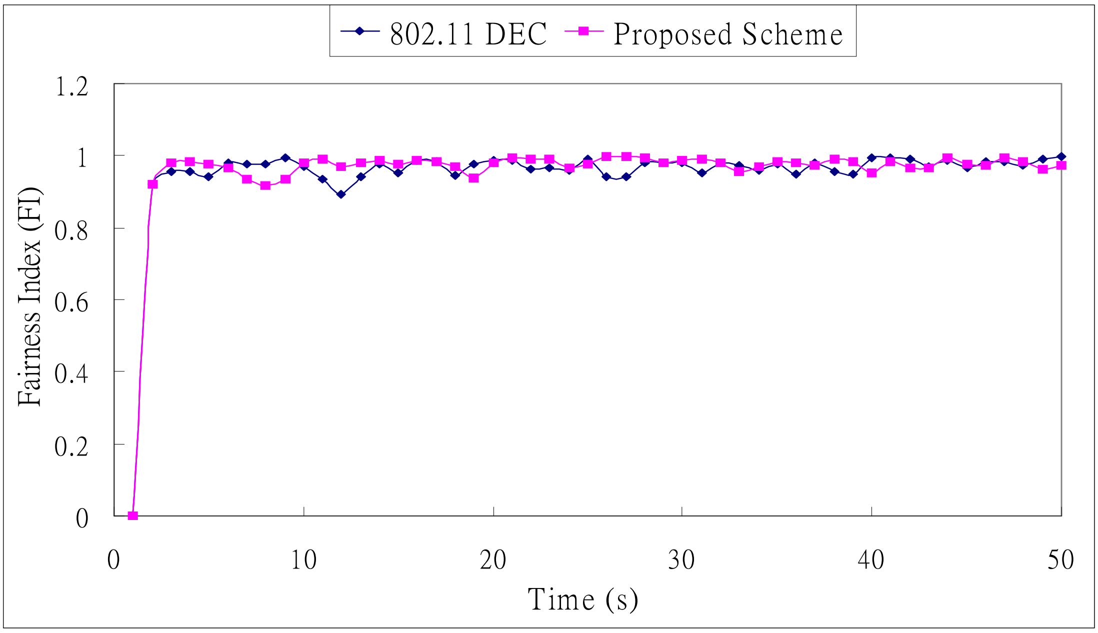


Figure 25: Fairness Index (Experiment 3)

As we can see in Figure 25, DCF and our proposed scheme achieve similar result in short-term FI as well as in the long-term FI. DCF achieve long-term FI of 0.94 and our proposed scheme achieves little higher long-term FI of 0.95.

With this result, we can conclude that the performance of DCF and our proposed scheme are similar in hidden terminal.

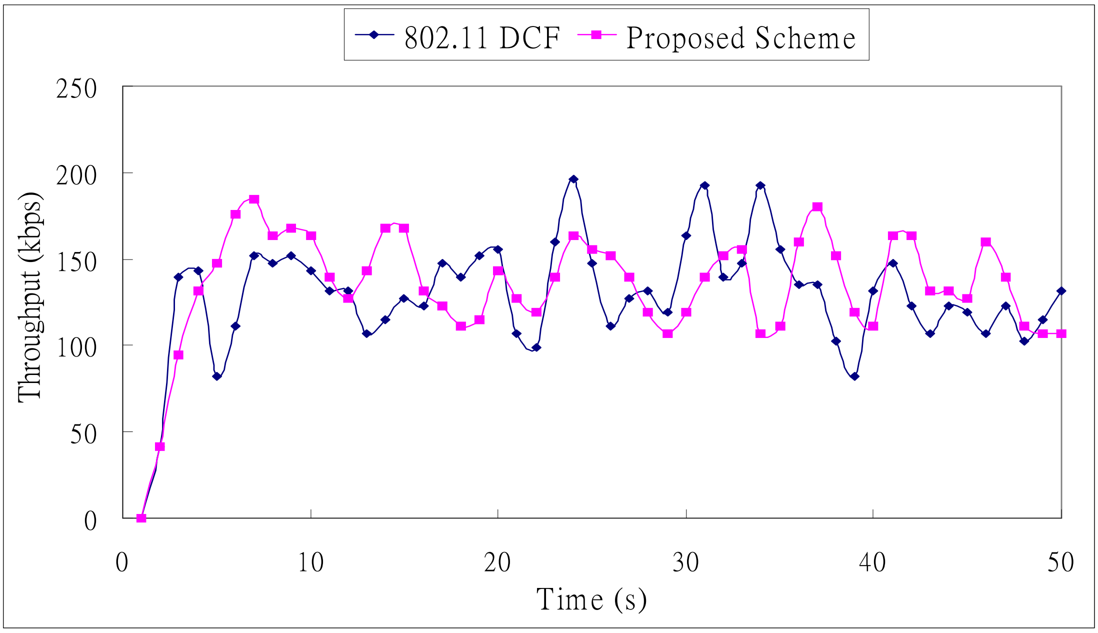


Figure 26: System Throughput (Experiment 3)

The result illustrated in Figure 26 shows that DCF achieved 40295 kb in system throughput and our proposed scheme with a slight increase of 40350 kb. As from this result, we can see that DCF and our proposed scheme may process similar amount of packets and achieve nearly the same FI in hidden terminal.

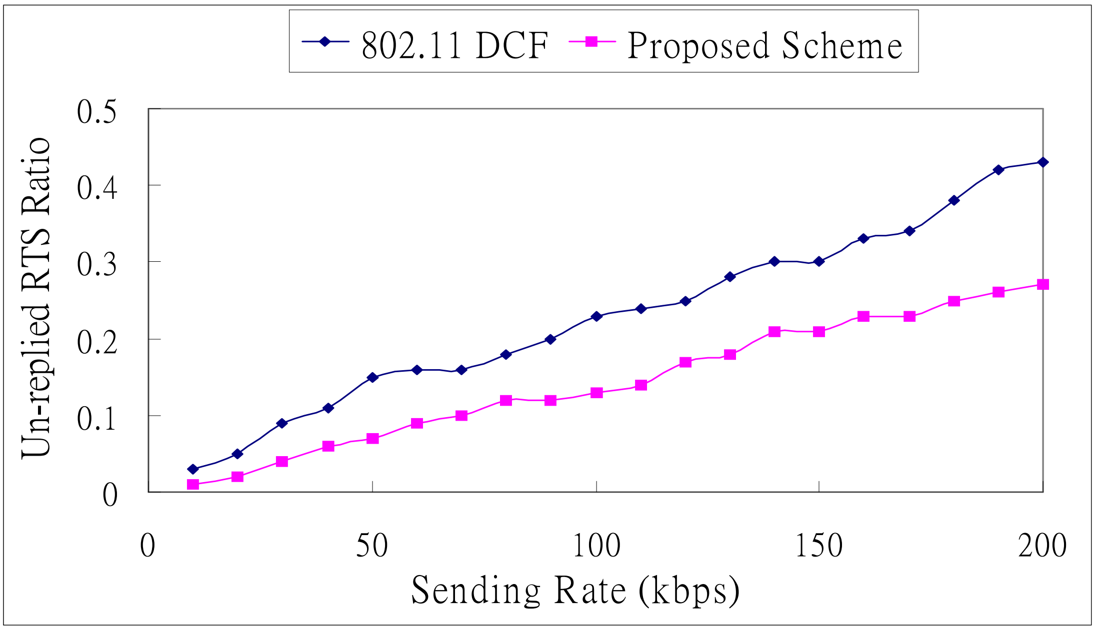


Figure 27: Unreplied RTS Ratio (Experiment 3)

Figure 27 shows the unreplied RTS ratio of proposed scheme is always lower than DCF with an average reduction of 45.18% in hidden terminal scenario, which indicates the efficiency of our proposed scheme in restricting unnecessary RTS propagation.

4.2.4 Experiment 4 (Hidden Receive)

For this simulation, we use the scenario illustrated in Figure 12 (d), it is a hidden receiver scenario; it has two sessions of long-lived data traffics, first session between node 2 and node 3 starts at 5 second, and second session between node 5 and node 4 starts at 10 seconds and both sessions ends at 100 seconds.

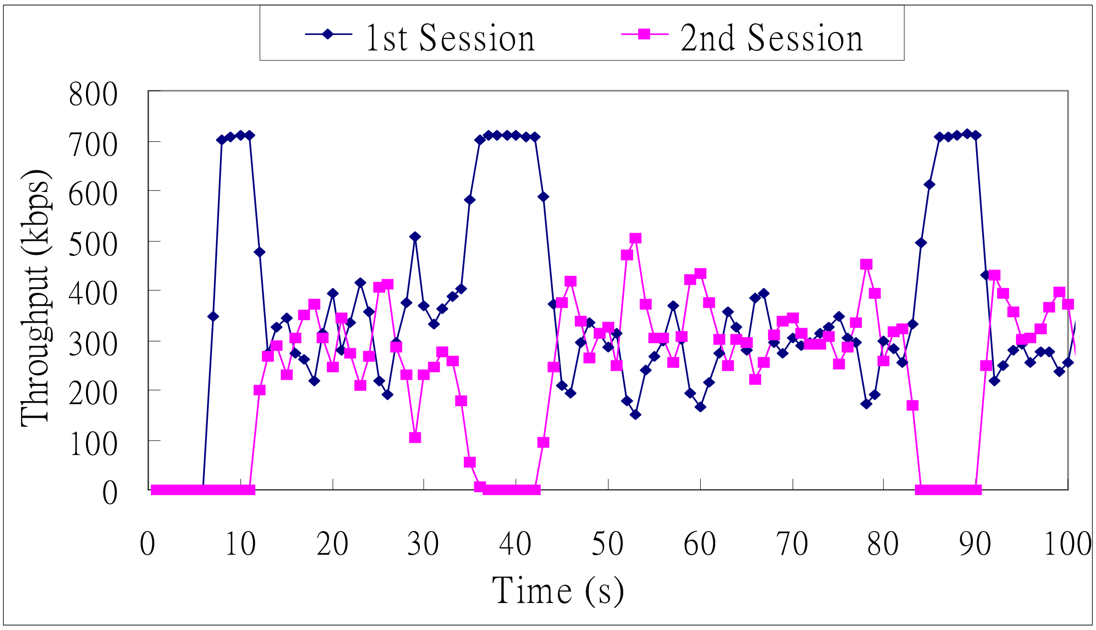


Figure 28: DCF (Experiment 4)

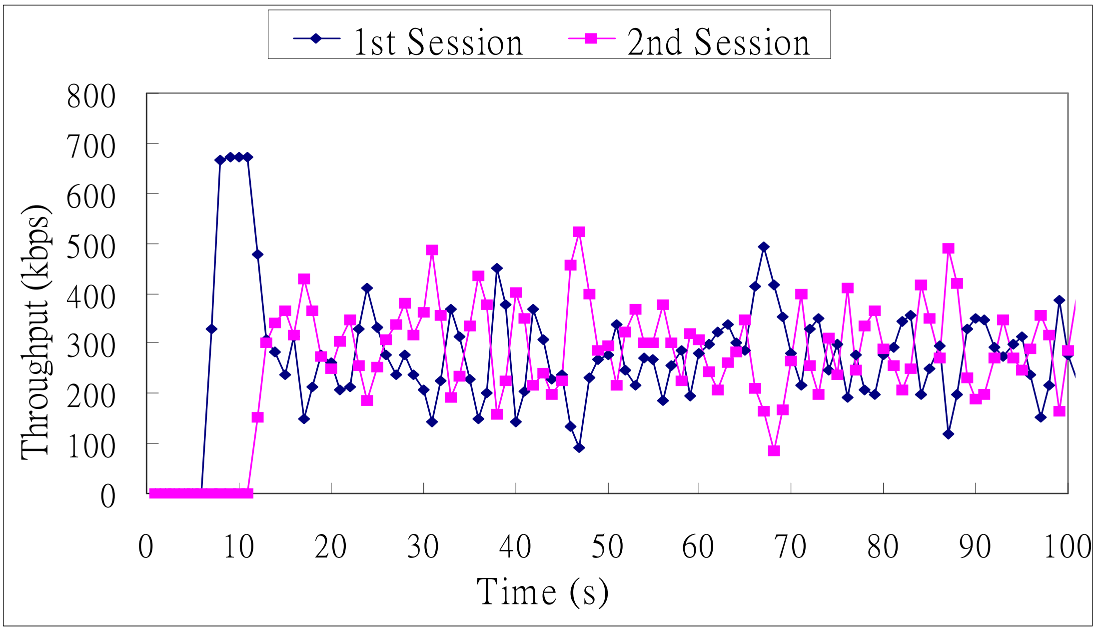


Figure 29: EBN Scheme (Experiment 4)

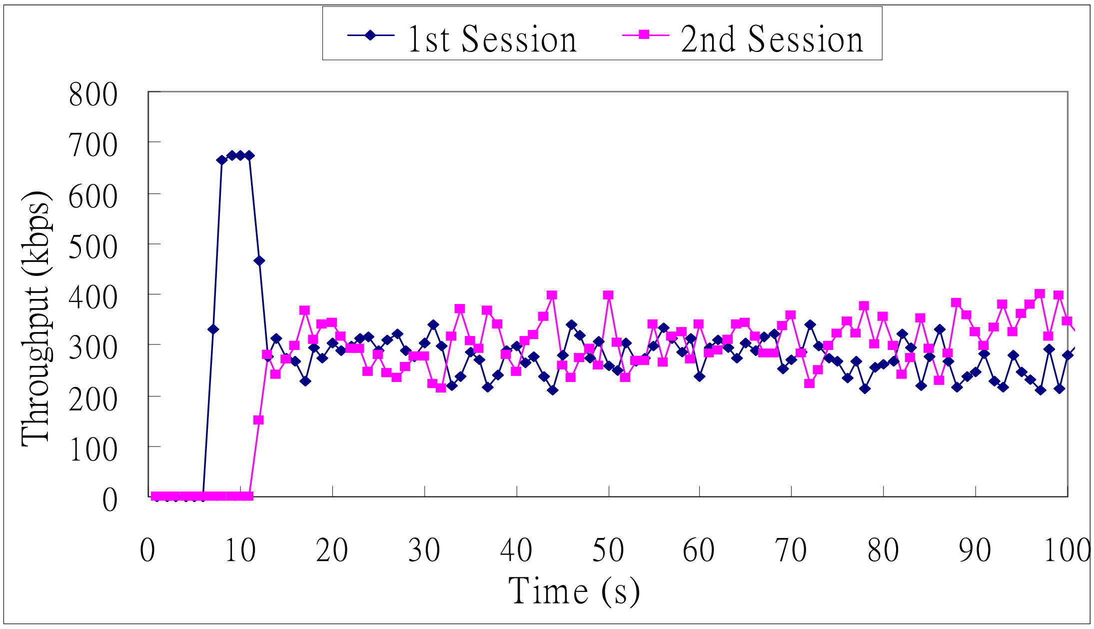


Figure 30: Proposed Scheme 2 (Experiment 4)

First, we compare the average throughput between DCF, EBN scheme and proposed scheme with two sessions, as illustrated in Figure 28, Figure 29 and Figure 30.

TABLE IV
SIMULATION RESULTS ON FIGURE 12 (d)

Session #	DCF (kbps)	EBN (kbps)	Proposed (kbps)
1	360	275	300
2	230	266	295
Total	590	541	595

As illustrated in Figure 28, Figure 29, Figure 30 and Table IV, our proposed scheme achieved a far more stabilized average throughput than DCF and EBN scheme. In DCF, the throughputs shared by these sessions are fluctuating from 0 kb to 700 kb (Figure 28) and EBN scheme more stable and more equally shared traffics then DCF which fluctuate from

100 kbps to 500 kbps (Figure 29). As for our proposed scheme, it only fluctuates from 260 kb to 360 kb (Figure 30). Therefore, channel resource of EBN scheme performed better then DCF but our proposed scheme outperformed both DCF and EBN scheme in this scenario.

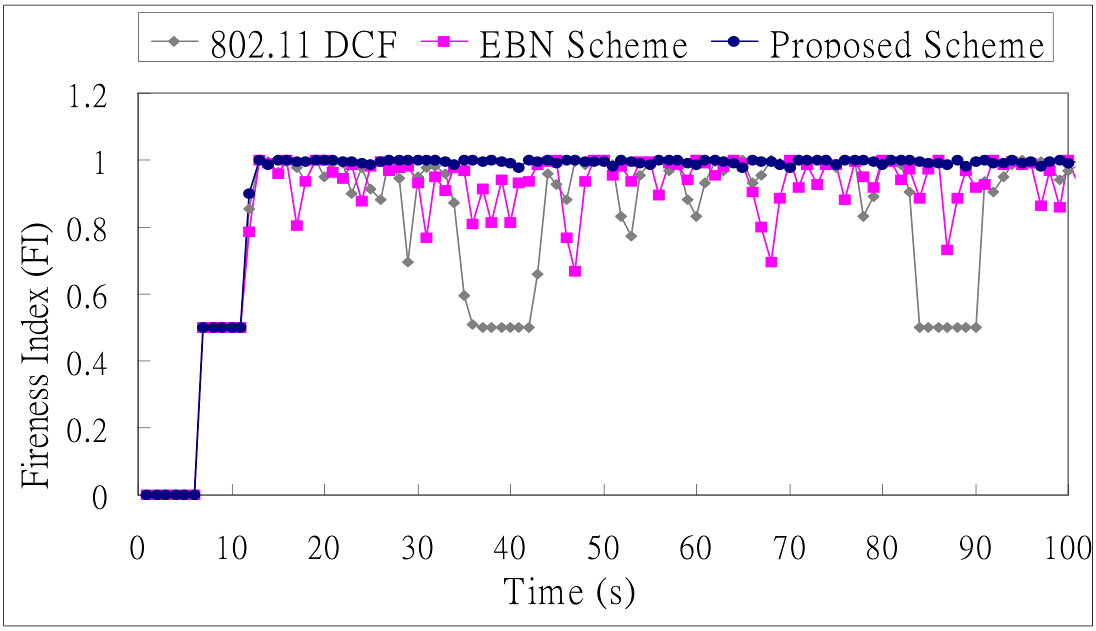


Figure 31: Fairness Index (Experiment 4)

As we can see in Figure 31, our proposed scheme outperformed both DCF and EBN scheme in the short-term FI as well as the long-term FI, because long-term FI of DCF is 0.81 and EBN scheme achieves higher long-term FI of 0.87. As for our proposed scheme, it achieves much higher long-term FI of 0.91. With this result, we concluded that our proposed scheme achieves far better stability and fairness than DCF and EBN scheme in this scenario.

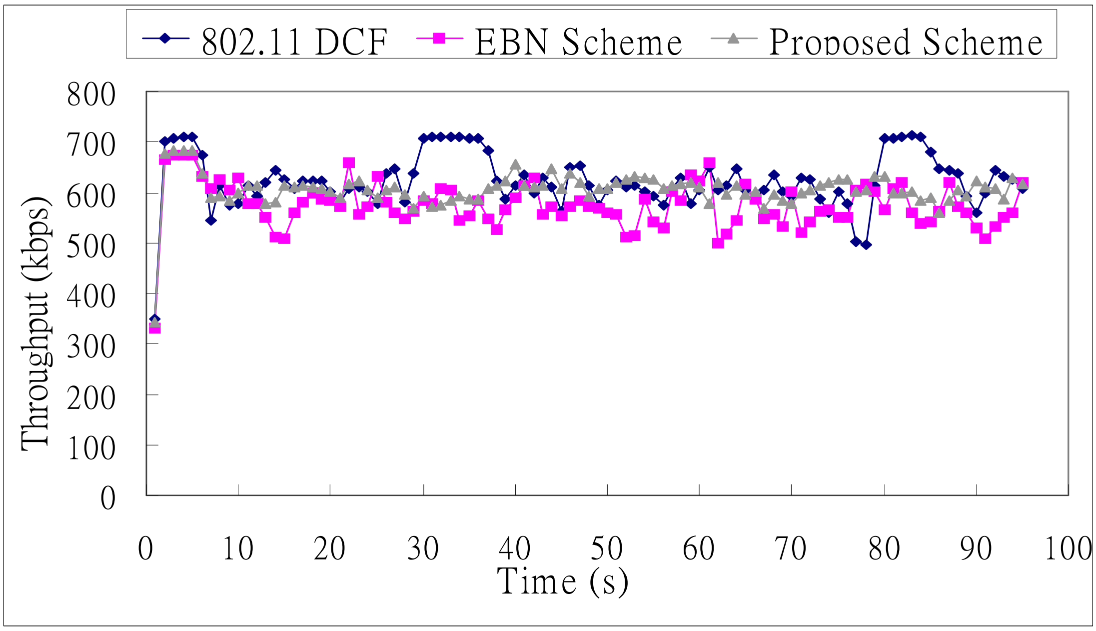


Figure 32: System Throughput (Experiment 4)

The result illustrated in Figure 32 shows that DCF achieves 59200 kb in system throughput, EBN scheme with a decease system throughput of 54340 kb and our proposed scheme with a slight decrease of 55500 kb. As from this result, we can see that DCF achieves the highest system throughput out of these all but achieves the lowest FI among them. EBN scheme achieves lowest system throughput among them but achieves little better FI then DCF. As for our proposed scheme, it achieves better system throughput then EBN scheme but achieves the highest FI of all.

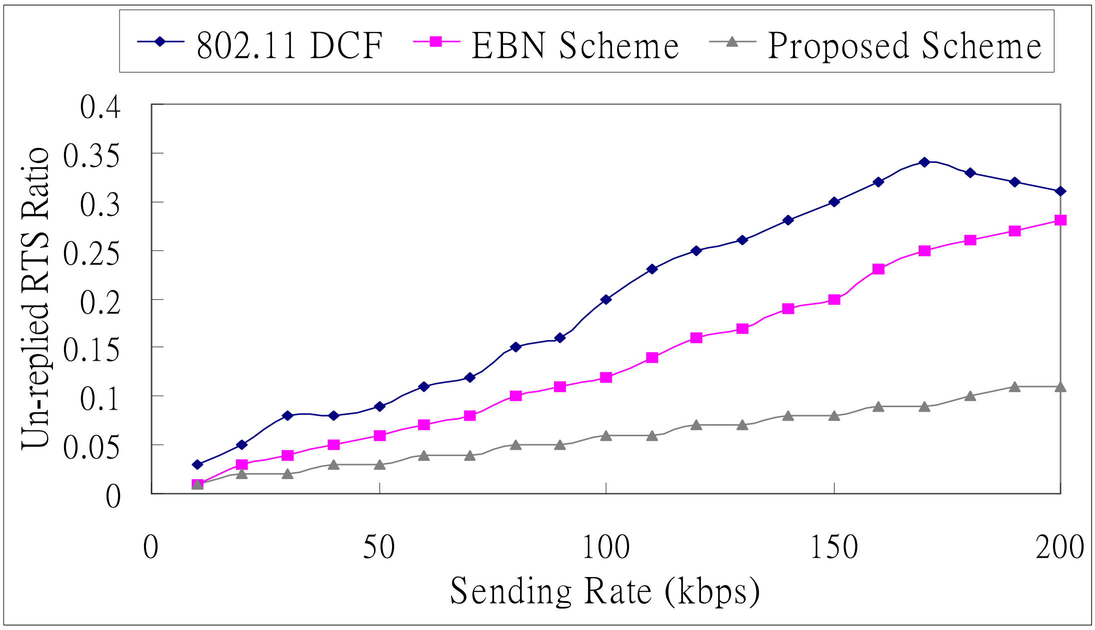


Figure 33: Unreplied RTS (Experiment 4)

Figure 33 shows the unreplied RTS ratio of EBN scheme and proposed scheme is always lower than DCF. EBN scheme performed well in unreplied RTS ratio simulation with an average reduction of 29.67%. However, our proposed scheme outperofrmed both DCF and EBN scheme with high average reduction of 66.08% in hidden receiver scenario, which indicates the efficiency of proposed scheme in restricting unnecessary RTS propagation.

Chapter 5
Conclusions and Future Works

5.1 Conclusions

In this thesis, we review the related problems in wireless environments and proposed two improved schemes with an additional procedure based on binary exponential backoff algorithm aiming at avoiding unnecessary RTS retransmission and achieving fairness between contention traffics.

Our proposed schemes modify the traditional RTS/CTS mechanism by resetting the backoff timer of the hidden node and potential sender after the data transmission has completed. The main idea of our proposed schemes is to reduce the difference in contention window of all senders so the chance of losing the next contention is negligible.

In our simulations, we performed four scenarios to examine our proposed schemes and the simulation results showed that the proposed schemes are much more efficient and fair than DCF and EBN scheme with high average reduction of unreplied RTS ratio.

5.2 Future Works

For our future work, we will perform more simulations and analysis our proposed schemes in a wider context. Like the impact of the routing algorithms and study the capacity influenced by different network models. Even try to adapt our proposed schemes to different type of networks.

BIBLIOGRAPHY

[1] IEEE Computer Society Committee LAN MAN Standards Committee, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. ANSI/IEEE Std. 802.11, 1999 Edition. The Institute of Electrical and Electronic Engineers.

[2] Vaduvur Bharghavan, Alan Demers, Scott Shenker, and Lixia Zhang, "MACAW: A media access protocol for wireless LANs," in Proceedings of ACM SIGCOMM '94. 1994, pp. 212–225, ACM.

[3] J. Goodman *et al.*, "Stability of Binary Exponential Backoff", app. In the Proc. of the 17-th Annual ACM Symp. Theory of Comp., Providence, May 1985.

[4] Raj Jain, Arjan Durresi, Gojko Babic, "Throughput Fairness Index: An Explanation," ATM Forum/99-0045, Feb. 1999.

[5] P. Karn, "MACA - A new channel access method for packet radio", ARRL/CRRL Amateur Radio 9th computer Networking Conference, 1990, pp. 134 - 140

[6] S. Khurana, A. Kahol, and A.P. Jayasumana, "Effect of hidden terminals on the performance of IEEE 802.11 MAC protocol," Proc. IEEE Conference on Local Computer Networks (LCN'98), pp.12 – 20, Oct. 1998.

[7] Leonard Kleinrock and Fouad A. Tobagi, "Packet switching in radio channels: Part 1 - Carrier Sense Multiple-Access modes and their throughput-delay characteristics," IEEE Transactions on Communications, vol. COM-23, no. 12, pp. 1400–1416, 1975.

[8] Dong Lihong and Jin Yan'an, "A Novel MAC Protocol for Hidden

Receiver Problem in Ad Hoc Networks.” Proceedings of the IEEE International Conference on Automation and Logistics August 18 - 21, 2007, Jinan, China

[9] S. Manaseer and M. Ould-kauoa, “A New Backoff Algorithm for MAC Protocol in MANETs,” 21st Annual UK Performance Engineering Workshop, pp 159-164, 2005.

[10]Network Simulator [Online]. Available: http://www.isi.edu/nsnam/ns/.

[11]Raphael Rom and Moshe Sidi, “Multiple Access Protocols: Performance and Analysis,” Springer Verlag, 1990.

[12]T. S. Shugong Xu, “Does the IEEE 802.11 MAC Protocol Work Well in Multihop Wireless Ad Hoc Networks?,” in IEEE Communications Magazine, 2001.

[13]Fouad Tobagi and Leonard Kleinrock, “Packet switching in radio channels: Part 2 - The hidden node problem in carrier sense multiple access modes and the busy tone solution,” IEEE Transactions on Communications, vol. COM-23, no. 12, pp. 1417–1433, 1975.