DUBLIN CITY UNIVERSITY
SCHOOL OF ELECTRONIC ENGINEERING

Simulation of IEEE 802.11 PCF function in GloMoSim

Michael Cloran
April 2004

99173719

BEng in Telecommunications

Supervised by
Dr. J. Murphy and DR. Sean Murphy.

Acknowledgements

I would like to take this opportunity to thank Dr. Sean Murphy and Dr. John Murphy for their guidance, patience, and commitment throughout this project.

Declaration

I declare that this project is solely my work and is not submitted in whole or in part to any other university.

Signed ....
Date ....

Abstract

The IEEE 802.11 standard has two access methods the distributed coordination function (DCF) and the point coordination function (PCF). The DCF access method is used for Ad-Hoc wireless networks and the PCF access method is used for infrastructure wireless networks.

This project attempts to simulate this PCF functionality under various traffic conditions and to gain experience of simulating wireless networks using the GloMoSim simulator.

The method of the solution used is to upgrade the existing IEEE 802.11 libraries contained in GloMoSim to agree fully with the most important parts of the IEEE 802.11 standard for the PCF access method. Specifically, this involves the design of the point coordination function access method.

Time was a issue with this project so only limited functionality was obtained but some results can be acquired from GloMoSim.

Table of Contents

Table of Figures

Table list

Abbreviations and acronyms

ACK	acknowledgement
Ad-Hoc	wireless network where the topology is changing in a random unplanned fashion
AMRIS	multicast protocol for Ad Hoc wireless networks
AMRoute	Ad-hoc multicast routing protocol
AP	access point
AST	multicast routing protocol
BSS	basic service set
BSSID	basic service set identification
CAMP	multicast routing protocol
CBR	continious bit rate
CF	contention free
CFP	contention-free period
CSMA	carrier sense multiple access
CSMA/CA	carrier sense multiple access with collision avoidance
CTS	clear to send
DBS	direct broadcast satellite
DCF	distributed coordination function
DIFS	distributed (coordination function) interframe space
DREAM	distance routing effect algorithm for mobility
DS	distribution system
DSR	The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks
DSS	distribution system service
DTIM	delivery traffic indication message
DVMRP	Distance-Vector Multicast Routing Protocol
EIFS	extended interframe space
FCS	frame check sequence
FH	frequency hopping
FTP	file transfer protocol
HTTP	hyper text transfer protocol
IBSS	independent basic service set
IFS	interframe space
IP	internet protocol
LAN	local area network
LAR	Location-Aided Routing in mobile ad hoc
MAC	medium access control
MACAW	Multiple Access with Collision Avoidance
MMWN	Multimedia Application in Mobile Wireless Networks
NAV	network allocation vector
NS-DSDV	routing protocol
ODMRP	On-Demand Multicast Routing Protocol
OSI	open systems interconnection

PCF	point coordination function
PIFS	point (coordination function) interframe space
RTS	request to send
SIFS	short interframe space
SSID	service set identifier
STA	station
TCP	transmission control protocol
TIM	traffic indication map
TU	time unit
UDP	user datagram protocol
WLAN	
WRP	Wireless Routing Protocol

Chapter 1 – Introduction

The IEEE 802.11 standard was devised in order to provide a reliable mechanism for delivery of information over a wireless medium and was designed for wireless LAN applications. Wireless communications has now become common place by the inception of this standard and by its use an station (STA) can communicate on a peer to peer basis. It is used for temporary and permanent wireless LAN implementations. This method of communication has become increasingly popular with mobile communication devices and the original IEEE 802.11 standard is being amended on an ongoing basis in order to improve the service, some of the amendments are, 802.11a increases data rates on 5GHz band with orthogonal frequency division multiplexing (OFDM), 802.11b increases data rates on 2.4GHz band with quaternary phase shift keying (QPSK) and differential QPSK, 802.11e Quality of Service, 802.11g increases data rates on 2.4GHz with OFDM[1].

IEEE 802.11 uses a Direct Sequence Spread Spectrum Modulation (DSSS) scheme1 at the Physical Layer. This standard operates in the unlicensed 2.4 GHz frequency band with a data transfer rate of 1-2 Mbps[2].

1.1 Basic Service Set

The basic service (BSS) is the range in which a communication can take place and is the basic building block for a IEEE 802.11 LAN.

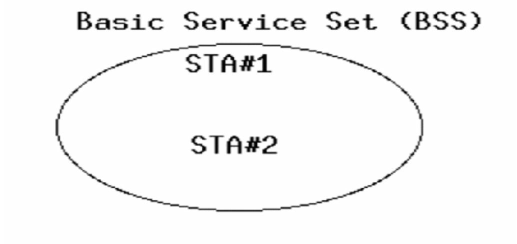


Fig.1.1 – Basic service set[3]

1 Section 3.4 Man03
2 802.11 standard 1999 edition
3 Similar diagram 802.11 standard section 5.2

Fig.1.1 shows two STAs within a BSS where the circle represents the range and if one moves outside this range communication is lost.

1.2 Infrastructure WLAN

This type of setup is for infrastructure wireless LANs Fig.1.2 where one of the STAs is the access point (AP) (gateway) to the distribution system. This AP allows each of the STAs in its BSS access through it to the distribution system and to other BSS[4]. For this type of wireless LAN each BSS is given an identity number.

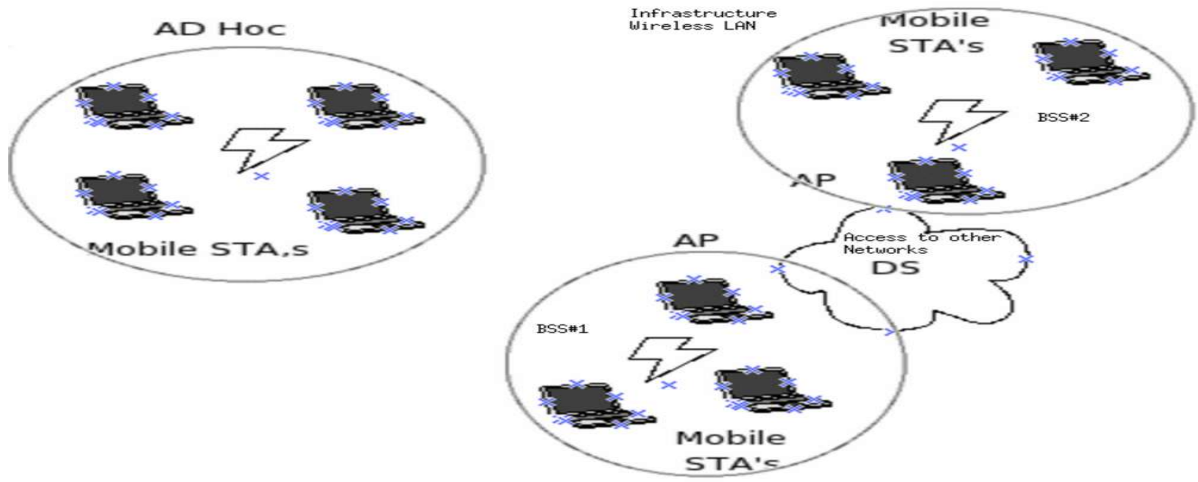


Fig.1.2.Basic topologies. 802.11[5]

1.3 Ad-Hoc wireless LAN

This is a wireless LAN which is not planned. Fig.1.2 shows several STAs in the same BSS and once they don't move out of the BSS communication can take place on a peer to peer basis. This is typically used for meetings where temporary communication is needed between mobile STAs but no wired access is available.

1.4 IEEE 802.11 Standard Introduction

The IEEE 802.11 standard has two access methods, the distributed coordination function (DCF) and the point coordination function (PCF). The PCF is only usable on infrastructure network configurations. DCF is used for Ad-hoc networks. The PCF access method allows fairness of use of resources amongst all STAs by the use of a polling mechanism. If both access methods are

[4] 802.11 standard section 5.2.2

[5] Similar diagram 802.11 standard section 5.2

used within the same BSS they coexist in a manner that permits both to operate concurrently within that BSS. When a PC is operating in a BSS, the two access methods alternate, with a contention-free period (CFP) followed by a contention period (CP)[6].

1.5 GloMoSim Implementation

In order to simulate the PCF access method the libraries contained in GloMoSim had to be modified. It was determined that the modifications to the current code were at a MAC level so only the 802_11.h, 802_11.pc, mac.pc, glomo.pc, and the makefile library files had to be modified aswell as the GloMoSim interface file config.in.

1.6 Document structure

In chapter 2 the IEE 802.11 standard for wireless communication is introduced.
In chapter 3, an introduction to the GloMoSim simulation package which will be followed by chapter 4, which is an explanation of how to implement 802.11 standard in GloMoSim by the use of PARSEC. Chapter 5, is an overview of what was done in order to implement the 802.11 PCF function in GloMoSim followed by chapter 6 which includes the testing and results and chapter 7 is the conclusion. Finally Appendix A contains a list of the software used for this project and Appendix B contains information on the GloMoSim interface.

1.7 Summary

In this chapter a brief explanation of what the 802.11 standard is, its uses and access methods are introduced. Then some of the amendments are mentioned in order to emphasize that mobile communication is becoming increasingly popular and these updates improve the service for modern devices. The basic topologies for communication under 802.11 are covered. And for the simulation of the PCF function the various files that had to be modified are introduced. Also the chapter allocation is outlined.

---

[6] 802.11 standard section 9

Chapter 2 - 802.11 Standard

The 802.11 standard defines the methods for channel allocation, frame formatting, error checking, fragmentation, transmission, momentum, reassembly etc. 802.11 allows for PCF and DCF access methods working in harmony as shown below.

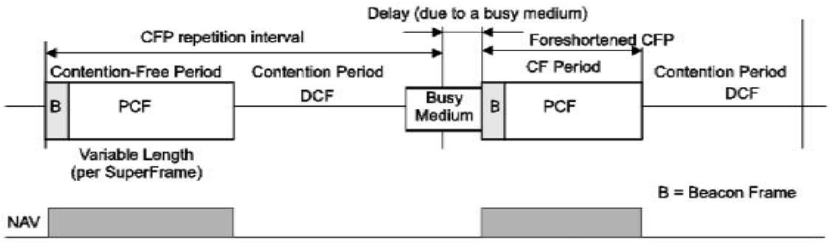


Fig.2.1. 802.11 operation for PCF and DCF access methods in the same BSS[7]

2.1 Time Conventions Fig.2.1

When DCF and PCF access methods are to be used in the same BSS a time is allocated for each of the access methods. The time where the PCF access method is used is called the contention free period and the time where the DCF access method is used is called the Contention period. For 802.11 the PCF access method has access first and then the DCF access method is used. For the two methods to work concurrently the PCF access method is cycled after what' scalled a contention free repetition interval.

2.2 Operation for PCF and DCF access methods in the same BSS

From the diagram shown above in Fig.2.1 the PCF access method is used during the Contention Free Period and the DCF access method is used during the Contention period and this alternates every Contention free period repetition interval. On occasion at the end of the Contention period the medium could still be in use, this has the effect of shortening the Contention free period therefore, at the start of the Contention Free Period a beacon is transmitted which contains timing information as well as other data which will be dealt with later.

[7] Similar diagram in 802.11 standard section 9.3.1

2.3 DCF Access Method

The DCF functionality will now be introduced by first introducing a mechanism for fairness of use of the medium and then how the method for setting and resetting the network allocation vector (NAV) is implemented and finally the various frames used for this access method.

2.3.1 Timers Used[8]

The time interval between frames is called an interframe space (IFS).

Short interframe space (SIFS) .

Point coordinator interframe space (PIFS).

Distributed coordination function interframe space (DIFS).

Extended interframe space (EIFS).

Time unit (TU) which is a measurement of time equal to 1024us.

SIFS = aRxRFDelay + aRxPLCPDelay + aMACProcessingDelay + aRxTxTurnaroundTime.

SlotTime = aCCATime + aRxTxTurnaroundTime + aAirPropagationTime + aMACProcessingDelay.

PIFS = aSIFSTime + aSlotTime

DIFS = SIFSTime + 2 x SlotTime

The EIFS is derived from the SIFS and the DIFS and the length of time it takes to transmit an ACK Control frame at 1 Mbit/s by the following equation:

EIFS = SIFSTime + (8 x ACKSize) + aPreambleLength + aPLCPHeaderLngth+ DIFS.

Backoff time = random integer * SlotTime[9]

where the random integer is determined by by the contention window (CW).

$$CW = 2^{(2+retrycount)} - 1$$

[8] 802.11 standard section 9.2.10

[9] 802.11 standard section 9.2.4

CWmin <= CW <= Cwmax.

The value for the contention window minimum value (Cwmin) is 31 and the contention window maximum value (Cwmax) is 1023 for direct spread spectrum transmission.

2.3.2 Network allocation vector (NAV)

STAs receiving a valid frame update their NAV with the information received in the Duration/ID field, but only when the new NAV value is greater than the current NAV value and only when the frame is not addressed to the receiving STA.

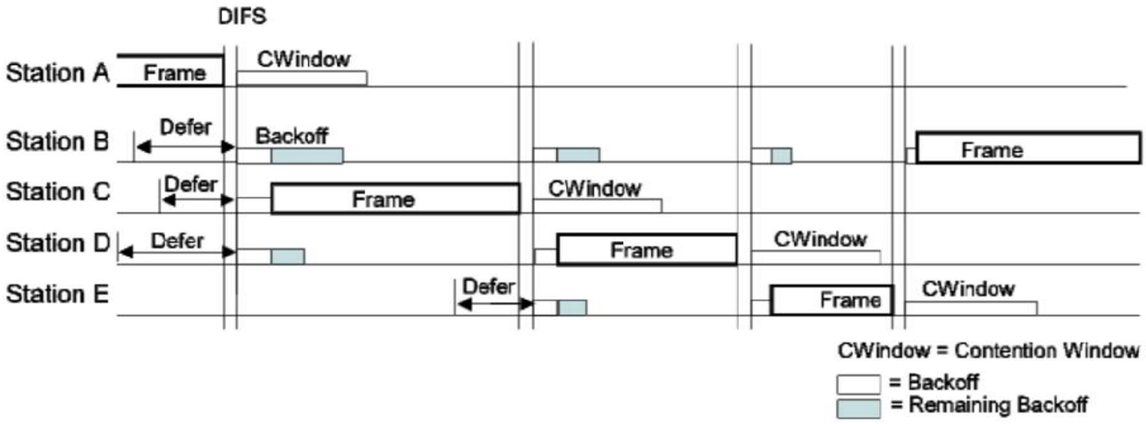


Fig.2.2.medium access procedure.[10]

2.3.3 Medium Access Procedure

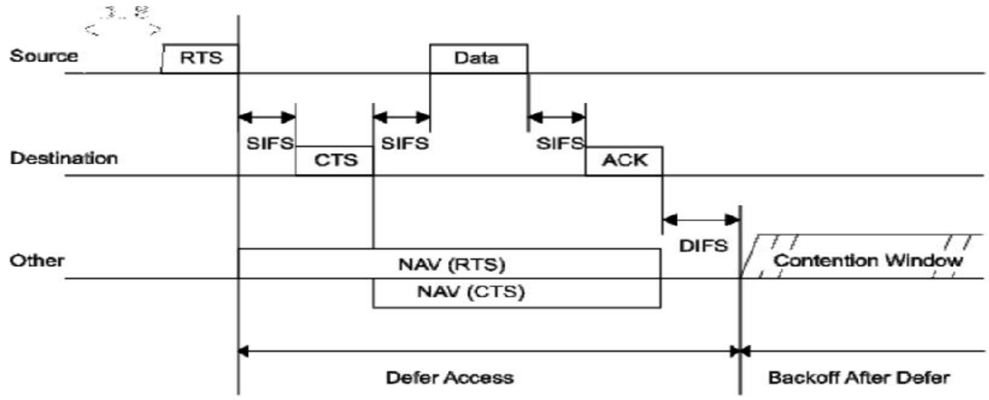
From the diagram above in Fig.2.2 there are five stations trying to gain access to the medium simultaneously. Firstly, each station senses the medium, waits for a DIFS time, waits a further random backoff time and then the station with the smallest backoff time which when it reaches zero gains access to the medium. The backoff time once the medium is deemed busy is stored and the next time the medium is idle the stations again wait a DIFS, then decrements the backoff time, the one with the smallest backoff gets access to the medium and by this algorithm collisions are minimizes during contention between multiple STAs that have been deferring to the same event[11].

[10] 802.11 standard section 9.2.5.2
[11] 802.11 standard section 9.2.4

2.3.4 Retry Counters

The contention window (CW) parameter takes an initial value of CWmin. Every STA maintains a short retry count (SRC) as well as a long retry count (LRC), both of which take an initial value of zero. The SRC is incremented whenever a timeout on an Ack frame is detected. The LRC is incremented whenever a timeout on a data frame is detected. The CW takes the next value in the series every time an unsuccessful attempt to transmit a frame is detected until the CW reaches the value of CWmax. Once CW reaches CWmax, the CW remains at the value of CWmax until it is reset. The CW is reset to CWmin after every successful attempt to transmit a frame, when the LRC reaches the LongRetryLimit, or when SRC reaches ShortRetryLimit. The SSRC shall be reset to 0 whenever a CTS frame is received in response to an RTS frame or whenever an ACK frame is received in response to a frame transmission, or whenever a frame with a group address in the Address1 field is transmitted. The LRC is reset to 0 whenever an ACK frame is received in response to transmission of a frame of length greater than RTSThreshold, or whenever a frame with a group address in the Address1 field is transmitted.

Fig.2.3 Frame Exchange Procedure[12].



2.3.5 Frame Exchange Procedure

This method in Fig.2.3 once access to the medium is acquired (2.3.3 as described above) the STA transmits a Request to send (RTS) frame (to its desired destination STA). Once the an STA receives this RTS it sets its NAV. If the receiving STA is the intended destination it sends out a CTS which STAs can use to update their NAV counters. This CTS frame tells the source STA that it is allowed to transmit a data frame and how long to wait for an acknowledgment frame

[12] 802.11 standard section 9.2.5.4

(Ack). If after an EIFS the source has not received an Ack a retransmit is initiated and a retry counter is incremented, but before the STA can retransmit it first has to regain access to the medium and the random integer which is used for backoff has its range altered to ensure longer backoff time than the previous attempt. Once the stations NAV timers have reached zero they sense the medium and try to gain access as explained above.

2.3.6 Frame Types and purposes.

2.3.6.1 General frame Format[13]

The MAC frame format has a set of fields that occur in a fixed order in all frames. Fig 2.4 depicts the general MAC frame format. The fields Address 2, Address 3, Sequence Control, Address 4, and Frame Body are only present in certain frame types.

Octets 2	2	6	6	6	2	6	0-2312	4
Frame Control	Duration ID	Address 1	Address 2	Address 3	Sequence Control	Address 4	Frame Body	FCS

<--->

Mac Header

Fig.2.4 General frame format.

There are several different frame types used for the contention period some of which are;

- ***DATA*** *a data frame which uses a format similar to the general frame format control frames*
- ***ACK*** *an acknowledgment frame*
- ***RTS*** *a request to send frame*
- ***CTS*** *a clear to send frame*

and each will be dealt with in turn because these are important for the implementation of the standard in GloMoSim.

2.3.6.2 Duration ID *(16 bits in length)*

Contains a duration value which is specific to a frame type but generally it contains a value the duration of the previously received frame – time to transmit a frame + SIFS. During the contention free period this is set to 32768.

[13] 802.11 standard section 7.1

2.3.6.3 Address fields (48bit addresses).

Address 1

Destination address which is the address of the intended recipient.

Address 2

Source address which is the address from which the frame was initiated.

Address 3

Receiver address is an individual or group address that identifies the intended immediate recipient STAs, on the wireless medium[14]. This is used when a temporary network is setup within a BSS and it may take several hops before the intended destination is reached.

Address 4

Transmitter address is an individual address that identifies the STA which transmitted onto medium. This is used when a temporary network is setup within a BSS where several hops may be needed before the destination is needed.

2.3.6.4 Sequence Control Field.

This field is subdivided into two fields the sequence number and the fragment number.

- Each packet transmitted from a STA is assigned a sequence number .
  Sequence numbers are assigned from a single modulo 4096 counter, starting at 0 and incrementing by 1 for each frame.
- The Fragment Number field is a field indicating the number of each fragment of a data stream. The fragment number is set to 0 in the first or only fragment of an data stream and is incremented by one for each successive fragment of that data stream. The fragment number remains constant in all retransmissions of the fragment.

2.3.6.5 Frame Body

This is a variable length field (0-2312 octets) and contains data which is specific to the frame types and subtypes.

2.3.6.6 Frame Check Sequence (FCS) Field.

This is a 32bit field containing a 32bit CRC which is used for checking data integrity.

---

[14] 802.11 standard section 7.1.3.3.6

2.3.7 Ack Frame Format

Octets 2	2	6	4
Frame Control	Duration	Receiver address	FCS

<--------------------------------->

Mac Header

Fig.2.5 Ack frame format.

2.3.8 Request to send (RTS) frame format

Octets 2	2	6	6	4
Frame Control	Duration	Receiver address	Transmitter address	FCS

<--->

Mac Header

Fig.2.6.RTS frame format

2.3.9 Clear to Send (CTS) Frame

Octets 2	2	6	4
Frame Control	Duration	Receiver address	FCS

<-------------------------------->

MAC Header

Fig.2.7. CTS frame format

2.3.10 Data Frame

Octets 2	2	6	6	6	2	0-2312	4
Frame Control	Duration	Address 1	Address 2	Address 3	Sequence control	Frame body	FCS

<--->

MAC Header

Fig.2.8.Data frame format

2.4 PCF Access Method

Firstly for this project it will be assumed that the STA used as the access point (Ap) knows how many STAs are in the network and it sets up a polling list and performs the job of a polling master.For this part an introduction to the frame types first, will help later in the development of the mechanism for contention free operation.

2.4.1 Timers Used[15]

The value of CFPMaxDuration is limited to allow coexistence between contention and contention-free traffic. Once this time is reached the PCF access method stops and then the DCF access method is used for PCF and DCF access methods operating in the same BSS.

*CFPMaxDuration = (BeaconPeriod*DTIMPeriod*CFPRate)*
*-(MaxMPDUTime+2*SIFSTime+2*SlotTime+8*ACKSize)*

which is expressed in microseconds. Where

MaxMPDUTime = TimeToTxMaxSizedMACFrame
+TimeToTx(Preamble+header+trailer+expansionBits)

This allows sufficient time to send at least one data frame during the CP.

The minimum value of the CFPMaxDuration is MinCFPMaxDuration and should allow sufficient time for the AP to send one data frame to a STA, while polling that STA, and for the polled STA to respond with one data frame.

*MinCFPMaxDuration = 2*MaxMPDUTime+TimeForBeaconFrame+TimeForCF-EndFrame.*

MinimumDataTransferTime = TimeToTxOneDataFrameToSTA
+TimeForSTAToRespondWithOneDataFrame

15 802.11 standard section 9.3.3.3

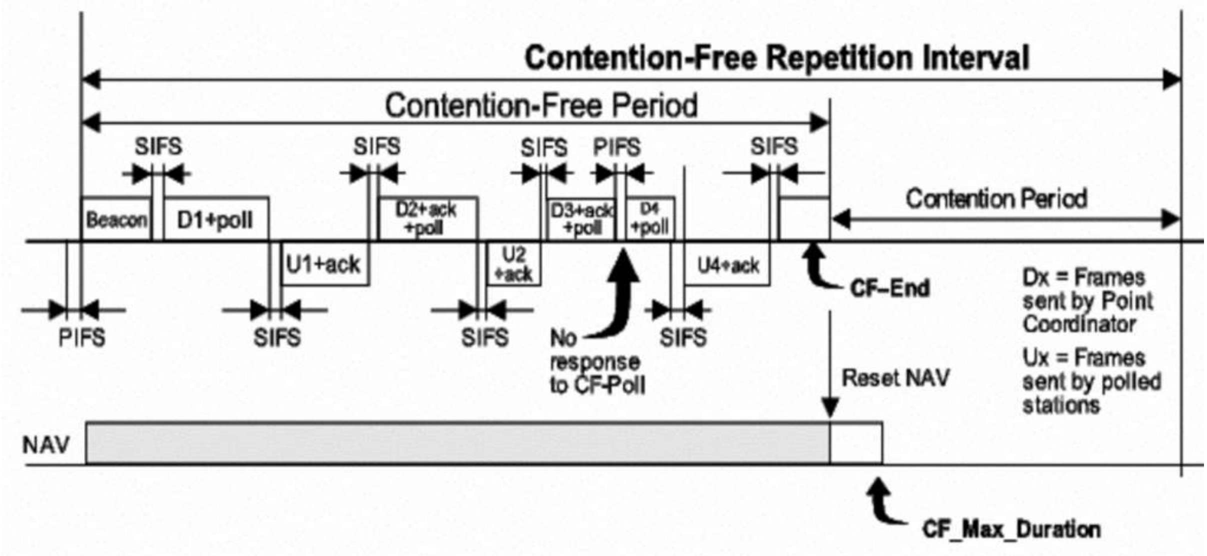


Fig.2.9.CFP mechanism[16] .

2.4.2 Medium Access Procedure

From the diagram in Fig.2.10 the mechanism for contention free medium access is shown which starts with the Ap sensing the medium and if the medium is idle the Ap waits a further PIFS period, then the Ap transmits a Beacon frame which contains information like the CFPMaxDuration which is measured in TU and the CFPDuration remaining which is also measured in TU. These duration fields are used by the STAs to set their NAV timers.

The Ap waits a further SIFS and if there is only MinimumDataTransferTime or less left a CF-End frame is sent and this signifies the end of the CFP. If there is more than the MinimumDataTransferTime then the Ap goes through its polling list and if an STA is determined to be next in the list the Ap checks the buffer and if data for that STA is found the Ap sends a Data+CF-Poll. If there is no data found a CF-Poll is sent. After this the polled STA is allowed a SIFS + timeToTxDataFrame interval to respond under normal conditions but, if after a further PIFS interval there is no response the Ap takes control and moves onto the next STA in the polling list. The next time the STA which didn't respond is polled, if data was sent to it initially then this data frame will be resent to it or if an acknowledgment for a frame received was sent this will be resent.

An STA wishing to respond to one of the polls mentioned above if it has data to send responds

[16] 802.11 standard section 9.3.3

with a Data+CF-Ack frame and if it has no data to send it responds with a CF-Ack frame. The Ap on reception of this information will if time allows poll the next STA in the polling list, if there is data for the STA it will send a Data+CF-Ack+CF-Poll frame, if there is no data a CF-Ack+CF-Poll frame will be sent and this procedure will continue until there is just MinimumDataTransferTime left on which a CF-End+CF-Ack frame will be sent if the Ap has received a data frame from an STA or it can send a CF-End frame to announce the end of the CFP and all NAVs will be set to null. It is also possible for the Apnode to end the CFP at any time by the use of CF-End or CF-End+Ack (which ever is relevant).

Another complication to the CFP is if a STA to STA communication within the same BSS is needed, for this a STA is polled as normal, sends its frame directly to another STA, the destination STA responds after a SIFS with an Ack and after a further PIFS the Apnode takes back control and continues as before.

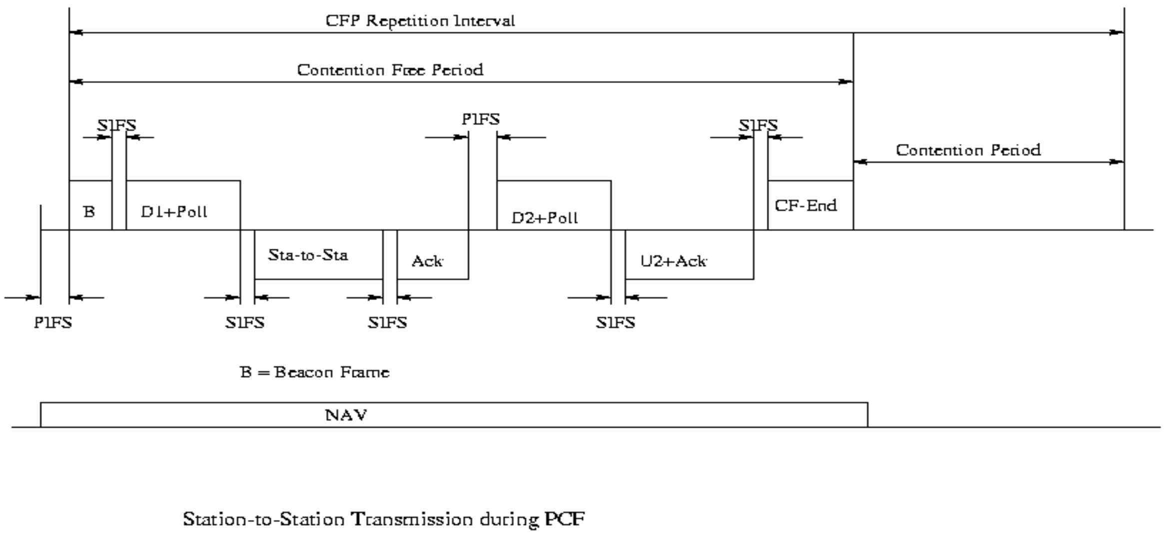


Fig.2.10.Station to station transmission during contention free period

2.4.3 Frames Types Used in CFP

Frames Types	***Purpose***
Beacon Frames	Used by Ap during the contention free period to announce the Ap address along with information for setting the NAVs of STAs in range.
Control frames	
***CF-End* frame**	used by Ap to announce the end of the CFP.
CF-End+CF-Ack frame	used by Ap to announce the end of the CFP and to piggyback an Acknowledgment to the previous frame received.
Data frames	
Data	Used when STA is polled but communicates with another STA.
Ack	Used when an STA receives data from another STA.
Data+CF-Ack frame	used by STA to send a data frame and to piggyback an Acknowledgment to previous frame received.
CF-Ack frame	used by STA to Acknowledge a previous frame received.
CF-Poll frame	used by Ap to poll an STA in the polling list.
Data+CF-Poll frame	used by Ap to poll an STA in the polling list and to piggyback a data frame for that STA.
CF-Ack+CF-Poll frame	used by Ap to poll an STA in the polling list and to piggyback an Acknowledgment to the previous frame received.
Data+CF-Ack+CF-Poll frame	used by Ap to poll an STA in the polling list and to piggyback a data frame for that STA and to Acknowledge the previous frame received.

Table 2.1 Frames used in CFP

2.4.3.1 Management Beacon frame.

Octets 2	***2***	***6***	***6***	***6***	***2***	***0-2312***	***4***
Frame Control	Duration	Destination address	Source address	BSSID	Sequence control	Frame body	FCS

<-->

MAC Header

Fig.2.11.General management frame.

During the CFP the duration field is set to 32768.[17]

[17] 802.11 standard section 7.1.3.2

For beacon frames the Frame body field is subdivided into the following fields and if an STA receives information in a field of which it does not understand it ignores it. **Note** that the order is important and those fields deemed not applicable are still included which will be ignored.

Table.2.2.Frame body of beacon management frame.

Order	***Information***	***Octets***	***Notes***
1	Timestamp	8	Timer
2	Beacon interval	2	Number in TU between between beacons
3	Capabitity information	2	Indicates requested or advertised capabilities
4	SSID	0 to32	Used to identify IBSS
5	Supported rates	1 to 8	Used to announce rates where each octet stands for a rate in units of 500Kbits/s
6	FH parameter set	5	Used when frequency hopping spread spectrum is used
7	DS parameter set	1	Used if direct spread spectrum is used and contains channel number
8	CF parameter set	6	Has subfields explained below which hold information like CFPMaxDuration and CFPDurationRemaining
9	IBSS parameter set	2	If set to 0 power management is not used
10	TIM	1 to 254	traffic indication map

2.4.3.1.1 The CF parameter of the beacon frame subfield[18] *Table.2.2*

The CFPCount is a count of how many DTIM intervals
(including the current one) before the next CFP start. If this is 0 it signifies the start of the CFP.
The CFPPeriod is a count of the DTIM intervals between the start of CFPs.
The CFPMaxDuration is a count in TU for the maximum CFP duration and is used for Aps to set their NAVs.
The CFPDurationRemaining is a count in TU remaining in the CFP and is used by STAs to update their NAVs.

2.4.3.2 Control CF-End frame, CF-End+CF-Ack frame

Octets 2	***2***	***6***	***6***	***4***
Frame Control	Duration	Receiver address	BSSID	FCS

<-->

MAC Header

Fig.2.13. CF-End frame, CF-End+CF-Ack

18 802.11 standard section 7.3.2.5

2.4.3.3 Data Frame Format

Octets 2	***2***	***6***	***6***	***6***	***2***	***0-2312***	***4***
Frame Control	Duration	Address 1	Address 2	Address 3	Sequence control	Frame body	FCS

<-->

MAC Header

Fig.2.12.Data frame format

where

- ***Address 1*** is used for the intended destination address.
- ***Address 2*** is used for the transmitter address.
- ***Address 3*** is used for the immediate recipients address this can be used via the Apnode to a destination elsewhere.

For CF-Ack, CF-Poll and CF-Ack+CF-Poll frames the frame body is set to null.

For Data+CF-Ack, Data+CF-Poll and Data+CF-Ack+CF-Poll frames types the full frame format shown in Fig.2.12. is used.

2.5 Summary

This chapter introduced the PCF and DCF access methods and the various frame used for these access methods. It also covers the various timers used and the method by which DCF and PCF can coexist within the same BSS. This coverage is extremely important for the understanding of what has to be added to the GloMoSim libraries.

Chapter 3 - GloMoSim Introduction

GloMoSim (Global Mobile Information System Simulation) is a network simulation package used for wireless simulation. It was designed in order to simulate various large scale wireless networks with fixed or variable mobility under a range of conditions. GloMoSim was developed by the Parallel Computing Laboratory in UCLA by the use of a C like platform PARSEC which is used for sequential and parallel execution of discrete event simulation models.

GloMoSim was designed by the use of a layered approach based on the OSI reference model. For each of these layers models can be added easily as long as the messages between each of the layers are the same (the various layers will be covered later).

3.1 GloMoSims Layered Architecture

GloMoSim was designed under a layered philosophy similar to the 7 layer OSI reference model. The difference being that with GloMoSim only 5 layers are used as shown in Table.3.1 Note that as long as the messages between each of these layers are the same, a model for a particular layer can be added or altered without affecting the rest of the layers.

Table.3.1. Layers and models present for each layer in GloMoSim

Layer	***Models***
Application	Telnet, FTP, Replicated file system, CBR, Web caching, Netmeeting, Webphone, Synthetic traffic generators.
Transport	TCP/IP, UDP, DBS satellite models.
Network	ODMRP, CAMP, AMRIS, AMRoute, AST, DVMRP, Fisheye, Flooding, OSPF, DSR, WRP, Bellman-ford, NS-DSDV, LAR, DREAM, MMWN.
MAC	CSMA, MACA, MACAW, FAMA, 802.11DCF, TSMA.
Radio	Analytical(Freespace, Rayleigh, Ricean), SIRCIM, 2-Ray ground reflection model, DSS, path loss trace files.

3.2 GloMoSim Interface

GloMoSim uses a text file based interface, in which all options are present and the user chooses certain set up simulation parameters and comments out by the use of the # key parameters which are not applicable to the simulation. This set up file is in the */bin* directory and is called *config.in*. Some of the options available to this file are now going to be introduced[19]. The setup file is used for simulation setup for the various layers *Table.3.1.* For the application layer a further file is needed *app.conf* which is used to setup the various applications for instance FTP, TELNET etc. In this file the application of choice is added and the various parameters for this application also have to be added. Also several applications can be added for a simulation.

3.3 Additions made to the GloMoSim Interface file

In order to simulate the PCF access method the following parameters were added to the *config.in* file.

- MAC-PROTOCOL 802.11new
- Apnode 0
- BeaconInterval 10S :This is the repetition interval of the beacon frame
- CFPRepetionInterval 10S :This is the time which when passed the contention free period will begin again.

3.4 Summary

In this chapter GloMoSim was introduced by firstly explaining what it is, then an explanation of the various applications available at the various layers was introduced. A general introduction to the interface was covered and then the modifications which had to be made to the setup file in order to simulate the PCF access method were covered.

[19] Appendix B

Chapter 4 - GloMoSim Implementation of the 802.11 DCF function.

For this chapter the medium access method used in GloMoSim is introduced. Then the frame sequence which is followed by the DCF access method and then the various structures which are used in order to simulate the various frame types. This is important because in the next chapter most of this has to be modified in order for the PCF functionality to be added and an understanding of what is already there will help in this implementation. Also note that the language used by GloMoSim is PARSEC which is a C like language.

4.1 Medium Access Method

From before[20] in order for an STA to gain access to the medium it would have to go through the medium access procedure. This requires waiting for a DIFS or EIFS (state waitForDIFSorEIFS) and a backoff timer.

4.2 Frame Sequence

When a packet is wished to be sent first a RTS is sent then the destination replies with a CTS and once this is done the source sends a data frame and the destination replies with an Ack and the packet is removed from the buffer.

4.3 Explanation of the DCF access method[21]

A basic flow chart to explain the main mechanism for the DCF access method is shown in Fig.4.1. This chart goes through a typical session where an STA wants to send data on a peer to peer basis to another STA. The main issues for understanding this implementation is the states which are used by a timeout mechanism. Once the medium access method is finished when a RTS is sent the state of the STA would be set to TxRTS and a timer would be set to waitForCTS. If the medium is busy the waitForCTS timer will timeout and a retransmit is initiated. The medium access method is again initiated and once contention is gained the frame is retransmitted

[20] Section 2.3.2
[21] Section 2.3

(source state TxRTS timer waitForCTS used above). Once the destination receives the RTS it will send a CTS to source (destination state TxCTS and a timer is set waitForData) to the source and on reception of this a data frame is sent (source state TxData timer waitForAck). If the destination waitForData timer times out it will simply check to see if there is a packet from the application to be sent and if so goes back through the medium access procedure and frame sequence for its packet. If the source waitForAck timer expires a retransmit is initiated. If the Ack is received the packet is removed from the buffer.

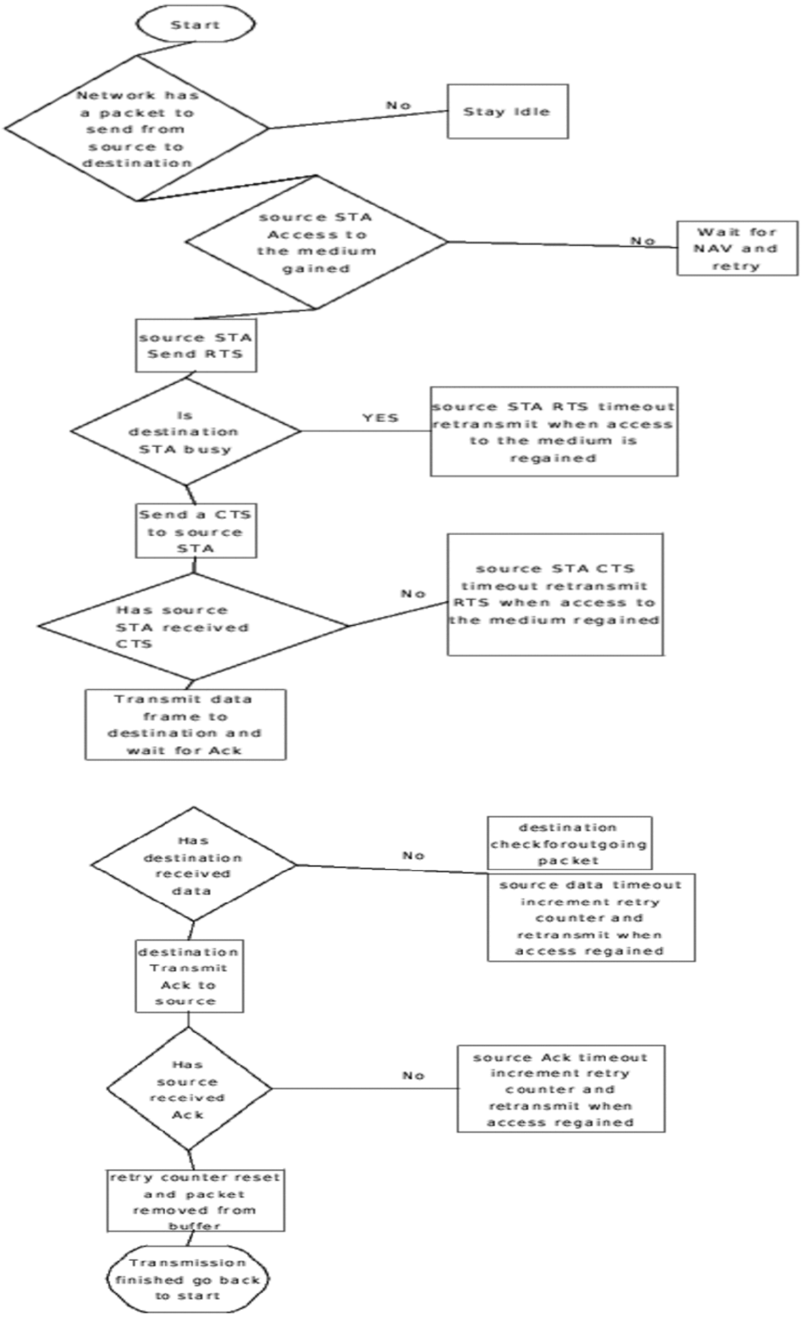


Fig4.1 Basic flow chart for the DCF access method.

4.4 Frame Types Used

RTS, CTS, Data, Ack

where CTS and Ack frames have a short control frame structure for the frames. RTS frames use a long control frame structure. Data frames use a frame header structure which is nested in the MAC frame structure.

4.5 Summary

In this chapter the DCF access method implemented in the libraries of GloMoSim is covered which introduces the basic concepts behind the mechanism by which it works. It also introduces some of the structures used for the various frames. This is vital to the understanding of how the DCF access method is coded and is the basic building block before the PCF access method can be added.

Chapter 5 - Modifications made to implement 802.11 PCF function

For this chapter the assumption will be first introduced then the medium access method which is followed by the various frame sequences. The the basic mechanism for the PCF access method is introduced. Then the structures used for the frame types is introduced. Then the states added for the PCF access method is mentioned folled by the frame types used.

5.1 Assumptions made

- •The access point knows how many STAs are in the system.
- •Only one BSS used.
- •Association and other complicated functionality is not covered.
- •During contention free period direct STA to STA communication is not covered.

5.2 Medium Access Method

At the start of the contention free period a beacon is sent out and after TimeToTxBeacon + SIFS the access point AP takes on the job of a polling master.

5.3 Frame Sequence

The following are the frame sequences used for this project. A Beacon is sent out first and then after some time a poll or a poll+data is sent out the destination STA responds with an Ack or a data+Ack. If the time left reaches certain value a end frame is sent.

5.4 Explanation of the PCF Access Method

From Fig.5.1. Once the CFP has begun the Ap senses the medium and if idle waits a PIFS (timer waitForPifsand) by setting this timer the normal timeout method is used. Then a beacon is sent out. Then the Ap after timeToTxBeacon+SIFS transmits either a poll or a poll+data. If a poll is sent then the state of the Ap is set to TxPoll and the timer is set to waitForAck and if a response is not received within this time the Ap will poll the next STA. On reception of this poll the STA checks the buffer for a packet and if there is none an Ack is sent (state TxAck). If the STA has data to sent then a Data+Ack will be sent. When the Ap receives a Data+Ack (state TxDataAck) it will send a piggybacked Ack in the next frame. If the Ap has a packet to send to a polling STA next in the list a poll+data frame is sent (state TxPollData timer waitForAck). If the Ap detects that the time left is below minDataTx + timeToTxEnd than an end frame is sent.

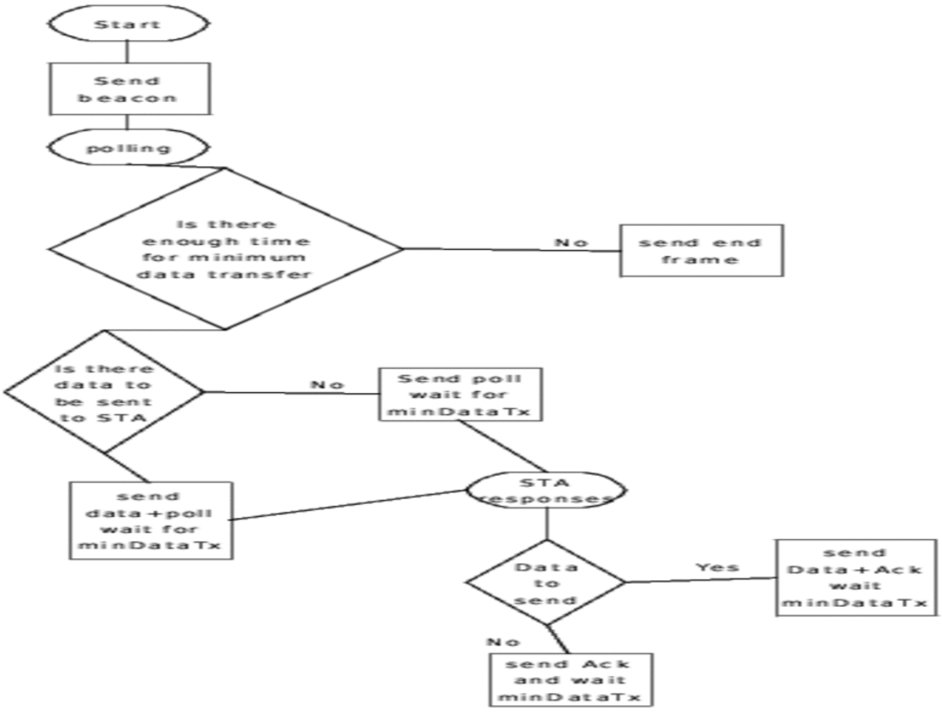


Fig.5.1. Basic flow chart for PCF access method

5.5 Frame Types Used

beacon, poll, poll+data, ack, data+ack, end.

Where for the beacon a new structure was added. For Ack, End frames the short control frame structure already there was used. For poll frames the long control structure was used. For poll+data and data+ack frames the current data frame structure was used.

5.6 States added for contention free operation.

Waiting for response states

M802_11new_S_Apnode_WFPIFS

M802_11new_S_WF_CF_ACK

M802_11new_S_Apnode_WFNAV :not functional

Transmitting states

M802_11new_X_CF_Beacon

M802_11new_X_CF_Poll

M802_11new_X_CF_Poll_Ack :not functional

M802_11new_X_CF_Poll_Ack_Data :not functional

M802_11new_X_CF_Poll_Data

M802_11new_X_CF_End_Ack :not functional

M802_11new_X_CF_End

M802_11new_X_CF_Ack

M802_11new_X_CF_Ack_Data

5.7 Frame types added for contention free operation.

M802_11new_BEACON

M802_11new_CF_POLL

M802_11new_CF_END_ACK :functionality to be added

M802_11new_CF_POLL_DATA

M802_11new_CF_POLL_ACK

M802_11new_CF_POLL_ACK_DATA :functionality to be added

M802_11new_CF_ACK

M802_11new_DATA_CF_ACK

M802_11new_CF_END

M802_11new_CF_END_ACK :functionality to be added

5.8 Summary

In this chapter the PCF access method is introduced along with the various frame types added. The various frame sequences are covered and an explanation of the states and timers are covered which give an a good understanding of the PCF access method.

Chapter 6 - Results and testing

For this project the main method for testing and debugging was to put print to screen commands inside every function and loop and then by the use of an X-term to scroll through this screen dump typical information of interest would be function, state, node, and if inside a loop or in sub-function to indicate a the migration to this. Also later in the project the GloMoSim statistical output was used.

Chapter 7 - Conclusions and Future Work

In this project the 802.11 standard was introduced with both PCF and DCF access methods. The implementation of the PCF and DCF access methods in the GloMoSim libraries was also covered along with the main concepts of how they work. In this project certain functionality was added to the libraries. It is possible to set the beacon interval along with the CFP repetition interval and my code does allow some DCF and PCF functionality to work in harmony.

This project has put in the ground work for further development. Although it does not function correctly because of a buffering problem it can be run from GloMoSim and some results can be acquired. The reasons for this being that time ran out but it is hoped that the work can be completed at a later stage.

The following could be changed

- An STA could sense that the time left is not enough for minimum data transfer and reply with just an Ack. This can be added easily.
- The data+poll+ack, end+ack frames can be added by modifying the polling function.

Appendix A

For this project the platform for development was Mandrake Linux 9.2 and the editor of choice was Kwrite which was chosen for its ease of use and also because of its highlighting modes. An X terminal was also used which was Konsole 1.2.3 by Lars Doelle which is quiet handy because a lot of the debugging done for this project was print to screen and with this X-term it allows copy and paste and easy scrolling through data by the use of the cursor keys where the output is held in a history list which can be easily cleared before a rerun of code output.

Appendix B

GloMoSim Interface

GloMoSim uses a text file based interface, in which all options are present and the user chooses certain set up simulation parameters and comments out by the use of the # key parameters which are not applicable to the simulation. This set up file is in the */bin* directory and is called *config.in.* Some of the options available to this file are now going to be introduced.

SIMULATION – TIME, this can be set to a value ranging from 100ns to 100 days

TERRAIN – DIMENSIONS (x, y), these values of x and y determine the area in metres of which is applicable to this simulation.

NUMBER – OF – NODES, this number is the number of nodes used for the simulation.

Node placement has to be covered and several options are available;

NODE – PLACEMENT FILE, this uses a file of values for nodes within the terrain dimensions.

NODE – PLACEMENT – FILE, ./nodes.input this is the file used if file placement is used

or

NODE – PLACEMENT GRID, where nodes are placed in a grid separated by a GRID-UNIT

GRID – UNIT, this is used if GRID placement is used to separate nodes but the number of nodes has to be a square of an integer.

or

NODE – PLACEMENT RANDOM, where nodes are placed randomly within terrain dimensions.

or

NODE – PLACEMENT UNIFORM, if this is enabled the terrain is divided into a number of cells and a node is placed in each cell.

For node mobility there are several different options available;

MOBILITY NONE, this is used where the nodes are stationary.

or

MOBILITY RANDOM-WAYPOINT, this is where the node is moving from one place to another where the destination is randomly chosen.

MOBILITY-WP-PAUSE, is dwell time, (the amount of time spent whilst waiting to move on to the next point).

MOBILITY-WP-MIN-SPEED, used for the minimum speed of node.

MOBILITY-WP-MAX-SPEED, used for the maximum speed of node.

or

There is also a trace option,

MOBILITY TRACE, this is used when a file is utilized for mobility.

MOBILITY-TRACE-FILE, this is the file used for trace mobility data.

For Propagation Modelling

PROPAGATION-LIMIT, measured in dBm and this value of power is used to determine if the signal will be delivered or lost.

Pathloss Models

PROPAGATION-PATHLOSS FREE-SPACE, friss free space model syntax (pathloss exponent, sigma) = (2.0,0.0).

or

PROPAGATION-PATHLOSS TWO-RAY, uses free space pathloss (2.0, 0.0) for near sight and for plain earth pathloss(4,0) for far sight also note that the antenna height is 1.5m.

The noise figure and temperature can also be set.

The following are radio related settings;

RADIO-TYPE RADIO-ACCNOISE, standard radio model.

or

RADIO-TYPE RADIO-NONOISE, abstract radio model.

RADIO FREQUENCY, frequency in Hertz.

RADIO BANDWIDTH, measured in bits per second.

RADIO-RX-TYPE SNR-BOUNDED, if signal SNR is more than RADIO-RX-SNR-THRESHOLD it receives the signal, otherwise it is dropped.

RADIO-RX-SNR-THRESHOLD, a value in dB used for receive decisions.

or

RADIO-RX-TYPE BER-BASED, if used uses a BER-TABLE-FILE for decisions

BER-TABLE-FILE ./ber_bpsk.in

RADIO-TX-POWER, a value in dBm

RADIO-ANTENNA-GAIN, a value in dB

RADIO-RX-SENSITIVITY, a value in dBm

RADIO-RX-THRESHOLD, value in dBm which is the minimum power for the received packet.

There are Various MAC Protocols;

MAC-PROTOCOL 802.11new

Apnode :integer value starting at zero used with 802.11new

or

MAC-PROTOCOL 802.11.
or
MAC-PROTOCOL CSMA
or
MAC-PROTOCOL MACA.
or
MAC-PROTOCOL TSMA

For the Network Protocol.

For this Version of GloMoSim there is just one model;

NETWORK-PROTOCOL IP

There is a Vast Array of Routing Protocols;

ROUTING-PROTOCOL BELLMANFORD

or

ROUTING-PROTOCOL AODV
or
ROUTING-PROTOCOL DSR
or
ROUTING-PROTOCOL LAR1

or
ROUTING-PROTOCOL WRP
or
ROUTING-PROTOCOL FISHEYE
or
ROUTING-PROTOCOL ZRP
or
ROUTING-PROTOCOL STATIC
STATIC-ROUTE-FILE ROUTES.IN, this contains routing information for static routing.

To setup applications like FTP/GENERIC, TELNET, CBR, and HTTP a separate file is needed app.conf.

In this file a clean distinction between FTP and FTP/Generic is made. The difference is basically that for FTP/Generic the client simply sends the data items to the server without the server sending any control information back to the client.

Syntax for FTP Usage.
FTP <source> <destination> <items to send> <start time>

<source> is the client node.
<destination> is the server node.
<items to send> is how many application layer items to send.
<start time> is when to start FTP during the simulation.
for example

a) FTP 0 1 20 0S

Node 0 sends node 1 twenty items at the start of the simulation,
with the size of each item randomly determined by tcplib.

b) FTP 0 1 0 200S

Node 0 sends node 1 the number of items randomly picked by tcplib after 200 seconds into the simulation. The size of each item is also randomly determined by tcplib.

FTP/Generic syntax

FTP/GENERIC <source> <destination> <items to send> <item size> <start time> <end time>

<source> is the client node.
<destination> is the server node.
<items to send> is how many application layer items to send.
<item size> is size of each application layer item.
<start time> is when to start FTP/Generic during the simulation.
<end time> is when to terminate FTP/Generic during the simulation.

FTP/Generic examples

a) FTP/GENERIC 0 1 20 1500 0S 500S

Node 0 sends node 1 twenty items of 1500B each at the start of the simulation up to 500 seconds into the simulation. If the twenty items are sent before 500 seconds elapsed, no other items are sent.

b) FTP/GENERIC 0 1 20 1500 0S 0S

Node 0 sends node 1 twenty items of 1500B each at the start of the simulation until the end of the simulation. If the twenty items are sent the simulation ends, no other items are sent.

c) FTP/GENERIC 0 1 0 1500 0S 0S

Node 0 continuously sends node 1 items of 1500B each at the start of the simulation until the end of the simulation.

Syntax for Telnet usage

TELNET <source> destination> <session duration> <start time>

<source> is the client node.

<destination> is the server node.

<session duration> is how long the telnet session will last.

<start time> is when to start TELNET during the simulation.

examples

a) TELNET 0 1 200S 0S

Node 0 sends node 1 telnet traffic for a duration of 200 seconds at the start of the simulation.

b) TELNET 0 1 0S 0S

Node 0 sends node 1 telnet traffic for a duration randomly determined by tcplib at the start of the simulation.

Syntax for CBR usage

CBR <source> <destination> <items to send> <item size><interval> <start time> <end time>

<source> is the client node.

<destination> is the server node.

<items to send> is how many application layer items to send.

<item size> is size of each application layer item.

<interval> is the inter-departure time between the application layer items.

<start time> is when to start CBR during the simulation.

<end time> is when to terminate CBR during the simulation.

Examples

a) CBR 0 1 10 1500 1S 0S 500S

Node 0 sends node 1 ten items of 1500B each at the start of the simulation up to 500 seconds into the simulation. The inter-departure time for each item is 1 second. If the ten items are sent before 500 seconds elapsed, no other items are sent.

b) CBR 0 1 0 1500 1S 0S 500S

Node 0 continuously sends node 1 items of 1500B each at the start of the simulation up to 500 seconds into the simulation.
The inter-departure time for each item is 1 second.

c) CBR 0 1 0 1500 1S 0S 0S

Node 0 continuously sends node 1 items of 1500B each at the start of the simulation up to the end of the simulation.
The inter-departure time for each item is 1 second.

Syntax for HTTP usage which simulates single-TCP connection web servers and clients.

HTTPD <address>
where
<address> is the node address of a node which will be serving Web pages.

For HTTP clients, the following format is used:

HTTP <address> <num_of_server> <server_1> ... <server_n> <start> <thresh>

<address> is the node address of the node on which this client resides
<num_of_server> is the number of server addresses which will follow
<server_1>
to

<server_n> are the node addresses of the servers which this client will choose between when requesting pages. There must be "HTTPD <address>" lines existing separately for each of these addresses.

<start> is the start time for when the client will begin requesting pages

<thresh> is a ceiling (specified in units of time) on the amount of "think time" that will be allowed for a client. The network-trace based amount of time modulo this thresh-hold is used to determine think time.

Examples

HTTPD 3

HTTPD 4

HTTPD 5

HTTPD 6

HTTP 1 3 2 5 11 20S 180S

There are HTTP servers on nodes 3, 4, 5, and 6. There is an HTTP client on node 1. This client chooses between servers {3, 4, 6} only when requesting web pages. It begins browsing after 20 seconds of simulation time have passed, and will "think" (remain idle) for at most 3 minutes of simulation time, at a time.

At the end of the *config.in* file there is some options for statical output for the various layers which are activated by using “YES” a nd deactivated by using “NO”.

APPLICATION-STATISTICS	YES
TCP-STATISTICS	NO
UDP-STATISTICS	NO
ROUTING-STATISTICS	NO

NETWORK-LAYER-STATISTICS	NO
MAC-LAYER-STATISTICS	NO
RADIO-LAYER-STATISTICS	NO
CHANNEL-LAYER-STATISTICS	NO
MOBILITY-STATISTICS	NO

Once this file is set up, save it and run it by entering at the command prompt in the /bin directory
>glomosim config.in.

This will produce an output file called *glomo.stat* which will contain statistical information.

References

1) Analysis of IEEE 802.11e and application of game models for support of Quality of service in coexisting wireless networks. *By Stefan Mangold. 2003.*

2) Introduction to ANSI/IEEE Std 802.11 1999 Edition.

3) Simulation of IEEE 802.11e in GloMoSim. *By Daragh Kelleher August 2003.*

4) http://pcl.cs.ucla.edu/projects/parsec/manual/

5) http://pcl.cs.ucla.edu/projects/glomosim/GloMoSimManual.html

6) http://www.itr.unisa.edu.au/~sgordon/doc/hybrid-interwork2002.pdf

7) http://pcl.cs.ucla.edu/projects/parsec/

8) http://pcl.cs.ucla.edu/slides/workshop99/Mineo-pw99/sld002.htm

9) http://pcl.cs.ucla.edu/slides/workshop99/Jaytut-pw99/sld001.htm

10) http://pcl.cs.ucla.edu/slides/Parsec99.html

11) http://pcl.cs.ucla.edu/slides/Parsec99.html

12) http://www.sss-mag.com/pdf/80211p.pdf

13) http://standards.ieee.org/getieee802/

14) http://www.cis.ohio-state.edu/~jain/talks/ftp/netsem7/sld006.htm

15) http://www.ece.uvic.ca/~fayez/talks/present.pdf

16) http://www.sss-mag.com/pdf/802_11tut.pdf

17) http://www.cs.tut.fi/kurssit/8305010/luennot/WLAN.pdf

18) http://ipsit.bu.edu/sc546/sc546Fall2002/wireless/Wire