Πανεπιστήμιο Θεσσαλίας
Τμήμα Μηχανικών Η/Υ, Τηλεπικοινωνιών &
Δικτύων

Διπλωματική Εργασία

Θέμα:

«Υλοποίηση αλγορίθμου δυναμικής επιλογής συχνότητας λειτουργίας σε ασύρματες κάρτες με χρήση κώδικα ανοιχτού λογισμικού»

Επιμελητές:

Φιαμέγκος Αδαμάντιος του Βλάσιου

Καζδαρίδης Ιωάννης του Κυριάκου

Επιβλέπων Καθηγητής:

Τασιούλας Λέανδρος

(Καθηγητής)

Συνεπιβλέπων Καθηγητής:

Κουτσόπουλος Ιορδάνης

(Επίκουρος*Καθηγητής)*

2ος Συνεπιβλέπων Καθηγητής:

Κοράκης Αθανάσιος

Βόλος , Σεπτέμβριος 2010

Ευχαριστίες

Ύστερα από μια πορεία πέντε και πλέον ετών στο Τμήμα Μηχανικών Η/Υ, Τηλεπικοινωνιών και Δικτύων του Πανεπιστημίου Θεσσαλίας ολοκληρώνουμε τις προπτυχιακές μας σπουδές με την εκπόνηση της παρούσας διπλωματικής εργασίας.

Θα θέλαμε αρχικά να ευχαριστήσουμε θερμά τον κ. Κοράκη Αθανάσιο , του Τμήματος Μηχανικών Η/Υ, Τηλεπικοινωνιών και Δικτύων, για τις χρήσιμες συμβουλές και υποδείξεις του καθώς και για την υποστήριξη που μας προσέφερε κατά την διάρκεια της φοίτησης μας αλλά και κατά την εκπόνηση της διπλωματικής μας εργασίας.

Ευχαριστούμε επίσης τον επιβλέποντα της εργασίας μας, Καθηγητή του Τμήματος Μηχανικών Η/Υ, Τηλεπικοινωνιών και Δικτύων, κ. Λέανδρο Τασιούλα και τον συνεπιβλέποντα Επίκουρο Καθηγητή του Τμήματος Μηχανικών Η/Υ, Τηλεπικοινωνιών και Δικτύων, κ. Ιορδάνη Κουτσόπουλο για την καθοδήγηση τους.

Από καρδιάς θα θέλαμε να ευχαριστήσουμε τον κ. Στράτο Κερανίδη, υποψήφιο Διδάκτορα του Τμήματος Μηχανικών Η/Υ, Τηλεπικοινωνιών και Δικτύων, για την πολύτιμη συνδρομή του στο προγραμματιστικό και πειραματικό σκέλος της διπλωματικής εργασίας.

Τέλος, ευχαριστούμε θερμά τις οικογένειές μας για την αμέριστη συμπαράσταση που μας παρείχαν όλα αυτά τα χρόνια για την ολοκλήρωση των προπτυχιακών σπουδών μας.

Στις οικογένειες μας

CONTENTS

ABSTRACT

The present Final Project Dissertation - Thesis is a depth study of how IEEE 802.11 works.

The first part of the project consists of an introduction in IEEE80211, a presentation of 802.11 network architecture and the possible 802.11 operating modes , a detailed analysis of the IEEE802.11 frame format. Moreover , we mention the 80211 Layer description and we analyze the MAC protocols that are used and the way that communication happens.

Then we present some various phases of 802.11 MAC layer and we focus on the DFS (Dynamic Frequency Selection) mechanism and how is implemented in MAD-WiFi open source driver .Moreover we supplied some information for the MAD-WiFi open source driver and we briefly describe the behavior of the mobile nodes (AP , STA) operations according to the driver. Finally, we mention the overlapping channel problem and how RSSI is used in MAD-WiFi.

In the second part, we suggest a DFS protocol, through which messages containing the information (RSSI measurements) are passed from the associated and contending stations to AP ,to support channel switching decisions for the quality of the link. We implement the proposed mechanism using the MAD-WiFi open source driver and moreover show through experiments in a wireless testbed that it significantly improves user performance in real conditions.

1. Introduction

1.1 IEEE 802.11

IEEE 802.11 is a standard for "wireless connectivity for fixed, portable, and moving stations within a local area" . IEEE 802.11 applies at the lowest two layers of the Open System Interconnection (OSI) protocol stack, namely the physical layer and the data link layer. The physical layer standard specifies the signaling techniques used and the implementation of media specific functions. The data link layer defines the frame transmission structure for control, data and management messages and the architecture for data transmission across a WLAN. Key notions are those of association, which is the mapping of wireless clients to a wireless access point, and of service set, which is a set of co-ordinated wireless clients that can be regarded as analogous to a wired network segment.

Included in the IEEE 802.11 standard is the requirement to provide for privacy of data transmission across wireless networks. The way that privacy is provided is by use of the WEP . WEP is a protocol that uses RSA's RC4 data stream encryption and CRC-32 integrity checking of data frames at the data link layer. WEP is usually implemented in the hardware and firmware of the wireless network interface cards. It is worth stressing that vulnerabilities in WEP will require a new generation of wireless interface cards.

2. 802.11 Network Architecture

A Wi-Fi network has access points (AP) to which stations get associated. Data transfer between stations take place through the AP. An AP and its associated mobile stations form a basic service set (BSS).

2.1. Wireless stations

Wireless stations fall into one of two categories: access points and clients. Access points (APs), normally routers, are base stations for the wireless network. They transmit and receive radio frequencies for wireless enabled devices to communicate with. Wireless clients can be mobile devices such as laptops, personal digital assistants , IP phones , or fixed devices such as desktops and workstations that are equipped with a wireless network interface.

2.2. Basic service set (bss)

The basic service set (BSS) is a set of all stations that can communicate with each other. There are two types of BSS: Independent BSS (also referred to as IBSS), and infrastructure

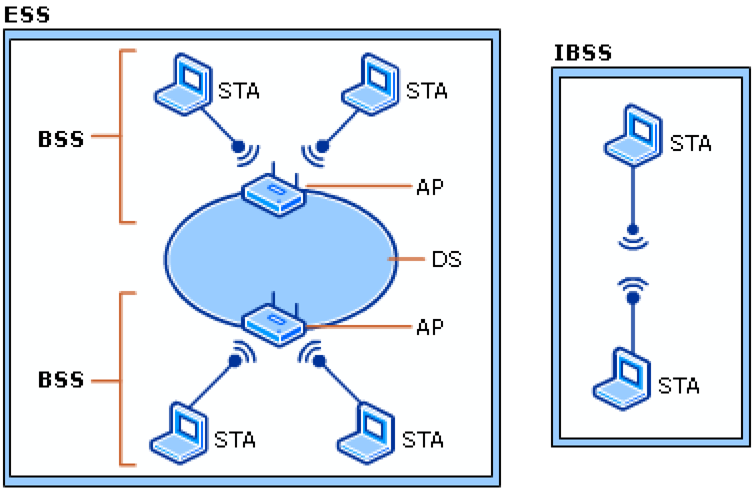
BSS. Every BSS has an identification (ID) called the BSSID, which is the MAC address of the access point servicing the BSS.

An independent BSS (IBSS) is an ad-hoc network that contains no access points, which means they can not connect to any other basic service set.

An infrastructure can communicate with other stations not in the same basic service set by communicating through access points.

2.3. Extended service set (ess)

An extended service set (ESS) is a set of connected BSSes. Access points in an ESS are connected by a distribution system. Each ESS has an ID called the SSID which is a 32-byte (maximum) character string.



2.4. Wireless Distribution system

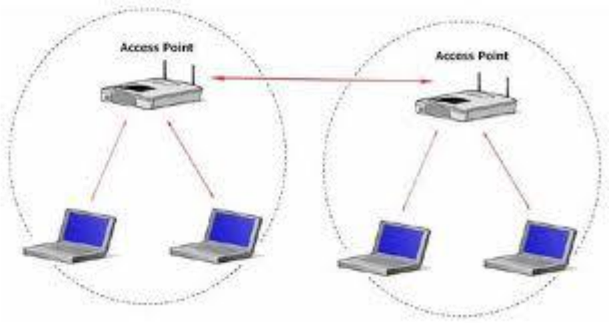
A Wireless Distribution System is a system that enables the wireless interconnection of access points in an IEEE 802.11 network. It allows a wireless network to be expanded using multiple access points without the need for a wired backbone to link them, as is traditionally required. The notable advantage of WDS over other solutions is that it preserves the MAC addresses of client packets across links between access points.

An access point can be either a main, relay or remote base station. A main base station is typically connected to the wired Ethernet. A relay base station relays data between remote base stations, wireless clients or other relay stations to either a main or another relay base station. A remote base station accepts connections from wireless clients and passes them to relay or main stations. Connections between "clients" are made using MAC addresses rather than by specifying IP assignments.

All base stations in a Wireless Distribution System must be configured to use the same radio channel, and share WEP keys or WPA keys if they are used. They can be configured to different service set identifiers. WDS also requires that every base station be configured to forward to others in the system.

WDS may also be referred to as repeater mode because it appears to bridge and accept wireless clients at the same time (unlike traditional bridging). It should be noted, however, that throughput in this method is halved for all clients connected wirelessly.

When it is difficult to connect all of the access points in a network by wires, it is also possible to put up access points as repeaters.



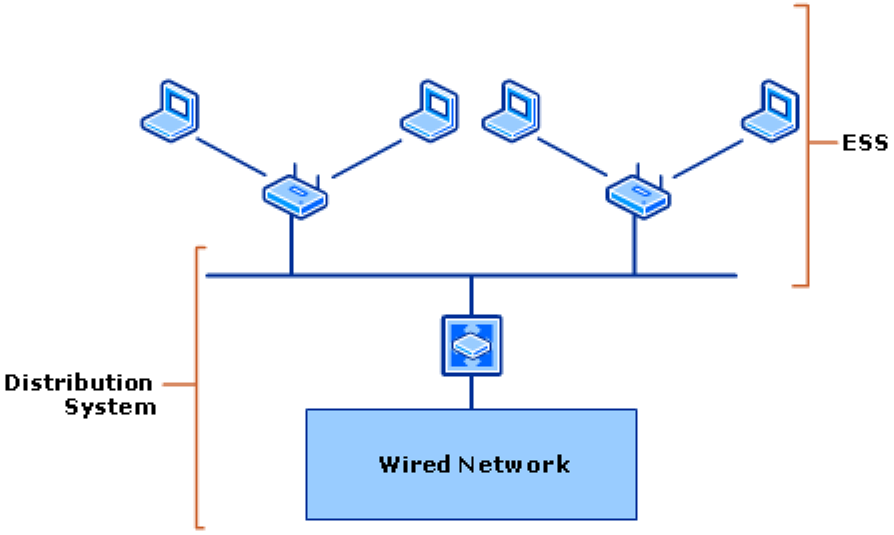
3. 802.11 Operating Modes

IEEE 802.11 defines the following operating modes:

- Infrastructure mode
- Ad hoc mode

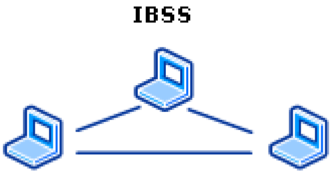
3.1 802.11 Infrastructure Mode

In infrastructure mode, there is at least one wireless AP and one wireless client. The wireless client uses the wireless AP to access the resources of a traditional wired network. The wired network can be an organization intranet or the Internet, depending on the placement of the wireless AP. An extended service set (ESS) is shown in the following figure.



3.2 802.11 Ad Hoc Mode

In ad-hoc mode , wireless clients communicate directly with each other without the use of a wireless AP, as shown in the following figure.



Ad hoc mode is also called peer to peer mode. Wireless clients in ad hoc mode form an independent basic service set (IBSS). One of the wireless clients, the first wireless client in the IBSS, takes over some of the responsibilities of the wireless AP. These responsibilities include the periodic beaconing process and the authentication of new members. This wireless client does not act as a bridge to relay information between wireless clients. Ad hoc mode is used to connect wireless clients together when there is no wireless AP present. The wireless clients must be explicitly configured to use ad hoc mode. There can be a maximum of nine members in an ad hoc 802.11 wireless network.

4. 802.11 Protocols

4.1. 802.11a

The 802.11a standard uses the same data link layer protocol and frame format as the original standard, but an OFDM based air interface (physical layer). It operates in the 5 GHz band with a maximum net data rate of 54 Mbit/s, plus error correction code, which yields realistic net achievable throughput in the mid-20 Mbit/s.

Since the 2.4 GHz band is heavily used to the point of being crowded, using the relatively unused 5 GHz band gives 802.11a a significant advantage. However, this high carrier frequency also brings a disadvantage: the effective overall range of 802.11a is less than that of 802.11b/g. In theory, 802.11a signals are absorbed more readily by walls and other solid objects in their path due to their smaller wavelength and, as a result, cannot penetrate as far as those of 802.11b. In practice, 802.11b typically has a higher range at low speeds (802.11b will reduce speed to 5 Mbit/s or even 1 Mbit/s at low signal strengths). However, at higher speeds, 802.11a often has the same or greater range due to less interference.

4.2. 802.11b

802.11b has a maximum raw data rate of 11 Mbit/s and uses the same media access method defined in the original standard. 802.11b products appeared on the market in early 2000, since 802.11b is a direct extension of the modulation technique defined in the original standard. The dramatic increase in throughput of 802.11b (compared to the original standard) along with simultaneous substantial price reductions led to the rapid acceptance of 802.11b as the definitive wireless LAN technology.

802.11b devices suffer interference from other products operating in the 2.4 GHz band. Devices operating in the 2.4 GHz range include: microwave ovens, Bluetooth devices, baby monitors and cordless telephones.

4.3. 80211g

The newer 802.11g standard improves on 802.11b. It still uses the same crowded 2.4 GHz shared by other common household wireless devices, but 802.11g is capable of transmission speeds up to 54 mbps. Equipment designed for 802.11g will still communicate with 802.11b equipment, however mixing the two standards is not generally recommended.

The 802.11a standard is in a whole different frequency range. By broadcasting in the 5 GHz range 802.11a devices run into a lot less competition and interference from household devices. 802.11a is also capable of transmission speeds up to 54 mbps like the 802.11g standard, however 802.11 hardware is significantly more expensive.

4.4. 802.11h FOR SPECTRUM AND TRANSMIT POWER MANAGEMENT

The 802.11h was originally developed to extend the 802.11 operation in the 5 GHz band in Europe. In order to co-exist with the primary users in the 5 GHz band in Europe — the radar and satellite systems, the 5GHz WLAN devices are required to support DFS (Dynamic Frequency Selection) and TPC (Transmit Power Control). For example, the WLAN devices are required to switch its operational frequency channel to another channel once a radar signal is detected in the current channel. On the other hand, when a satellite signal is detected, the WLAN devices are allowed to use the transmit power up to the regulatory maximum minus 3 dB while normally they can transmit at up to the regulatory maximum level.

The802.11h defines the DFS and TPC mechanisms on top of the 802.11 MAC and the 802.11a PHY for these purposes. Note that, even though the 802.11h has been developed to satisfy the European regulatory requirements, it can be apparently used in other countries for multiple purposes, such as automatic frequency planning, reduction of energy consumption, range control, reduction of interference, and QoS (Quality of Service) enhancement.

5. IEEE802.11 Frame Format

The format of the IEEE802.11 frame sent over the air is shown in Figure 1 .The frame consists of physical information from PHY, MAC headers and the payload.

Preamble	PLCP Header	MAC Frame	CRC

Figure 1

The Preamble field is PHY dependent and includes:

- Synch: An 80-bit sequence of alternating 0 and 1, which is used by the PHY circuitry to select the appropriate antenna (if diversity is used), and to reach steady-state frequency offset correction and synchronization with the received packet timing
- SFD: A start Frame delimiter which consists of a 16-bit binary pattern 00000 1100 1011 1101, which is used to define the frame timing

The PLCP Header field is always transmitted at the basic rate (1 Mbit/sec for 802.11b) and contains Logical Information that will be used by the PHY Layer to decode the frame. It consists of:

- PLCP_PDU Length Word: Represents the number of bytes contained in the packet, this is useful for the PHY to correctly detect the end of the packet.
- PLCP Signalling Field: Contains only the rate information, ecoded in 0,5 Mbps

Header Error Check Field: Is a 16 bit CRC error detection field

5.1 MAC Frame format

MAC management frames are also called MAC Management Protocol Data Units (MMPDUs). Management frames provide such services as authentication, association, and reassociation. Management frame bodies are never relayed through an access point. Instead they are “sourced” (generated) and sunk (read and disposed of) at the MAC layer, and therefore are never passed to the distribution system service or LLC.. Unicast MMPDUs may be acknowledged, retransmitted, and fragmented·

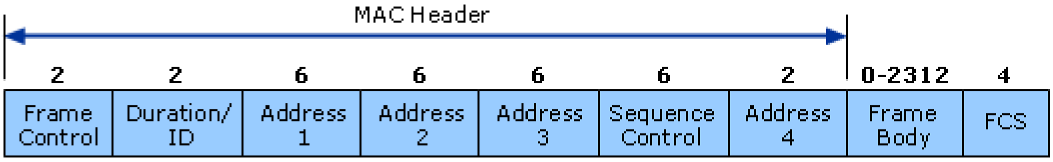


Figure 2

5.1.1 MAC HEADER

5.1.2. FRAME CONTROL FIELD

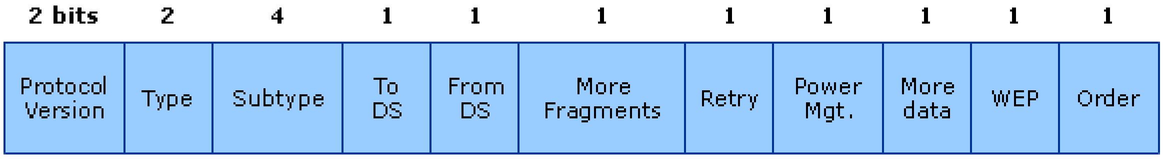


Figure 3

A description of each Frame Control field subfield are as follows:

Protocol Version provides the current version of the 802.11 protocol used. Receiving STAs use this value to determine if the version of the protocol of the received frame is supported.

Type and Subtype determines the function of the frame. There are three different frame type fields: control, data, and management. There are multiple subtype fields for each frame type . Each subtype determines the specific function to perform for its associated frame type.

To DS and From DS indicates whether the frame is going to or exiting from the DS (distributed system), and is only used in data type frames of STAs associated with an AP.

More Fragments indicates whether more fragments of the frame, either data or management type, are to follow.

Retry indicates whether or not the frame, for either data or management frame types, is being retransmitted.

Power Management indicates whether the sending STA is in active mode or power-save mode.

More Data indicates to a STA in power-save mode that the AP has more frames to send. It is also used for APs to indicate that additional broadcast/multicast frames are to follow.

WEP indicates whether or not encryption and authentication are used in the frame. It can be set for all data frames and management frames, which have the subtype set to authentication.

Order indicates that all received data frames must be processed in order.

5.1.3 Duration/ID

The Duration/ID field is the station ID, in power-save Poll messages and in all other frames this is the duration value used for the NAV calculation.

5.1.4 Address

The Address fields, identify the BSS, the destination and source addresses

5.1.5 Sequence Control

The Sequence Control field contains two subfields, the Fragment Number field and the Sequence Number field, as shown in the following figure.

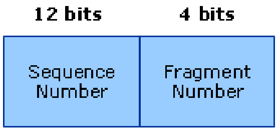


Figure 4

5.1.6 Frame Body

The frame body contains the data or information included in either management type or data type frames.

5.1.7 Frame Check Sequence

The transmitting STA uses a cyclic redundancy check (CRC) over all the fields of the MAC header and the frame body field to generate the FCS value. The receiving STA then uses the same CRC calculation to determine its own value of the FCS field to verify whether or not any errors occurred in the frame during the transmission.

5.2 Frame types

There are three main types of frames:

Data Frames : which are used for data transmission

Control Frames : which are used to control access to the medium (e.g RTS, CTS and ACK)

Management Frames : which are frames that are transmitted the same way as data frames to exchange management information, but are not forwarded to upper layers.

Each of these types is as well subdivided into different Subtypes, according to their specific function.

5.2.1 Management Frames

802.11 management frames enable stations to establish and maintain communications. The following are common 802.11 management frame subtypes

- Authentication frame
- Deauthentication frame
- Association request frame
- Association response frame:
- Reassociation request frame
- Reassociation response frame
- Disassociation frame
- Beacon frame
- Probe request frame
- Probe response frame

5.2.2 Control Frames

802.11 control frames assist in the delivery of data frames between stations. The following are common 802.11 control frame subtypes:

- Request to Send (RTS) frame
- Clear to Send (CTS) frame
- Acknowledgement (ACK) frame

5.2.3 Beacon Frame

An access point (or mobile station in an Ad Hoc network) periodically sends a beacon frame at a rate based on the value of Beacon Interval field**.**

The beacon's frame body resides between the header and the CRC field and constitutes the other half of the beacon frame. Each beacon frame carries some of the following information in the frame body:

- Timestamp
- Beacon Interval
- Capability Info
- SSID
- Supported Rates
- FH parameter set
- DS parameter set
- CF parameter set
- Country code
- Power constraint
- Channel Switch Announcement
- Extended Rate PHY (ERP)
- Extended supported rates
- WME parameters
- WPA/RSN parameters
- Atheros Advanced Capabilities

We explain some of these information:

Beacon interval. This represents the amount of time between beacon transmissions. Before a station enters power save mode, the station needs the beacon interval to know when to wake up to receive the beacon (and learn whether there are buffered frames at the access point).

Timestamp. After receiving a beacon frame, a station uses the timestamp value to update its local clock. This process enables synchronization among all stations that are associated with the same access point.

Service Set Identifier (SSID)**.** The SSID identifies a specific wireless LAN. Before associating with a particular wireless LAN, a station must have the same SSID as the access point. By default, access points include the SSID in the beacon frame to enable sniffing functions (such as that provided by Windows XP) to identify the SSID and automatically configure the wireless network interface card (NIC) with the proper SSID. Some access point vendors have an option to disable the SSID from being broadcast in beacon frames to reduce security issues.

Supported rates**.** Each beacon carries information that describes the rates that the particular wireless LAN supports. For example, a beacon may indicate that only 1, 2, and 5.5Mbps data rates are available. As a result, an 802.11b station would stay within limits and not use 11 Mbps. With this information , stations can use performance metrics to decide which access point to associate with.

Parameter Sets**.** The beacon includes information about the specific signaling methods (such as frequency hopping spread spectrum, direct sequence spread spectrum, etc.). For example, a beacon would include in the appropriate parameter set the channel number that an 802.11b access point is using. Likewise, a beacon belonging to frequency hopping network would indicate hopping pattern and dwell time.

Capability Information. This signifies requirements of stations that wish to belong to the wireless LAN that the beacon represents. For example, this information may indicate that all stations must use wires equivalent privacy (WEP) in order to participate on the network.

Traffic Indication Map (TIM). An access point periodically sends the TIM within a beacon to identify which stations using power saving mode have data frames waiting for them in the access point's buffer. The TIM identifies a station by the association ID that the access point assigned during the association process.

6. IEEE 80211 Layers Description

As any 802.x protocol, the 802.11 protocol covers the MAC and Physical Layer, the Standard currently defines a single MAC which interacts with three PHYs (all of them running at 1 and 2 Mbit/s):

- Frequency Hopping Spread spectrum in the 2.4 GHz Band
- Direct Sequence Spread Spectrum in the 2.4 GHz Band, and
- InfraRed

802.2			Data Link Layer
802.11 MAC			
FH	DS	IR	PHY Layer

Figure

Beyond the standard functionality usually performed by MAC Layers, the 802.11 MAC performs other functions that are typically related to upper layer protocols, such as Fragmentation , Packet Retransmissions and Acknowledges.

7. 802.11 MAC protocols

The Mac Layer defines two different access methods, the Distributed Coordination Function (DCF) and the Point Coordination Function (PCF).

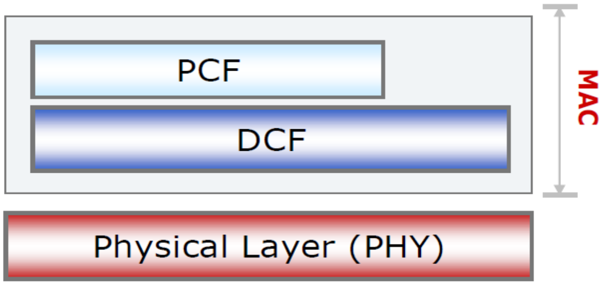


Figure 1

7.1. PCF

PCF has a higher access priority than DCF and is only possible in infrastructure mode. The access-point of a cell acts as a coordinator called the point coordinator (PC) for that cell. The PC grants a contention free channel access to individual nodes by polling them for transmissions. On being polled, a node transmits a single frame destined for any node in the network. In infrastructure mode of 802.11b, the time is divided into periodic superframes which start with the beacon frames. Each superframe is divided into two units, namely, Contention Free Period (CFP) and Contention Period (CP). CFP is the period when contention free channel access is provided by the PC to individual nodes. CP is the period when all nodes contend for the channel using DCF.

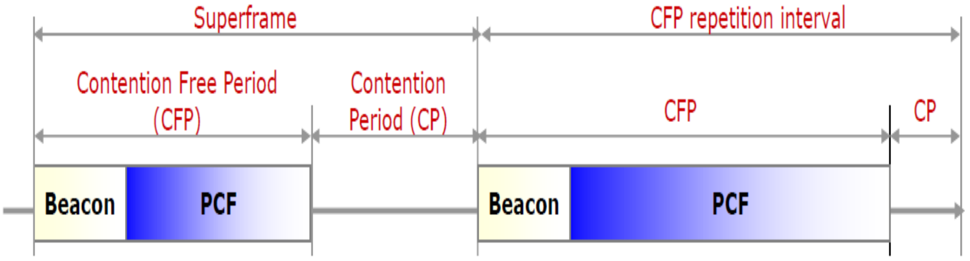


Figure 2

7.2. DCF

The basic access mechanism , called Distributed Coordination Function**,** that is used in contention period**,** is basically a Carrier Sense Multiple Access with Collision Avoidance mechanism (usually known as CSMA/CA).CSMA protocols are well known in the industry, where the most popular is the Ethernet, which is a CSMA/CD protocol(CD standing for Collision Detection).

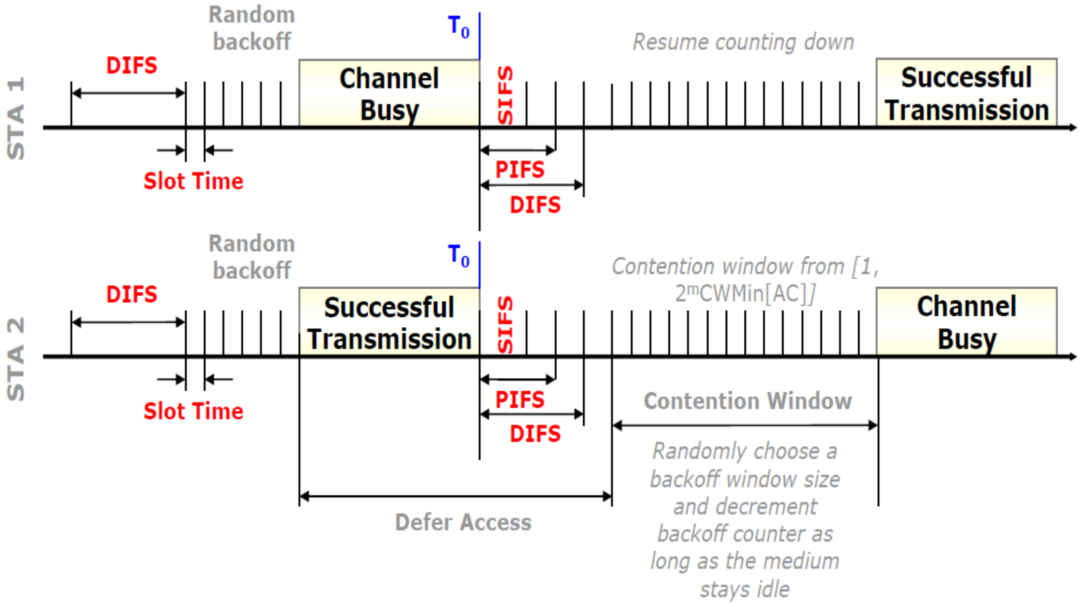


Figure 3

7.3. CSMA-CSMA /CD Algorithms

A CSMA protocol works as follows: A station desiring to transmit senses the medium, if the medium is busy (i.e some other station is transmitting) then the station will defer its transmission to a later time , if the medium is sensed free then the station is allowed to transmit.

These kind of protocols are very effective when the medium is not heavily loaded, since it allows stations to transmit with minimum delay, but there is always a chance of stations transmitting at the same time (collision), caused by the fact that the stations sensed the medium free and decided to transmit at once.

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. In the Ethernet case this collision is recognized by the transmitting stations which go to a retransmission phase based on an exponential random backoff algorithm.

While these Collision Detection mechanisms are a good idea on a wired LAN, they cannot be used on a Wireless LAN environment , because of two main reasons:

1. Implementing a Collision Detection Mechanism would require the implementation of a Full Duplex radio , capable of transmitting and receiving at once, an approach that would increase the price significantly.

2. On a Wireless environment we cannot assume that all stations hear each other(which is the basic assumption of the Collision Detection scheme), and the fact that a station willing

to transmit and senses the medium free ,doesn't necessarily mean that the medium is free around the receiver area.

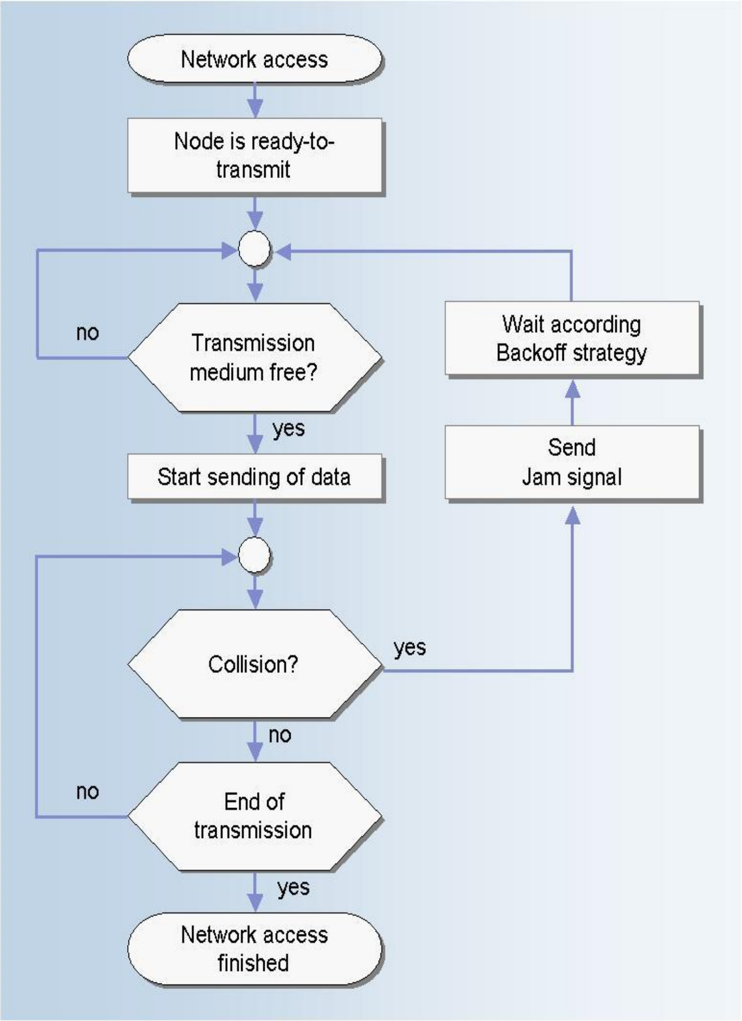


Figure 4

7.4 CSMA /CA Algorithm

In order to overcome these problems, the 802.11 uses a Collision Avoidance mechanism together with a Positive Acknowledge scheme , as follows:

A station willing to transmit senses the medium, if the medium is busy then it defers. If the medium is free for a specified time (called DIFS, Distributed Inter Frame Space, in the standard) then the station is allowed to transmit, the receiving station will check the CRC of the received packet and send an acknowledgment packet (ACK). Receipt of the acknowledgment will indicate the transmitter that no collision occurred. If the sender doesn't receive the acknowledgment then it will retransmit the fragment until it gets acknowledged or thrown away after given number of retransmissions.

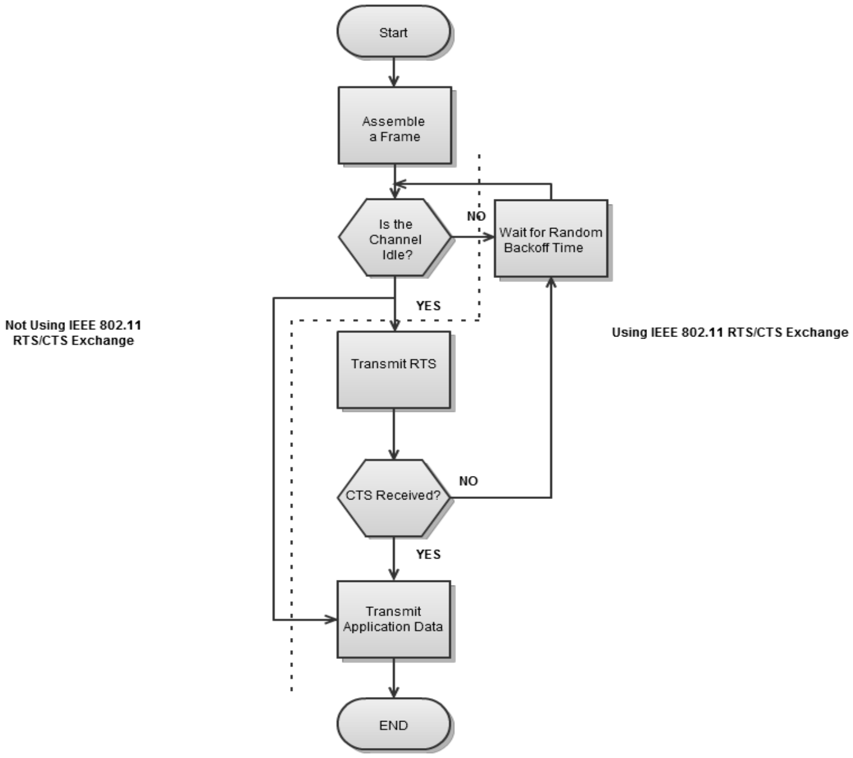

Start
Assemble a Frame
Is the Channel Idle?
NO
Wait for Random Backoff Time
YES
Not Using IEEE 802.11 RTS/CTS Exchange
Using IEEE 802.11 RTS/CTS Exchange
Transmit RTS
CTS Received?
NO
YES
Transmit Application Data
END

Figure 5

7.5 Virtual Carrier Sense

In order to reduce the probability of two stations colliding because they cannot hear each other , the standard defines a Virtual Carrier Sense mechanism:

A station willing to transmit a packet will first transmit a short control packet called RTS (Request To Send), which will include the source, destination and the duration of the following transaction (i.e the packet and the respective ACK), the destination station will respond (if the medium is free) with a response control Packet called CTS (Clear to Send), which will include the same duration information.

All stations receiving either the RTS and/or the CTS, will set their Virtual Carrier Sense indicator (called NAV, for Network Allocation Vector),for the given duration, and will use this information together with the Physical Carrier Sense when sensing the medium.

This mechanism reduces the probability of a collision on the receiver area by a station that is “hidden” from the transmitter ,to the short duration of the RTS transmission, because the station will hear the CTS and “reserve” the medium as busy until the end of the transaction. The duration information on the RTS also protects the transmitter area from collisions during the ACK (by stations that are out of range from the acknowledging station).

It should also be noted that because of the fact that the RTS and CTS are short frames, it also reduces the overhead of collisions, since these are recognized faster than it would be recognized if the whole packet was to be transmitted, (this is true if the packet is significantly bigger than the RTS, so the standard allows for short packets to be transmitted without the RTS/CTS transaction, and this is controlled per station by a parameter called RTS Threshold).

The following diagrams show a transaction between two stations A and B, and the NAV setting of their neighbors:

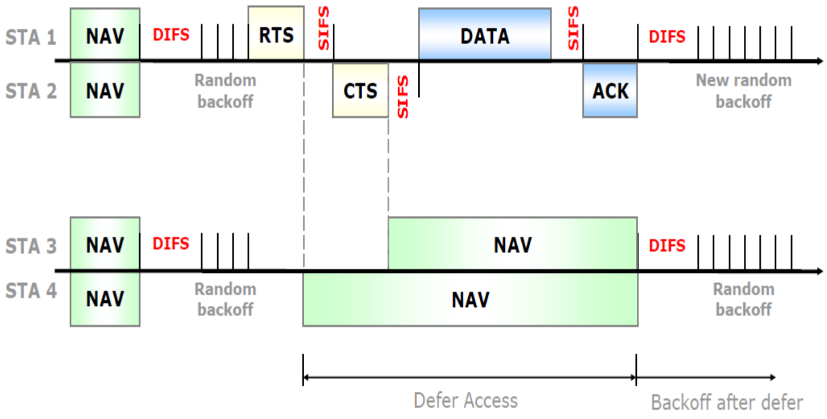


Figure 6

The NAV State is combined with the physical carrier sense to indicate the busy state of the medium.

7.6 MAC Level Acknowledgments

As mentioned earlier ,the MAC layer performs the Collision Detection by expecting the reception of an acknowledge to any transmitted fragment (exception to these are packets that have more than one destination , such as Multicasts, which are not acknowledged.).

7.7 Inter Frame Spaces

The Standard defines 4 types of Inter Frame Spaces, which are used to provide different priorities:

SIFS:- Which stands for Short Inter Frames Space, is used to separate transmissions belonging to a single dialog (e.g Fragment- Ack), and is the minimum Inter Frame Space, and there is always at most one single station to transmit at this given time ,hence having priority over all other stations. This value is a fixed value per PHY and is calculated in such a way that the transmitting station will be able to switch back to receive mode and be capable of decoding the incoming packet, on the 802.11 FH PHY this value is set to 28 microseconds.

PIFS:- Point Coordination IFS, is used by the Access Point (or Point Coordinator, as called in this case), to gain access to the medium before any other station.This value is SIFS plus a Slot Time, i.e 78 microseconds.

DIFS:- Distributed IFS , is the Inter Frame Space used for a station willing to start a new transmission, which is calculated as PIFS plus one Slot Time, i.e 128 microseconds.

EIFS:- Extended IFS ,which is al onger IFS used by a station that has received a packet that could not understand, this is needed to prevent the station (who couls not understand the duration information for the Virtual Carrier Sense) from colliding with a future packet belonging to the current dialog.

8. Various phases of 802.11 MAC layer

8.1. Scanning phase

In scanning phase, mobile node tunes its radio interface on all the available 802.11 b/g channels one by one and tries to extract the signal strength and other useful parameters for those channel. Scanning techniques are broadly classified into two categories namely, active and passive scanning. In passive scanning, a mobile node switches to a particular channel and waits for the beacon frames to arrive on that channel. Beacon frames are normally sent by access points after every 100 ms interval. In active mode of scanning, a mobile node, after switching to a particular channel, sends a probe request on that channel. Probe request is immediately answered by the Access point operating on that channel (if any) in the form of a probe response frame. Scanning phase takes around 350–1000 ms.

8.2. Authentication

Authentication is a process through which the mobile node tries to prove its identity to the Access point. This is done by sending authentication frame to Access point, which can accept or reject the request based on some policy. Time to authenticate is normally less than 10 ms . This time can be further reduced by using preauthentication techniques .

8.3. Association

Association is followed by successful authentication and involves sending reassociation request frames by the mobile node to access point. The access points respond by sending a reassociation reply. After successful association, a mobile node is assigned proper association identity and required resources by the concerned access point. Association process normally takes less than 10 ms of time.

8.4. Background Scanning

Background scanning, which is available in MadWiFi, is the key to faster handoff and channel switch .In background scanning, the client periodically goes in to power save mode for a fixed duration,the duration is also called the dwell time. But instead of going in to sleep mode, the client changes its current working channel and starts active scan to find out potential APs in its neighborhood. The information (e.g. RSSI, rate, etc) about the scanned APs is cached is scan cache. After finishing an active scan on one channel, or on completion of maximum dwell time, the device goes back to its original channel and waits to receive beacon frame. If the beacon frame suggests that AP has frames buffered, then the station comes out of power save mode and gather those frames.

9. DFS (Dynamic Frequency Selection)

DFS is used to switch the operational frequency channel of a BSS to another dynamically , for avoiding interference with other devices, such as radar systems and other WLAN segments, and for uniform utilization of channels. An access point specifies that it uses DFS in the frames WLAN stations use to find access points. When a WLAN station associates or re-associates with an access point, the station reports a list of channels that it can support. When it's necessary to switch to a new channel, the access point uses this data to determine the best channel.

9.1. Types of measurements

The AP should monitor the status of the current and other frequency channels and it may also request other stations to measure and report the channel status. There are three types of measurement:

- Basic type — determines whether another BSS, a non-802.11 OFDM signal, an unidentified signal, or a radar signal is detected in the measured channel

- CCA (Clear Channel Assessment) type — measures the fractional duration of the channel busy period during the total measurement interval
- RPI (Received Power Indication) histogram type —measures the histogram of the quantized measures of the received energy power levels as seen at the antenna connector during the measurement interval.

Based on its own measurement as well as the reports from the associated stations, the AP continues to monitor the channel status so that the channel switching can be conducted at a proper instance.

9.2 Channel Switch

The access point initiates a channel switch by sending a frame to all stations associated with the access point that identifies the new channel number, the length of time until the channel switch takes effect, and whether or not transmission is allowed before the channel switch. Stations that receive the channel switch information from the access point change to the new channel after the elapsed time. Access point measures channel activity to determine if there is other radio traffic in the channels being used for a WLAN. The stations perform measurement of channels activity and generate a report to the access point.

10. MadWiFI Driver Summary

10.1 MadWiFi Device Driver

Our proposed mechanisms were implemented in the Mad- WiFi driver , which is a Linux kernel device driver for wireless LAN chipsets from Atheros Communications. The MadWiFi code consists of four main modules: net80211 stack, the Atheros specific ‘ath’ part, Hardware Abstraction Layer (HAL) and rate algorithms for selecting the best transmission rates. Our modifications have been in the net80211 module.

Madwifi driver has been selected as it is open source and the code is available for modification. Using MadWifi , we can create multiple virtual interfaces on one physical network card. Each virtual interface can work in different modes, namely AP, STA, IBSS(adhoc), Monitor and WDS(bridge).

This driver is capable of performing both active and passive scanning.

Also implements background scanning, but only if there is no traffic for the mobile node for last 250 ms (default value), in which the active scanning on neighboring channels is performed for neighbor discovery..In case traffic resumes on a channel, the background scan is cancelled and priority is accorded to data traffic.

The following section describe the behavior of the AP and STA operations according to the MadWifi driver.

10.2 AP Operation in MadWifi

When the driver is loaded, it probes the physical network card and then sets up the device (ath_attach()**).** The driver also automatically creates a virtual network interface operating in the mode specified (ieee80211_create_vap()). The initial state of the virtual interface is INIT. While in INIT state, the hardware does not receive packets. When the AP interface is opening (for example, by using the command ifconfig ath0 up), the driver sets the hardware properly and enters SCAN state.

10.3 Best-channel selection

In SCAN state(ieee80211_scan_ap.c), the AP scans all supported channels(ap_start()). The scan includes active scan, in which the AP sends out probe request, and passive scan, in which the AP listens to beacons from neighboring APs. In SCAN state, the AP does not transmit packets. After all channels have been scanned(add_channel() , ap_add()), the AP picks a quiet channel(pick_channel()) which has the smallest rssi , and then enters RUN state (ap_end()**).**

In RUN state, the AP performs normal operations of an access point. It broadcasts beacon messages about every 100 ms(ath_beacon_send())**,** replies probe requests sent by other APs, replies authentication and (re)association messages sent by clients, and transmits data packets. When the interfacing is closing, the AP sends disassociation messages to every associated client, then it cleans up the resources and enters INIT state. One thing to note is that the control messages, such as RTS, CTS and ACK, are handled by the HAL part of the driver. The open source part does not process these control messages.

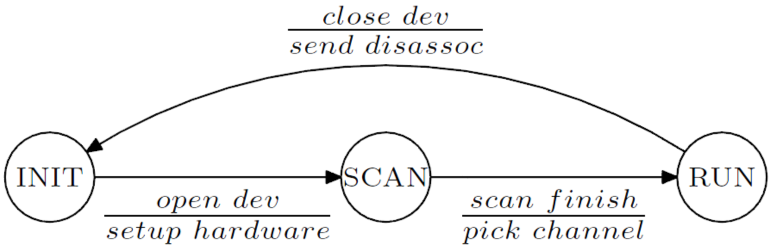


Figure 1

10.4 STA Operation

For STA, the operations in INIT state are the same as those for AP. In SCAN state, the STA also scans all the supported channels. After the scan is done, the STA selects one bss (AP) that has a desired essid and the highest rssi (sta_pick_bss()). If no bss with a matching essid is found, the STA restarts a new scan. If a bss is selected, the STA sets up the parameters required to communicate with the bss, and then enters AUTH state (ieee80211_sta_join1()).

On entering AUTH state, the STA starts an authentication procedure by sending an authentication message to the bss. The authentication procedure includes a sequence of messages exchanged between STA and AP. If the authentication succeeds, the STA enters ASSOC state. On the other hand, if the authentication fails or if the STA does not receive any response from the bss within 5 seconds (ieee80211_tx_timeout()))the STA goes back to SCAN state.

On entering ASSOC state, the STA sends an association request message to bss and waits for a response. If the STA receives a successful association response, it goes to RUN state. If the association fails (eg. error response or rate negotiation failure), or if the STA does not receive any association response within 5 seconds, the STA goes to SCAN state.

In RUN state, the STA can exchange data packets with the bss. The STA also listens to management messages. If the STA receives a dis-association message from the bss, it sends an association request and goes to ASSOC state. If the STA receives a dis-authentication message, it sends an authentication message and goes to AUTH state. In RUN state, the STA maintains the connectivity to the bss by listening to beacon messages. If 10 consecutive beacons are missed (ath_beacon_config()), the connection to the bss is considered broken. The STA sends a re-association request to the bss and enters ASSOC state (ieee80211_beacon_miss()).When the interface is closing, the STA informs the bss by sending a disassociation message to the bss and then enters INIT state.

From the above procedure we can identify a handoff procedure of a STA. The handoff procedure starts when the STA is in RUN state and has 10 consecutive beacons missing. The STA sends a re-association request to the old bss and enters ASSOC state. Without hearing any reply from the bss, the STA enters SCAN state to search for any new bss.

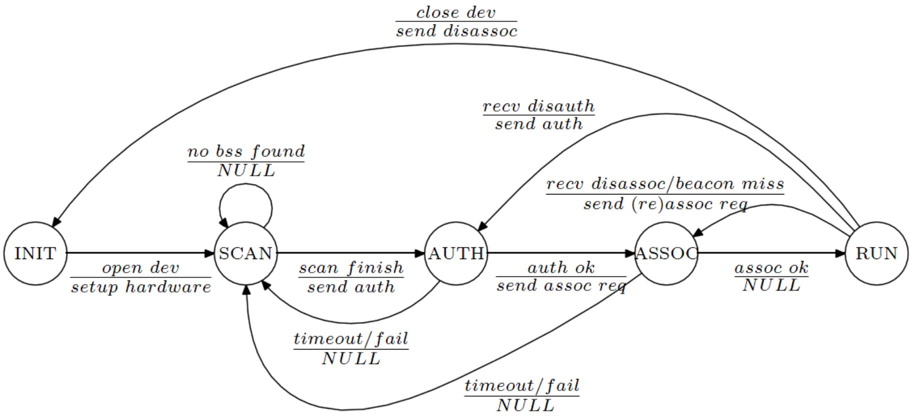


Figure 2

10.5 Background Scanning

MadWiFi provides the functionality of background scanning since version 0.9.3.2. We have modified this to periodically probe only the neighbor APs by calling the ieee80211_bg_scan function. The function calls the scan_restart function to initiate the power save mode. The device then goes off-channel (switches to another channel) for a fixed interval that is large enough to scan most APs. Then the scan_next function is used to scan the channels with active scan. In active scan, the device sends a probe request frames with blank SSID field and waits for probe responses. The received probe response RSSI values, from all the APs in that channel, are extracted and stored in scan_cache. After scanning one channel, the device comes to its original channel and waits to receive a beacon frame. If the beacon frame suggests that the AP has frames queued, then the station comes out of power save mode, and gathers buffered frames from the AP. After collecting these frames, the whole process is repeated again for other channels marked for scanning.

11. DFS IN MADWIFI

The implementation of DFS in MadWifi driver is divided in :

- Radar detection and pattern matching (detection is done in hardware and pattern matching is done in software, in if_ath_radar.c and ieee80211.c)
- ETSI/FCC requirements like CAC which is implemented in if_ath.c for all VAP.
- Best-channel selection in ieee80211_scan_ap.c
- 802.11h Channel Switch Announcement (in beacon/action frames) which is done at the net80211 layer

11.1 Radar Detection

This process takes place in ath/if_ath_radar.c file.

Specifies radar pattern according to the country (since there is a limited and known number of radar patterns, it loops on all possible radar pulse periods)

Checks if radar detection is enabled

Reads and updates the radar pulse detection parameters (according to the local specifications) – I.e. collects pulses that will later on be processed to see if they match any of the known radar patterns

Adds flags to the channels (if DFS is required or if they are for indoor/ outdoor use)

Determines whether DFS is required according to the setup it has previously built

After collecting the above data for a period of time starts searching backwards to discover any radar presence in order to leave channel (if needed).Checks if the new radar pulse is after the last one recorded, or else, it flushes the history.

11.2 802.11h Channel Switch Announcement (CSA - in beacon/action frames)

This functionality is done in ieee80211_beacon.c. In line 143 driver can send channel switch announcement (bo->bo_chanswitch = frm;), size of packet is calculated and as far as the appropriate buffer size is allocated beacon is sent. If a node, for any reason looses the CSA beacon it starts scanning very fast in order to re-find its AP at their new channel.

12. Overlapping Channels in Wireless Networks

The IEEE 802.11 b/g standards operate in the unlicensed ISM 2.4 Ghz spectrum which has 11 out of 14 channels available for use. The channel number (1 . . . 11) represents the center frequency on which the radios operate (e.g. 2.412 Ghz for channel 1). The center frequencies are separated by 5 MHz, while the channels have a spread of about 30 MHz around the center frequency.

As a result, a signal on any 802.11b channel overlaps with several adjacent channels, with the extent of overlap decreasing with increasing separation between the center frequencies. Attributing to this overlap, a transmission on one channel becomes interference to stations on an overlapping channel, also known as adjacent channel interference. For example channel 1 takes up bandwidth from 2412Mhz to 2435Mhz but channels 2-5 overlap with that (see diagram below). With an overlap, the bandwidth has to be shared with any other access points operating in those overlapping channels.

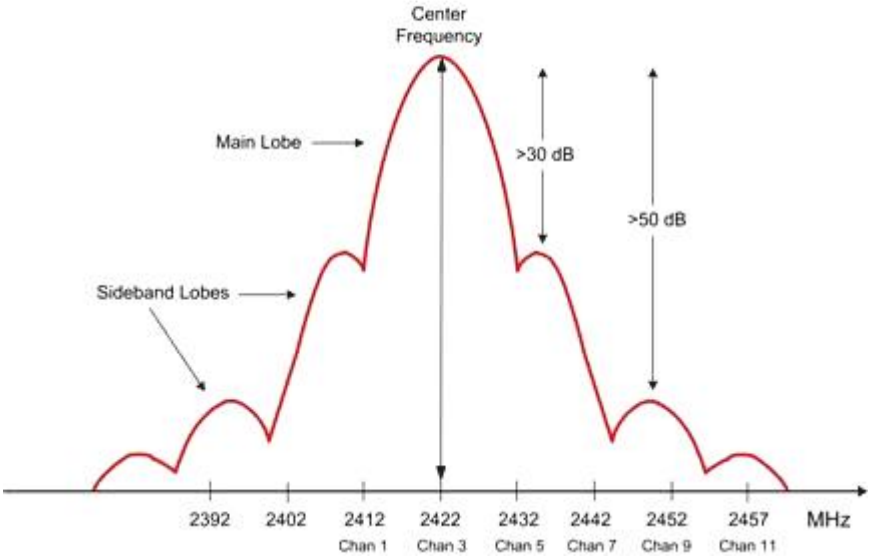


Figure 1

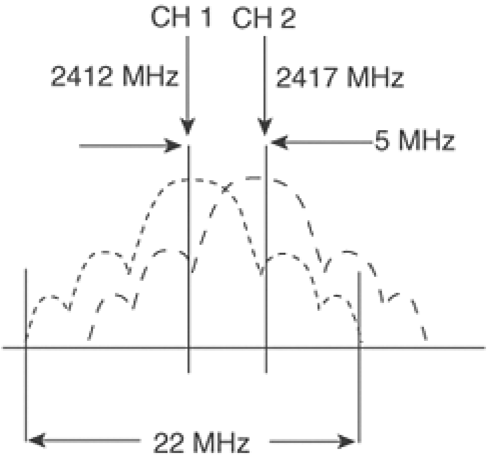


Figure 2

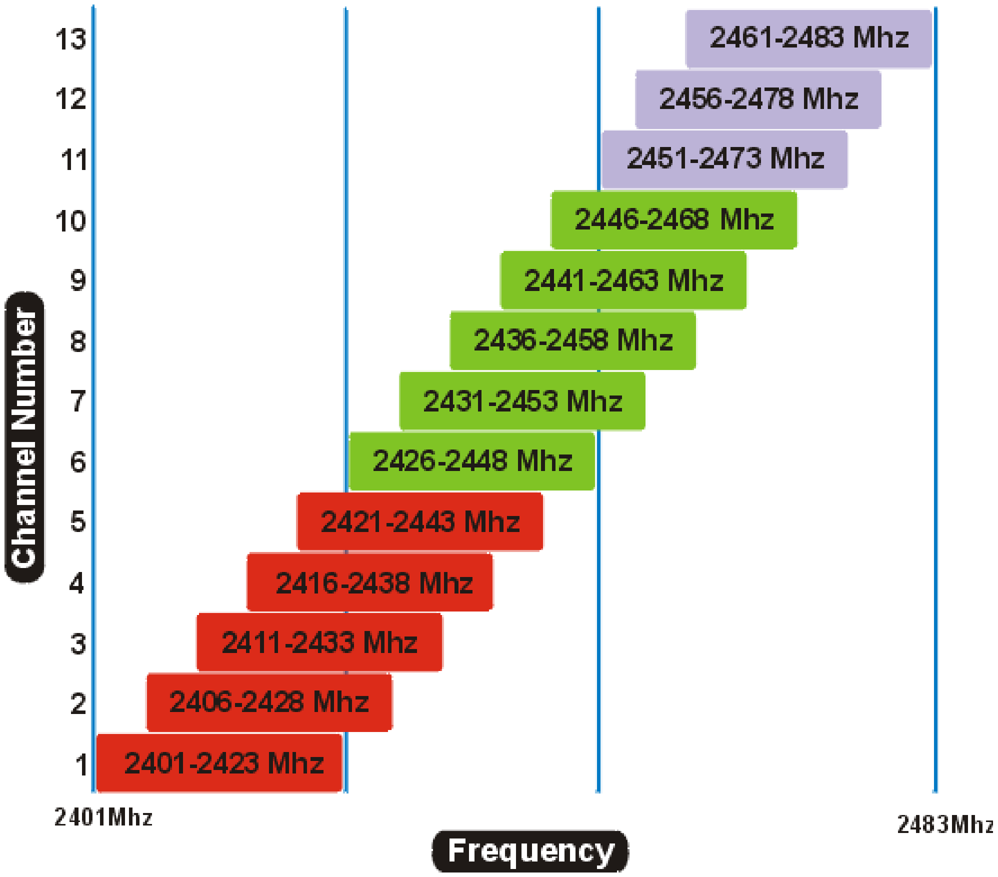

13
12
11
10
9
8
7
6
5
4
3
2
1
Channel Number
2461-2483 Mhz
2456-2478 Mhz
2451-2473 Mhz
2446-2468 Mhz
2441-2463 Mhz
2436-2458 Mhz
2431-2453 Mhz
2426-2448 Mhz
2421-2443 Mhz
2416-2438 Mhz
2411-2433 Mhz
2406-2428 Mhz
2401-2423 Mhz
2401Mhz
2483Mhz
Frequency

Figure 3

12.1 Maximizing Throughput

The easiest and best means of increasing throughput for a weak Wi-Fi connection is to change the channel on which you transmit and receive data. IEEE 802.11 defines several different channels on which a wireless device can broadcast

You need to choose a channel that does not have a lot of traffic or interference so that your data acquisition device has the maximum bandwidth available for your data. Channel six is the default channel for most routers on the market and, therefore, the most crowded. The easiest way of determining which channels are open is by using an RSSI (received signal strength indication) scanner or monitor. Many wireless network interface card (NIC) configuration utilities have a built-in function for this purpose

Using the three nonoverlapping channel (1 , 6 , 11), you can reuse the channels in a rotating scheme and carefully define adjacent cells on channels that are noninterfering.

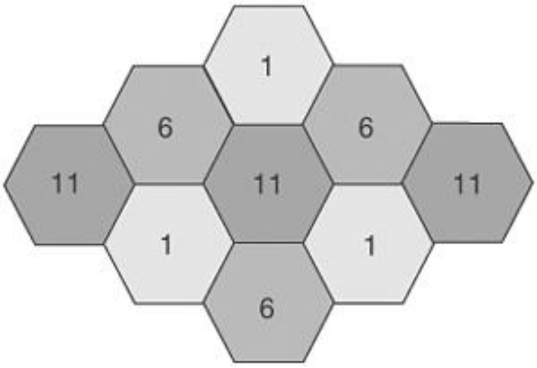


Figure 4

12.2 Experiment

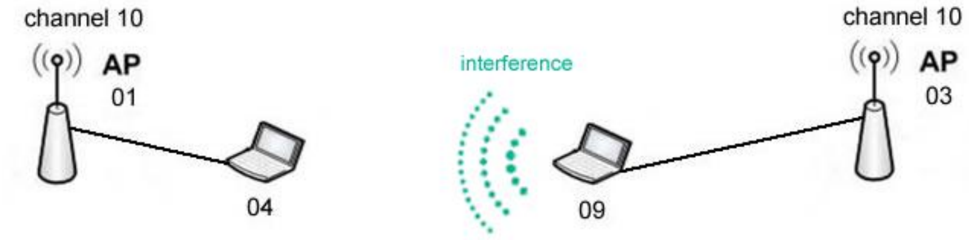


Figure 5

In this experiment, we set up a network on channel 10,that consists of two bss.Bss1 has one AP , node 3 and one STA, node 9 .Bss2 has one AP , node 1 and one STA, node 4. The stations 4 , 9 generate UDP traffic, using two Iperf clients that run simultaneously, while the corresponding Iperf servers runs at the APs. When the STAs associate with the APs, Iperf server start receiving data and measuring the actual throughput. Figure illustrates how the average throughput for bss2, achieved per interference rate. This throughput is attained for channel separation (between bss1-bss2) increased from 0 (same channel) to 5 (non-overlapping channels). As clearly shown in the Figure, when the interference rate is increasing the link quality is degraded but notice that a step by step decrease in overlap ,increases or stabilizes the AP-STA throughput.

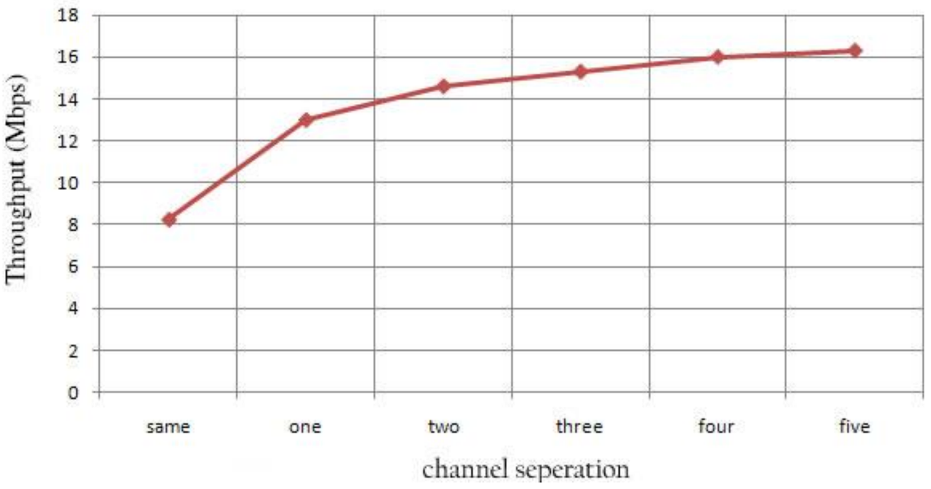
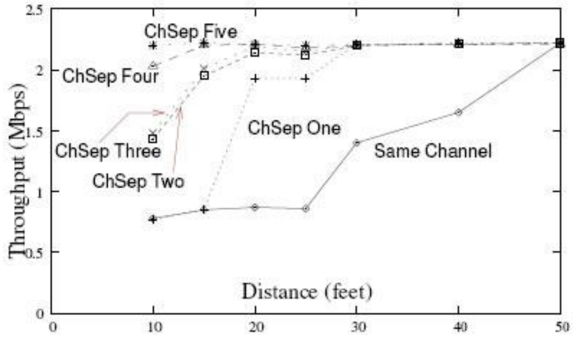


Figure 6

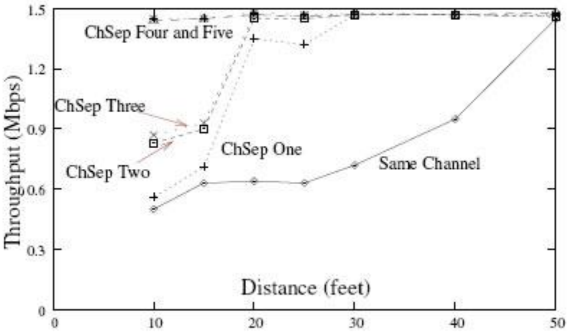
12.3 Throughputs versus spatial separation (*Channel Separation*)

In the figure 7(a) we plot the UDP throughput achieved by the AP-STA Pair-B against distance from Pair-A. This experiment used a data rate of 2 Mbps which best shows how a step by step decrease in overlap increases the AP-STA throughput (other data rates omitted due to space restrictions). Figure shows that using the same channel, a distance of around 50 feet would be necessary to eliminate the interference and attain the maximum possible throughput. This throughput is attained for much smaller distances as the channel separation (indicated by the legend *ChSep*) is increased from 0 (same channel) to 5 (non-overlapping channels). Figure 7(b) shows the same effect on TCP throughput.

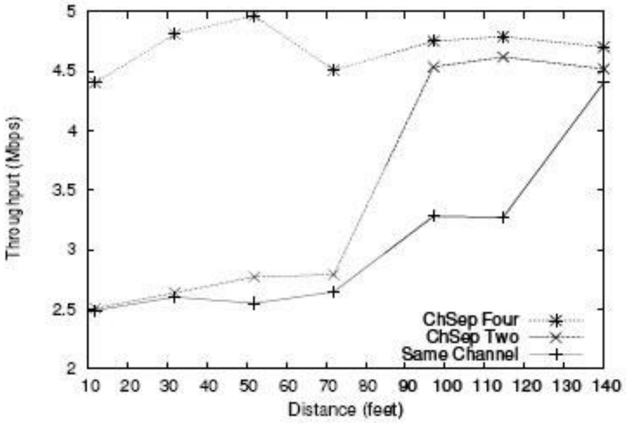
Figure 7(c) shows the TCP throughput in a second environment, which is an office building with a long corridor on which the experiments were performed. The data rate here was set at 11 Mbps.



7(a) UDP throughput



7 (b) TCP throughput



7(c) TCP throughput at 11Mbps in asecond office-building environment

Figure 8 shows the number of collisions that occurred at the AP-STA Pair-B. The number of collisions have a significant amount of variation from one execution of the experiment to another because of the randomness present in the MAC protocol. However, clearly noticeable is the fact that a drastic reduction in the number of collisions (as shown by the *cutoff* at around 100 in Figure 8) indicates significant reduction in interference. Thus, the points at which each curve takes a plunge below the *cutoff* line (shown in the figure) indicates the distance at which interference does not occur between the two APSTA pairs.

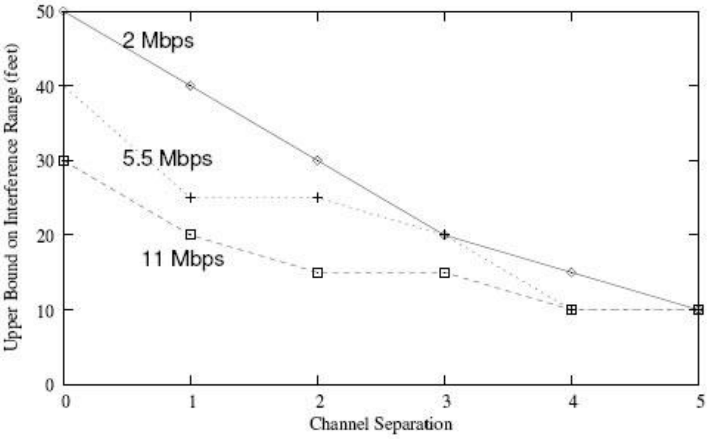


Figure 8: Number of collisions versus spatial separation.

Figure 9 shows the effect of the 802.11 MAC data rate on the interference range of a BSS. Each point on a curve plots the minimum observed distance (modulo discrete observation points) on the y-axis at which with the given channel separation (x-axis) the two AP-STA pairs do not interfere with each other and attain the maximum possible TCP/UDP throughput – which is the interference range of the BSS formed by an AP-STA pair. One observes that for all three data rates, the interference range decreases consistently. Also as expected, for a given amount of channel overlap, a higher data rate has a smaller interference range.

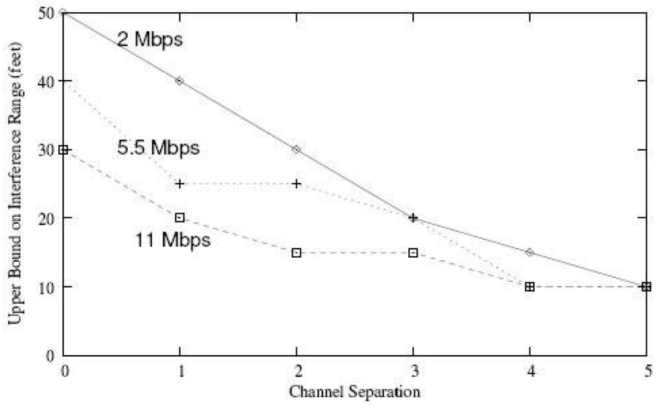


Figure 9: Interference ranges vs channel separation for data rates of 2, 5.5 and 11 Mbps.

13. Received signal strength indication

In a IEEE 802.11 system RSSI is the received signal strength in a wireless environment, in arbitrary units. RSSI can be used internally in a wireless networking card to determine when the amount of radio energy in the channel is below a certain threshold at which point the network card is clear to send (CTS). Once the card is clear to send, a packet of information can be sent. The end-user will likely observe an RSSI value when measuring the signal strength of a wireless network through the use of a wireless network monitoring tool like Network Stumbler.

13.1 How is RSSI used in the Madwifi driver?

In MadWiFi, the reported RSSI for each packet is actually equivalent to the Signal-to-Noise Ratio (SNR) and hence we can use the terms interchangeably. This does not necessarily hold for other drivers though. This is because the RSSI reported by the MadWiFi HAL is a value in dBm that specifies the difference between the signal level and noise level for each packet. Hence the driver calculates a packet's absolute signal level by adding the RSSI to the absolute noise level.

As of MadWiFi version 0.9.3, the noise floor is no longer assumed to be a constant -95 dBm. The noise floor is now updated on each receive interrupt by calling out to the HAL, which returns the absolute noise level in dBm. However, because each interrupt can service several packets we cannot get a noise reading for each and every packet. MadWiFi simply takes the measured noise level at each interrupt and assumes all packets serviced during the interrupt were received with this noise level. The measured noise is not used during scanning at present, so all scan results return with a noise level of -95 dBm.

The "Quality" parameter reported by some of the Wireless Tools such as iwconfig is used by MadWiFi to report the SNR (RSSI). It should not be regarded as a percentage (ignore the /94 part). It simply specifies the average RSSI of the last few received frames. You can see the relationship in the following equation: SNR = Signal - Noise.

In general, an RSSI of 10 or less represents a weak signal although the chips can often decode low bit-rate signals down to -94dBm. An RSSI of 20 or so is decent. An RSSI of 40 or more is very strong and will easily support both 54MBit/s and 108MBit/s operation. Don't be surprised if you get very different figures across invocations of iwconfig (or whichever iw tool), the RSSI will change with time due to interference, channel fading etc.

13.2 Using a Percentage Signal Strength Metric

To circumvent the complexities (and potential inaccuracies) of using RSSI as a basis for reporting dBm signal strength, it is common to see signal strength represented as a percentage. The percentage represents the RSSI for a particular packet divided by the RSSI_Max value (multiplied by 100 to derive a percentage). Hence, a 50% signal strength with a Symbol card would convert to an RSSI of 16 (because their RSSI_Max = 31). Atheros, with RSSI_Max=60, would have RSSI=30 at 50% signal strength. Cisco ends up making life easy with an RSSI_Max =100, so 50% is RSSI=50. It can be seen that use of a percentage for signal strength provides a reasonable metric for use in network analysis and site survey work. If signal strength is 100%, that’s great! When signal strength falls to roughly 20%, you’re going to reach the Roaming Threshold. Ultimately, when signal strength is down somewhere below 10% (and probably closer to 1%), the channel is going to be assumed to be clear. This conceptualization obviates the need to consider dBm, the RSSI_Max, or the “knee” in the logarithmic curve of mW to dBm conversion. It allows a reasonable comparison between environments even though different vendor’s NICs were used to make the measurements. Ultimately, the generalized nature of a percentage measurement allows the integer nature of the RSSI to be overlooked.

13.3 The Impossibility of Measuring 0% Signal Strength

In the preceding paragraph, you may have noticed the note in parentheses stating that the clear channel threshold was “probably closer to 1%.” There is a very profound reason why this was not “0%.” If signal strength falls to 0%, it can be assumed that RSSI=0 and, hence, the signal strength is at, or below, the Receive Sensitivity of the NIC. A NIC can’t report that a particular packet has “0% signal strength,” because if there were no available signal, there would be no packet to measure! It is impossible for any tool using a standard wireless NIC to measure signal strength below the NIC’s Receive Sensitivity threshold.

13.4 Conversion for Atheros

Unlike the other vendors described, Atheros uses a formula to derive dBm.

RSSI_Max = 60

Convert % to RSSI

Subtract 95 from RSSI to derive dBm

Notice that this gives a dBm range of –35dBm at 100% and –95dBm at 0%.

14. OUR DFS IMPLEMENTATION

The non-existence of a dfs implementation for avoiding interference with other WLAN devices and for uniform utilization of channels provided motivation for the present work.

We propose a DFS algorithm ,which periodically discovers the best operating channel according to RSSI measures of background scanning .The associated and the contending stations send measurements to the AP. Based on its own measurement as well as the reports from the associated and contending stations, the AP makes the channel switching decision .Through this process the AP performs better area discovery and upgrades the link quality .

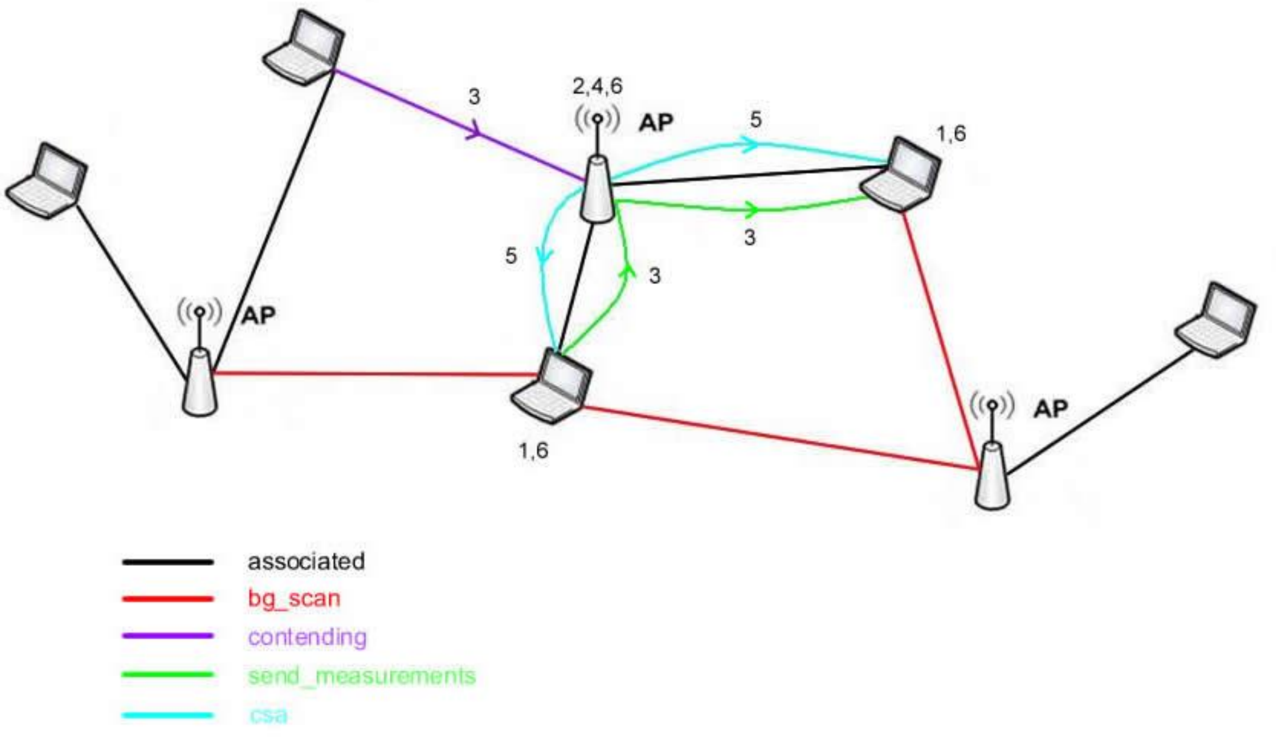
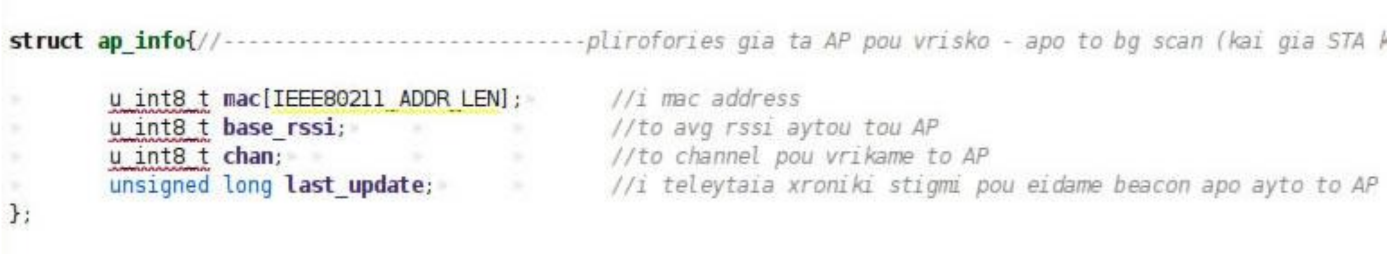


Figure 1 shows an example of the proposed dfs operation being performed on a network that consists of 3 BSS

A detailed description of the proposed DFS protocol steps is given below.

14.1. Station Background Scanning



```
struct ap_info{//-----------------------------plirofories gia ta AP pou vrisko - apo to bg scan (kai gia STA

        u_int8_t mac[IEEE80211_ADDR_LEN];       //i mac address
        u_int8_t base_rssi;                     //to avg rssi aytou tou AP
        u_int8_t chan;                          //to channel pou vrikame to AP
        unsigned long last_update;              //i teleytaia xroniki stigmi pou eidame beacon apo ayto to AP
};
```

Figure 2 shows the format of background list

In this step stations use periodic (40 seconds) background scanning(Figure 3) for discovery and for ascertaining signal strength of the access points operating in any channel, without losing connectivity with the current access point. In background scanning, the client periodically goes in to power save mode, but instead of going in to sleep mode, the station switches to a particular channel and waits for the beacon frames to arrive on that channel or starts active scan to find out potential Aps in the channel. Beacon frames are normally sent by access points after every 100 ms interval .

The information like the mac address of the AP , the channel number , a kind of average value of the receiving signal strength(RSSI) and a timestamp (represents the last time we heard a beacon or probe response) about the scanned APs is cached in background list(Figure 2). If the information includes a mac address that already exists in the background scanning list ,then we update the registration.. After finishing a scan on one channel , or on completion of maximum dwell time, the device goes back to its original channel and waits to receive beacon frame. If the beacon frame suggests that AP has frames buffered, then the station comes out of power save mode and gather those frames.

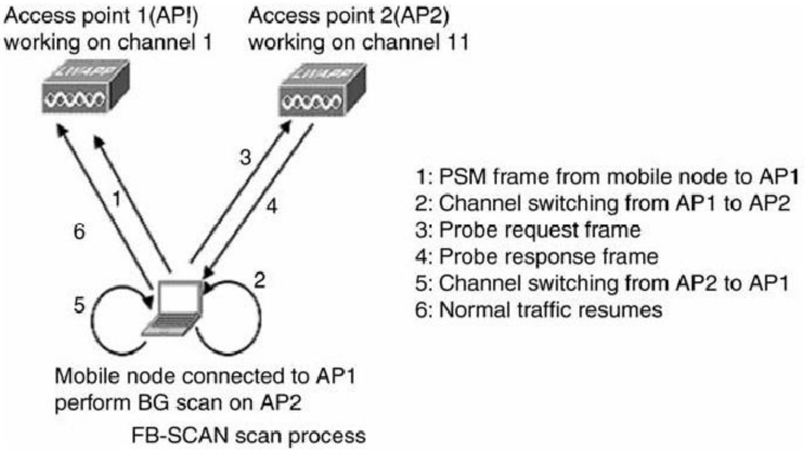


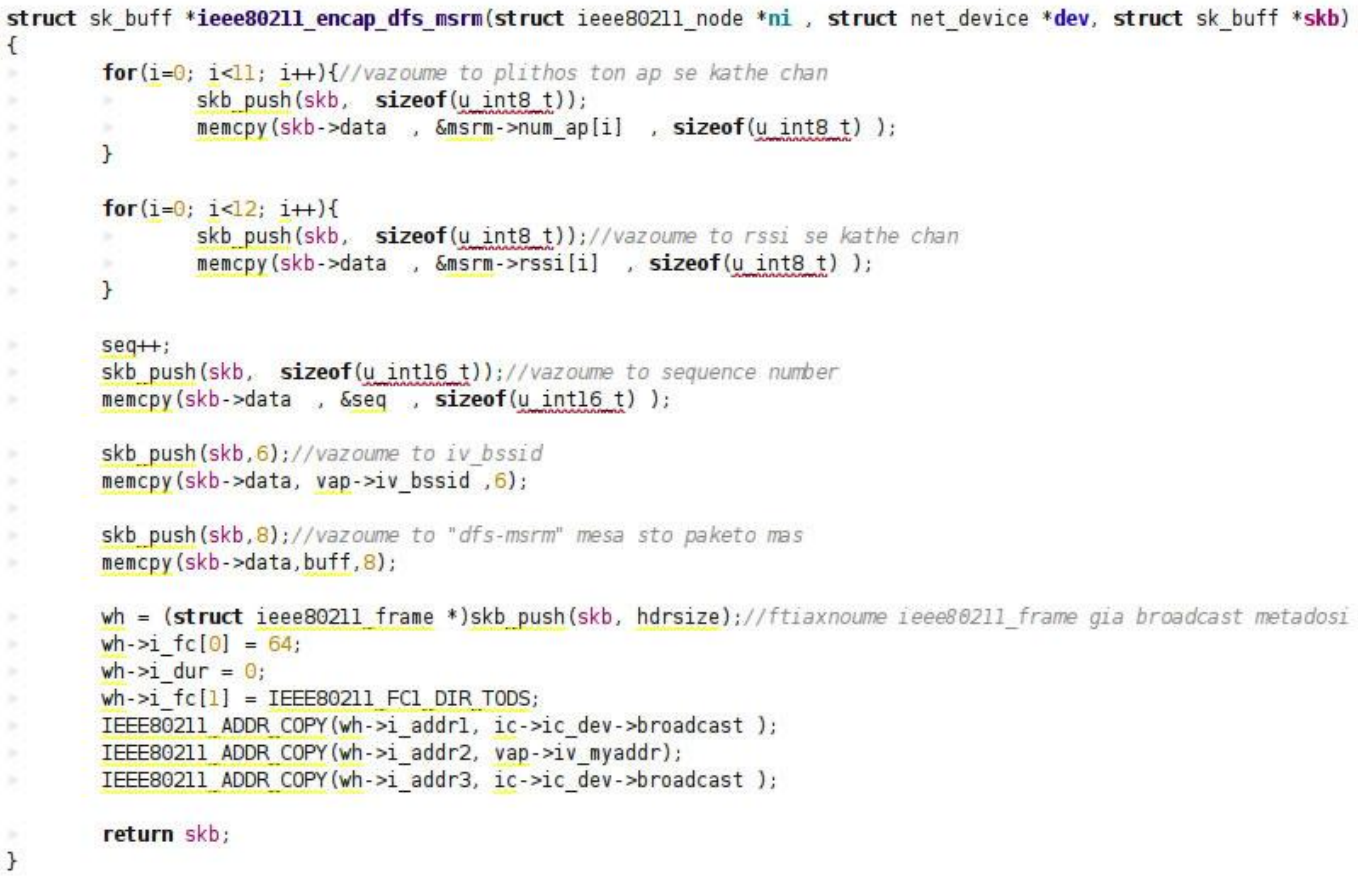
Figure 3 Representation of background scan operation on a single neighbor channel

14.2. AP Background Scanning

In AP mode, a periodic background scan (every 50 seconds) starts . Instead of scanning all the 11 channels at one go, channels are divided into smaller groups (consisting of 4 different channels) and a given group is scanned at one go. The next 50 seconds , AP scans the next 4 channels of the list .This is happening because AP must wait on the operating channel, for stations requests. This is why ,we scan only 4 channels each time and not all the channel list like stations. We use the same background scanning list with the same information, as stations.

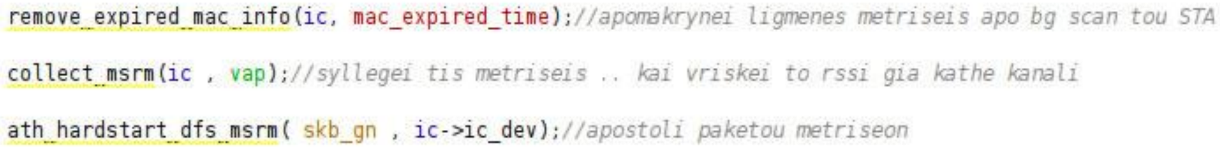
14.3. Station sends Measurements to associated AP

The station after finishing a background scan sends a frame(ath_hardstart_dfs_msrm()) ,which contains the background scanning measurements, to the associated AP. The format of the measurement frame sent over the air is shown in Figure 6 .The frame contains the string “ dfs-msrm ” at the beginning , then the RSSI of each channel, the number of AP found on each channel, the sequence number of each packet and its bssid . For implementing measurement packet transmission support in Mad-WiFi , we have added a function that broadcasts this type of packets.



```
struct sk_buff *ieee80211_encap_dfs_msrm(struct ieee80211_node *ni , struct net_device *dev, struct sk_buff *skb)
{
        for(i=0; i<11; i++){//vazoume to plithos ton ap se kathe chan
                skb_push(skb,  sizeof(u_int8_t));
                memcpy(skb->data  , &msrm->num_ap[i]  , sizeof(u_int8_t) );
        }

        for(i=0; i<12; i++){
                skb_push(skb,  sizeof(u_int8_t));//vazoume to rssi se kathe chan
                memcpy(skb->data  , &msrm->rssi[i]  , sizeof(u_int8_t) );
        }

        seq++;
        skb_push(skb,  sizeof(u_int16_t));//vazoume to sequence number
        memcpy(skb->data  , &seq  , sizeof(u_int16_t) );

        skb_push(skb,6);//vazoume to iv_bssid
        memcpy(skb->data, vap->iv_bssid ,6);

        skb_push(skb,8);//vazoume to "dfs-msrm" mesa sto paketo mas
        memcpy(skb->data,buff,8);

        wh = (struct ieee80211_frame *)skb_push(skb, hdrsize);//ftiaxnoume ieee80211_frame gia broadcast metadosi
        wh->i_fc[0] = 64;
        wh->i_dur = 0;
        wh->i_fc[1] = IEEE80211_FC1_DIR_TODS;
        IEEE80211_ADDR_COPY(wh->i_addr1, ic->ic_dev->broadcast );
        IEEE80211_ADDR_COPY(wh->i_addr2, vap->iv_myaddr);
        IEEE80211_ADDR_COPY(wh->i_addr3, ic->ic_dev->broadcast );

        return skb;
}
```

Figure 4 shows a pseudo code description of measurement frame encapsulation



```
remove_expired_mac_info(ic, mac_expired_time);//apomakrynei ligmenes metriseis apo bg scan tou STA

collect_msrm(ic , vap);//syllegei tis metriseis .. kai vriskei to rssi gia kathe kanali

ath_hardstart_dfs_msrm( skb_gn , ic->ic_dev);//apostoli paketou metriseon
```

Figure 5 shows a pseudo code description of station actions in this step

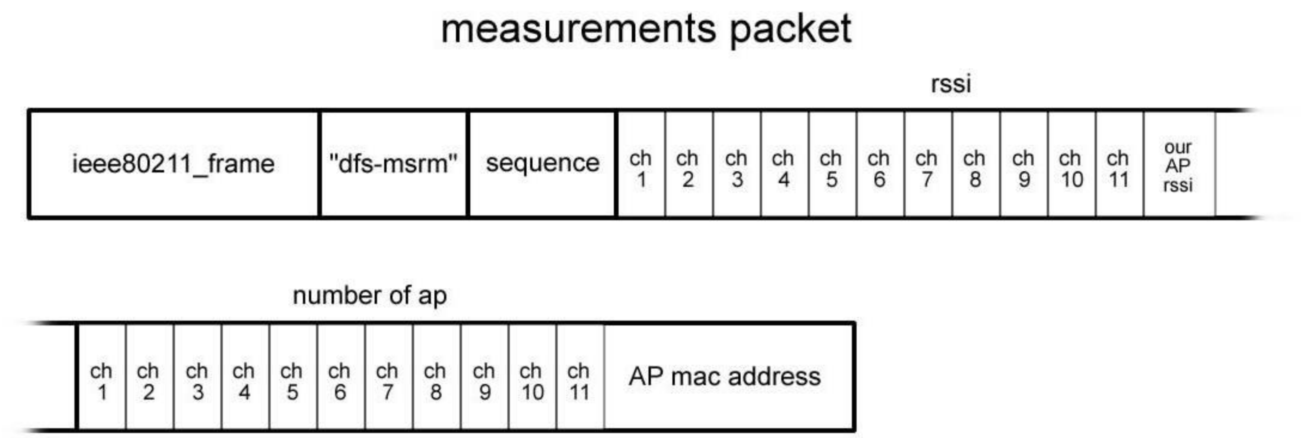
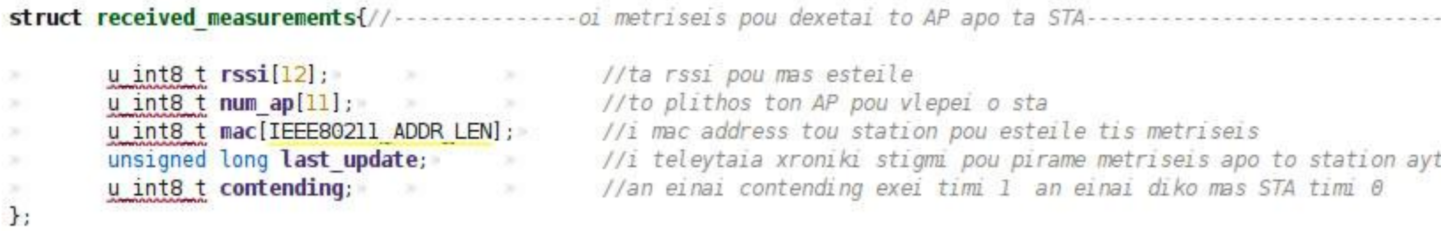


Figure 6 shows the measurement frame format

The string “ dfs-msrm ” is used at the AP side ,to recognize a measurement packet. Moreover, station scans its background scanning list and imports the highest RSSI of each channel and the number of AP found on each channel, in the measurement packet(collect_msrm()).The RSSI of the associated AP is not interference so it is not included in the array of the highest RSSI. Additionally , station removes the AP expired information from the background scanning list(remove_expired_mac_info()).The bssid is used , so the AP can recognize if the receiving measurement packet is coming from an associated or a contending station.



```
struct received_measurements{//---------------oi metriseis pou dexetai to AP apo ta STA----------------------------

        u_int8_t rssi[12];                      //ta rssi pou mas esteile
        u_int8_t num_ap[11];                    //to plithos ton AP pou vlepei o sta
        u_int8_t mac[IEEE80211_ADDR_LEN];       //i mac address tou station pou esteile tis metriseis
        unsigned long last_update;              //i teleytaia xroniki stigmi pou pirame metriseis apo to station ayt
        u_int8_t contending;                    //an einai contending exei timi 1  an einai diko mas STA timi 0
};
```

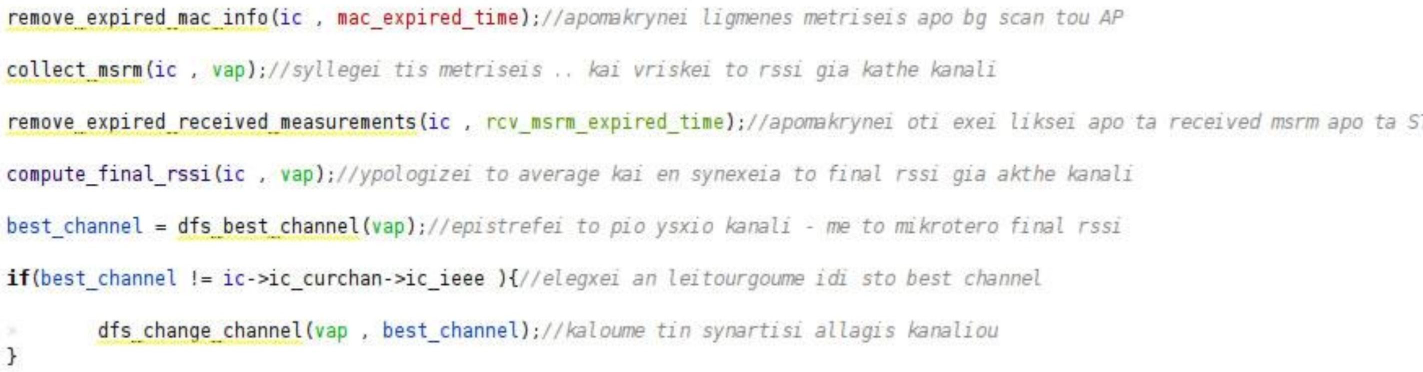
Figure 7 shows the format of measurement list

14.4. AP collects measurements , computes final RSSI for each channel and selects the best channel

In this step, AP must receive the measurement packets from the stations(Figure 9) and collect the background scanning information/measurements ,like the way stations do, to find the biggest RSSI of each channel and the number of AP found on each channel.(collect_msrm ()).

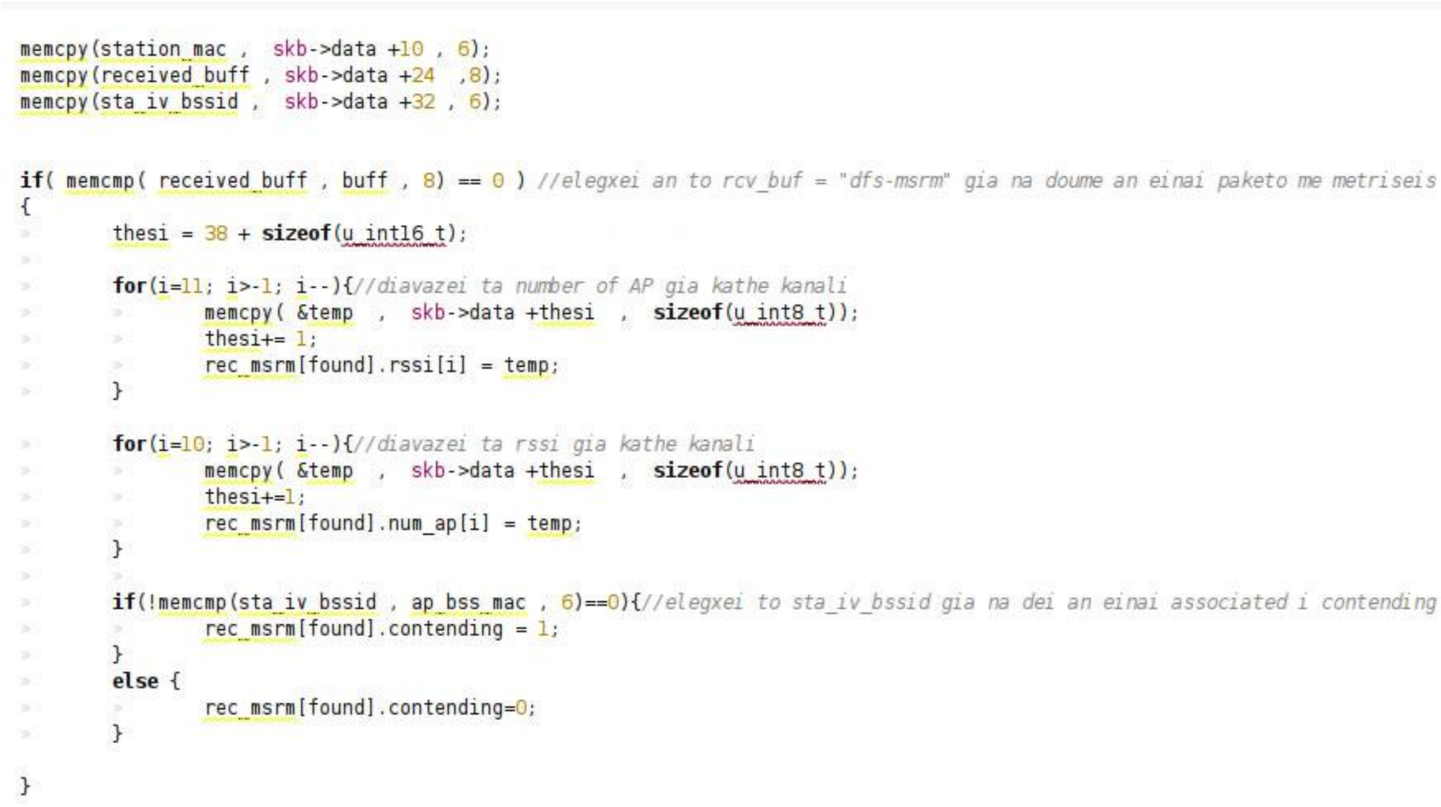
Every time there is an interrupt of receiving measurement packet (containing string “dfs-msrm”,(Figure 9) , AP caches this information in a measurement list for each station . In this way, AP has (N + 1) RSSI values for each channel ,from its background measurement list and from the associated stations or the contending stations ,if they are operating on the same channel.

Periodically(50 seconds) , AP checks the measurement lists(remove_expired_received_measurements()) and the background measurement list (remove_expired_mac_info())and removes the old receiving measurements (we use timestamps).After that AP is in position to compute the average RSSI for each channel . The RSSI measurement ,of the AP operating channel , which is received from a contending station is not used by the AP in calculating the average value. The reason AP does that, is the fact that contending stations hear and measure it like interference.



```
remove_expired_mac_info(ic , mac_expired_time);//apomakrynei ligmenes metriseis apo bg scan tou AP

collect_msrm(ic , vap);//syllegei tis metriseis .. kai vriskei to rssi gia kathe kanali

remove_expired_received_measurements(ic , rcv_msrm_expired_time);//apomakrynei oti exei liksei apo ta received msrm apo ta S

compute_final_rssi(ic , vap);//ypologizei to average kai en synexeia to final rssi gia akthe kanali

best_channel = dfs_best_channel(vap);//epistrefei to pio ysxio kanali - me to mikrotero final rssi

if(best_channel != ic->ic_curchan->ic_ieee ){//elegxei an leitourgoume idi sto best channel

        dfs_change_channel(vap , best_channel);//kaloume tin synartisi allagis kanaliou
}
```

Figure 8 shows a pseudo code description of AP actions in this step

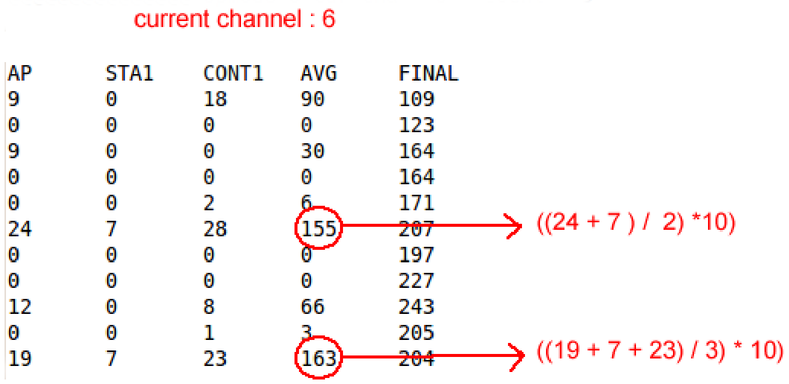


```
memcpy(station_mac ,  skb->data +10 , 6);
memcpy(received_buff , skb->data +24  ,8);
memcpy(sta_iv_bssid ,  skb->data +32 , 6);


if( memcmp( received_buff , buff , 8) == 0 ) //elegxei an to rcv_buf = "dfs-msrm" gia na doume an einai paketo me metriseis
{
        thesi = 38 + sizeof(u_int16_t);

        for(i=11; i>-1; i--){//diavazei ta number of AP gia kathe kanali
                memcpy( &temp  ,  skb->data +thesi  ,  sizeof(u_int8_t));
                thesi+= 1;
                rec_msrm[found].rssi[i] = temp;
        }

        for(i=10; i>-1; i--){//diavazei ta rssi gia kathe kanali
                memcpy( &temp  ,  skb->data +thesi  ,  sizeof(u_int8_t));
                thesi+=1;
                rec_msrm[found].num_ap[i] = temp;
        }

        if(!memcmp(sta_iv_bssid , ap_bss_mac , 6)==0){//elegxei to sta_iv_bssid gia na dei an einai associated i contending
                rec_msrm[found].contending = 1;
        }
        else {
                rec_msrm[found].contending=0;
        }

}
```

Figure 9 shows a pseudo code description of AP actions when is receiving a measurement packet

In 802.11g, where we are operating , the channels are overlapped. Attributing to this overlap, a transmission on one channel becomes interference to stations on an overlapping channel, also known as adjacent channel interference This adjacent channel interference is considered a peril. In order to avoid this peril, two simultaneously communicating nodes that are in close proximity are assigned to different non-overlapping channels, i.e., channels 1, 6, and 11.This is the reason why the average RSSI is not enough. So we are compelled to use some rates-covering / percentage covering (Interference-factor) ,as the extent of overlap between 5 neighboring channels ,to compute the Final RSSI for each channel.(compute_final_RSSI()).

.



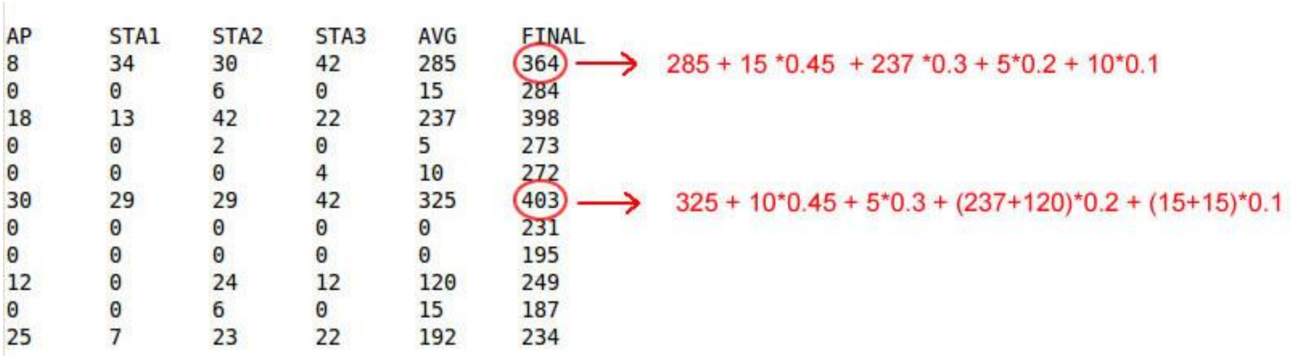
current channel : 6

```
AP      STA1    CONT1   AVG     FINAL
9       0       18      90      109
0       0       0       0       123
9       0       0       30      164
0       0       0       0       164
0       0       2       6       171
24      7       28      155     207
0       0       0       0       197
0       0       0       0       227
12      0       8       66      243
0       0       1       3       205
19      7       23      163     204
```

((24 + 7) / 2) *10)

((19 + 7 + 23) / 3) * 10)

Figure 10 shows how the average RSSI is computed



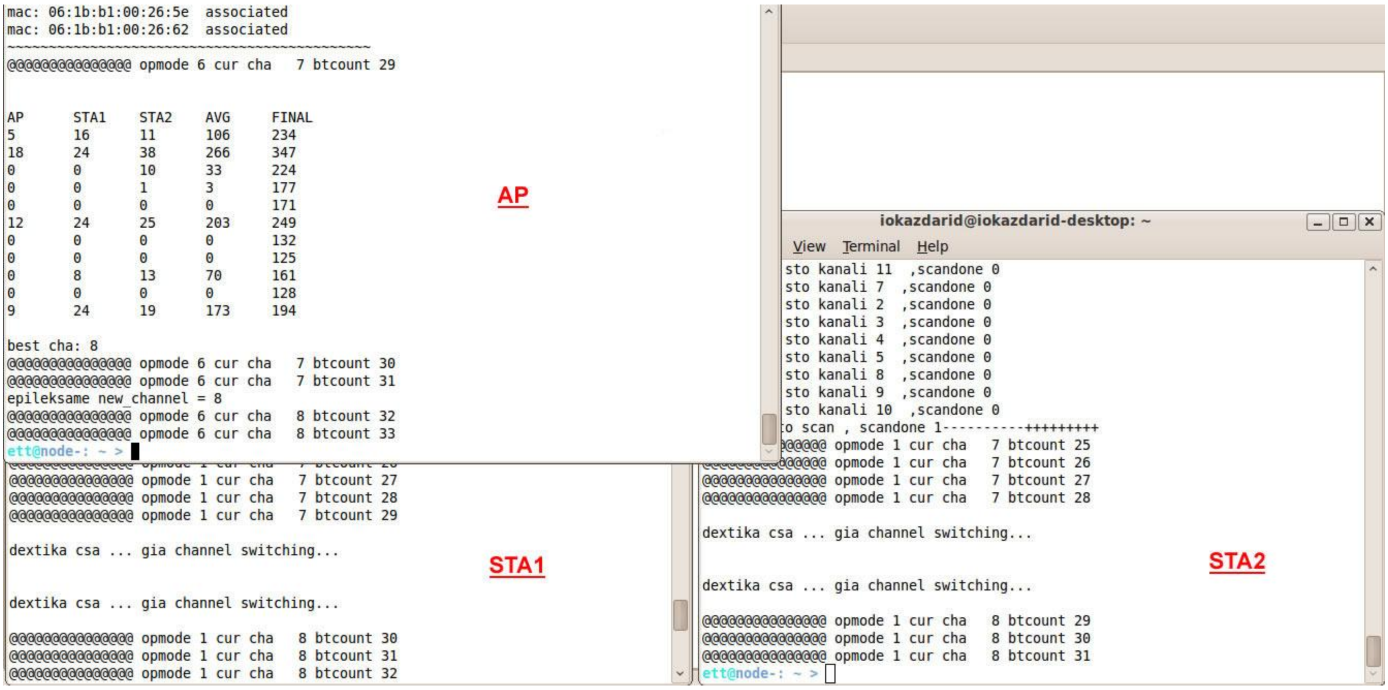
```
AP      STA1    STA2    STA3    AVG     FINAL
8       34      30      42      285     364
0       0       6       0       15      284
18      13      42      22      237     398
0       0       2       0       5       273
0       0       0       4       10      272
30      29      29      42      325     403
0       0       0       0       0       231
0       0       0       0       0       195
12      0       24      12      120     249
0       0       6       0       15      187
25      7       23      22      192     234
```

285 + 15 *0.45 + 237 *0.3 + 5*0.2 + 10*0.1

325 + 10*0.45 + 5*0.3 + (237+120)*0.2 + (15+15)*0.1

Figure 11 shows how the Final RSSI is computed

The AP based on its own measurements as well as the received measurements from the associated or contending stations, makes the channel switching decision. The channel switching decision can be better described in the following figure 12.(dfs_best_channel ()).



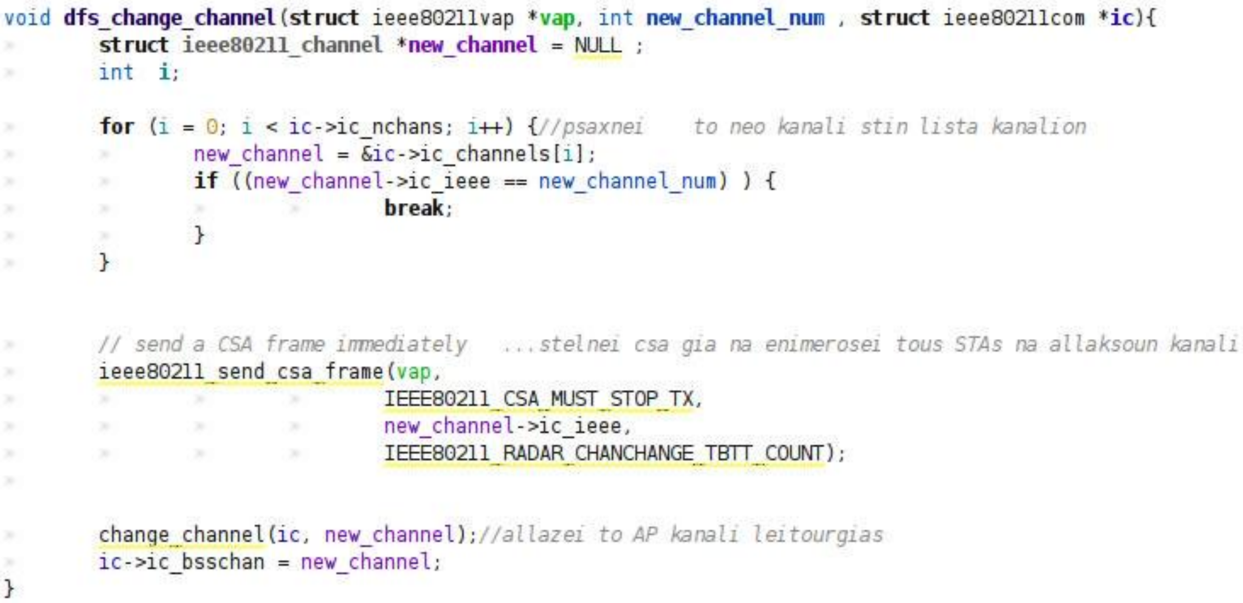
```
mac: 06:1b:b1:00:26:5e  associated
mac: 06:1b:b1:00:26:62  associated
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
@@@@@@@@@@@@@@@ opmode 6 cur cha   7 btcount 29


AP      STA1    STA2    AVG     FINAL
5       16      11      106     234
18      24      38      266     347
0       0       10      33      224
0       0       1       3       177
0       0       0       0       171
12      24      25      203     249
0       0       0       0       132
0       0       0       0       125
0       8       13      70      161
0       0       0       0       128
9       24      19      173     194

best cha: 8
@@@@@@@@@@@@@@@ opmode 6 cur cha   7 btcount 30
@@@@@@@@@@@@@@@ opmode 6 cur cha   7 btcount 31
epileksame new_channel = 8
@@@@@@@@@@@@@@@ opmode 6 cur cha   8 btcount 32
@@@@@@@@@@@@@@@ opmode 6 cur cha   8 btcount 33
ett@node-: ~ >
```

```
@@@@@@@@@@@@@@@ opmode 1 cur cha   7 btcount 27
@@@@@@@@@@@@@@@ opmode 1 cur cha   7 btcount 28
@@@@@@@@@@@@@@@ opmode 1 cur cha   7 btcount 29

dextika csa ... gia channel switching...


dextika csa ... gia channel switching...

@@@@@@@@@@@@@@@ opmode 1 cur cha   8 btcount 30
@@@@@@@@@@@@@@@ opmode 1 cur cha   8 btcount 31
@@@@@@@@@@@@@@@ opmode 1 cur cha   8 btcount 32
```

```
sto kanali 11  ,scandone 0
sto kanali 7  ,scandone 0
sto kanali 2  ,scandone 0
sto kanali 3  ,scandone 0
sto kanali 4  ,scandone 0
sto kanali 5  ,scandone 0
sto kanali 8  ,scandone 0
sto kanali 9  ,scandone 0
sto kanali 10  ,scandone 0
to scan , scandone 1----------+++++++++
@@@@@@@@@@@@@@@ opmode 1 cur cha   7 btcount 25
@@@@@@@@@@@@@@@ opmode 1 cur cha   7 btcount 26
@@@@@@@@@@@@@@@ opmode 1 cur cha   7 btcount 27
@@@@@@@@@@@@@@@ opmode 1 cur cha   7 btcount 28

dextika csa ... gia channel switching...


dextika csa ... gia channel switching...

@@@@@@@@@@@@@@@ opmode 1 cur cha   8 btcount 29
@@@@@@@@@@@@@@@ opmode 1 cur cha   8 btcount 30
@@@@@@@@@@@@@@@ opmode 1 cur cha   8 btcount 31
ett@node-: ~ >
```

Figure 12 channel switching decision

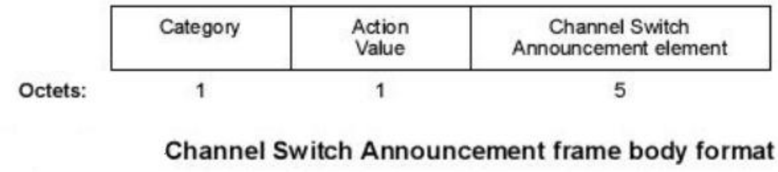
14.5. AP advertises the best channel to associated stations



```
void dfs_change_channel(struct ieee80211vap *vap, int new_channel_num , struct ieee80211com *ic){
        struct ieee80211_channel *new_channel = NULL ;
        int  i;

        for (i = 0; i < ic->ic_nchans; i++) {//psaxnei    to neo kanali stin lista kanalion
                new_channel = &ic->ic_channels[i];
                if ((new_channel->ic_ieee == new_channel_num) ) {
                                break;
                }
        }


        // send a CSA frame immediately   ...stelnei csa gia na enimerosei tous STAs na allaksoun kanali
        ieee80211_send_csa_frame(vap,
                                IEEE80211_CSA_MUST_STOP_TX,
                                new_channel->ic_ieee,
                                IEEE80211_RADAR_CHANCHANGE_TBTT_COUNT);


        change_channel(ic, new_channel);//allazei to AP kanali leitourgias
        ic->ic_bsschan = new_channel;
}
```

Figure 13 shows a pseudo code description of switching channel

After channel switching decision AP must advertise to the associated stations the new channel number by sending the channel switch announcement frame(dfs_change_channel()). The Channel Switch Announcement frame uses the Action frame body format . The format of the Channel Switch Announcement frame body is shown in Figure 13.



	Category	Action Value	Channel Switch Announcement element
Octets:	1	1	5

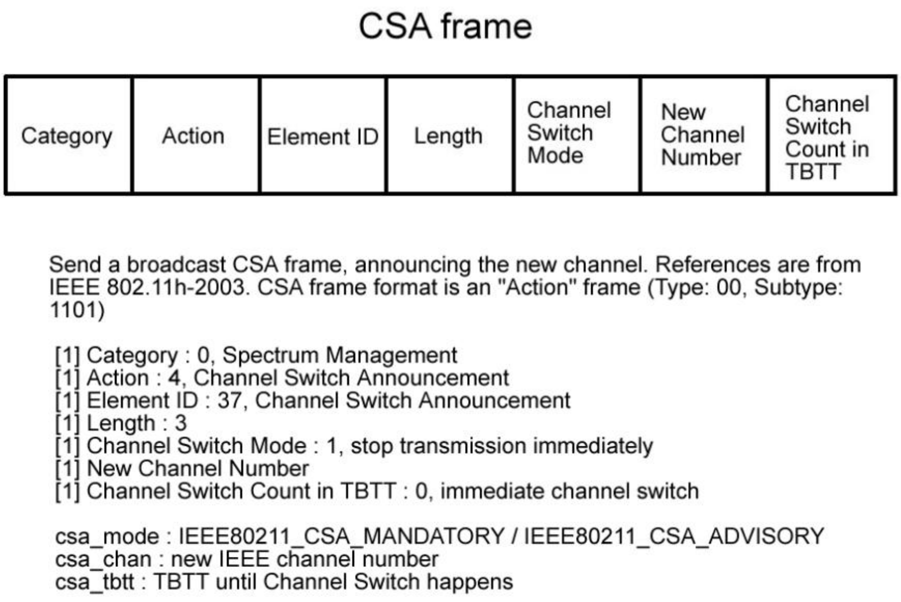
Channel Switch Announcement frame body format

Figure 14

The Category field is set to 1 (representing QoS).

The Action field is set to 3 (representing Schedule).

The format of the Channel Switch Announcement element is shown in Figure 14.



CSA frame

Category	Action	Element ID	Length	Channel Switch Mode	New Channel Number	Channel Switch Count in TBTT

Send a broadcast CSA frame, announcing the new channel. References are from IEEE 802.11h-2003. CSA frame format is an "Action" frame (Type: 00, Subtype: 1101)

[1] Category : 0, Spectrum Management
[1] Action : 4, Channel Switch Announcement
[1] Element ID : 37, Channel Switch Announcement
[1] Length : 3
[1] Channel Switch Mode : 1, stop transmission immediately
[1] New Channel Number
[1] Channel Switch Count in TBTT : 0, immediate channel switch

csa_mode : IEEE80211_CSA_MANDATORY / IEEE80211_CSA_ADVISORY
csa_chan : new IEEE channel number
csa_tbtt : TBTT until Channel Switch happens

Figure 1

The Length field shall be set to 3.

The Channel Switch Mode field indicates any restrictions on transmission until a channel switch. An AP in a BSS or a STA in an IBSS shall set the Channel Switch Mode field to either 0 or 1 on transmission. A Channel Switch Mode set to 1 means that the STA in a BSS to which the frame containing the element is addressed shall transmit no further frames within the BSS until the scheduled channel switch. A STA in an IBSS may treat a Channel Switch Mode field set to 1 as advisory. A Channel Switch Mode set to 0 does not impose any requirement on the receiving STA.

The New Channel Number field shall be set to the number of the channel to which the STA is moving

The Channel Switch Count field either shall be set to the number of TBTTs until the STA sending the Channel Switch Announcement element switches to the new channel or shall be set to 0. A value of 1 indicates that the switch shall occur immediately before the next TBTT. A value of 0 indicates that the switch shall occur at any time after the frame containing the element is transmitted.

The Channel Switch Announcement element is included in Channel Switch Announcement frames, and may be included in Beacon frames, and Probe Response frames.

15. EXPERIMENTAL CONFIGURATION

In order to evaluate the performance and study the behavior of the association scheme that we implemented, we used a large scale programmable testbed of wireless nodes, called NITOS. By testing the proposed scheme in a real large scale testbed, we were able to measure the performance under real conditions.

15.1. NITOS Testbed

NITOS (Network Implementation Testbed for using Open Source platforms) is a wireless testbed , that is designed to achieve reproducibility of experimentation. Users can perform their experiments by reserving slices (nodes, frequency spectrum) of the testbed through NITOS scheduler, that together with OMF management framework, support ease of use for experimentation and code development. It is remotely accessible and currently consists of 40 wireless nodes, outdoor located in a non-RF-isolated environment . The nodes are equipped with 2 wireless interfaces using Wistron CM9 - mPCI Atheros 802.11a/b/g 2.4 and 5 GHz cardsand , that run MAD-WiFi open source driver. NITOS is deployed at the exterior of the University of Thessaly campus building.

15.2. Experimental study and Results

15.2.1 Measurement Methodology

The throughput performance of the experiments, is measured by using Iperf , which is a powerful tool for traffic generation and measurement. A typical experimental setup for experiments considering only uplink transmissions, would be to run an Iperf client at each node, that act as Station, in order to generate traffic streams, having an Iperf server residing on the AP, receiving the traffic and collecting the measurements. To remove any random effect and short-term fluctuation, we run each experiment 5 times and each run lasts for 10 minutes at least. In order to get final results we average the results of the five experiments.

15.2.2 EXPERIMENTAL EVALUATION

Based upon the testbed described in before, numerous experiments were conducted, and the results obtained are reported and analyzed in this section. The first four experiments, were performed in two discrete phases. In the first phase, we use the unmodified MAD-WiFi driver, while in the second phase we used the modified driver that implements our mechanism.

15.2.2.1. Uplink Experiment 1

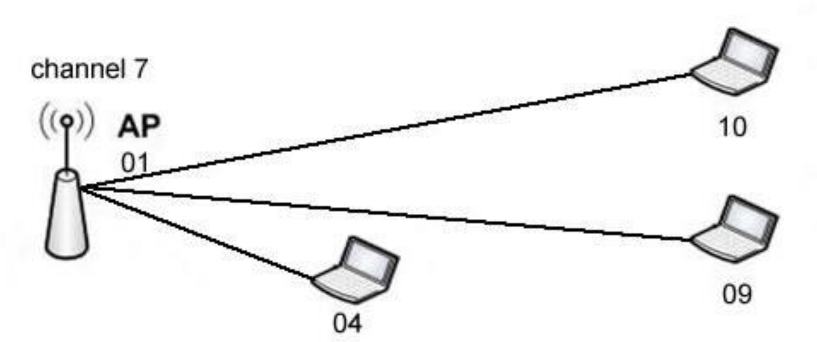


Figure 1

In this experiment, we set up a network that consists of one AP , node 1 and one STA, nodes 10. The station 10 generates UDP traffic, using 1 Iperf client , while the corresponding Iperf server runs at the AP. When STA associates with the AP, Iperf server starts receiving data and measuring the actual throughput. Figure 2 illustrates how the average throughput achieved per channel.

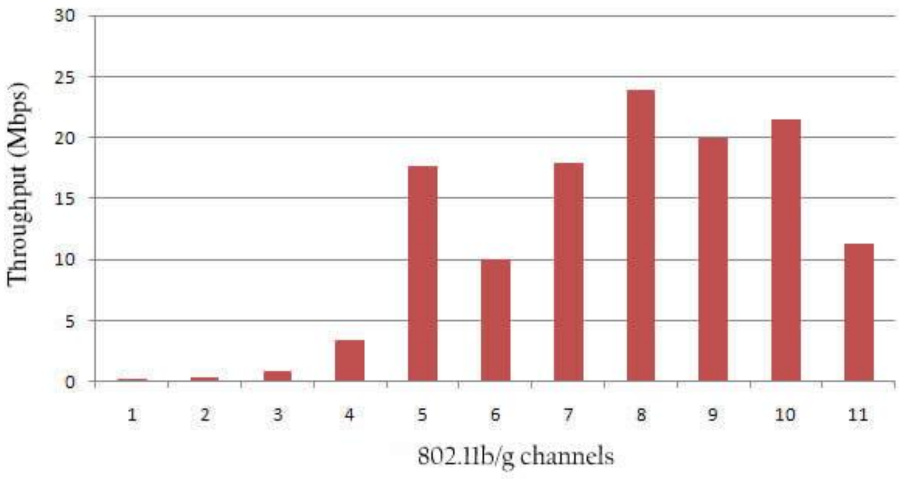
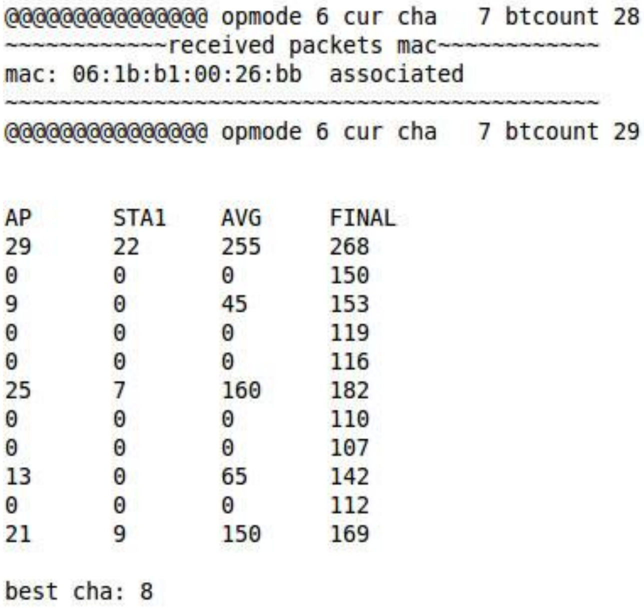


Figure 2

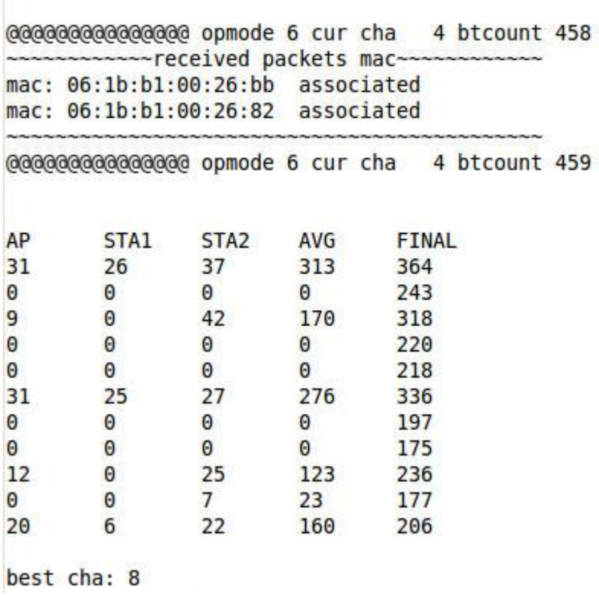
At   the first phase , the unmodified MAD-Wifi driver selects channel 7 to operate. At the second phase ,the AP based on the measurement results finds as best channel the frequency channel 8(Figure 3) . Notice in the Figure 2, that our mechanism succeeds much better throughput.



```
@@@@@@@@@@@@@@@ opmode 6 cur cha   7 btcount 28
~~~~~~~~~~~~received packets mac~~~~~~~~~~~~
mac: 06:1b:b1:00:26:bb  associated
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
@@@@@@@@@@@@@@@ opmode 6 cur cha   7 btcount 29


AP      STA1    AVG     FINAL
29      22      255     268
0       0       0       150
9       0       45      153
0       0       0       119
0       0       0       116
25      7       160     182
0       0       0       110
0       0       0       107
13      0       65      142
0       0       0       112
21      9       150     169

best cha: 8
```

Figure 3 shows the switching channel decision with 1 associated station

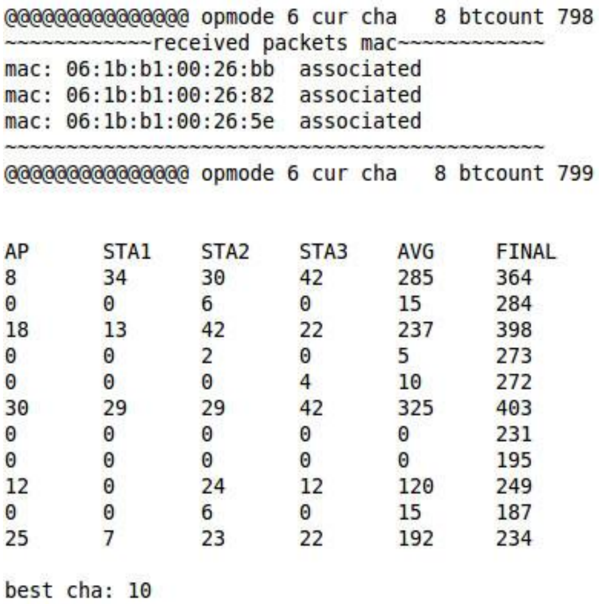
At this point, node 4 associates with the AP. Now AP   gets measurement packets from two stations. Based on the measurement results, finds as best channel the frequency channel 8(Figure 4).



```
@@@@@@@@@@@@@@@ opmode 6 cur cha   4 btcount 458
~~~~~~~~~~~~received packets mac~~~~~~~~~~~~
mac: 06:1b:b1:00:26:bb  associated
mac: 06:1b:b1:00:26:82  associated
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
@@@@@@@@@@@@@@@ opmode 6 cur cha   4 btcount 459


AP      STA1    STA2    AVG     FINAL
31      26      37      313     364
0       0       0       0       243
9       0       42      170     318
0       0       0       0       220
0       0       0       0       218
31      25      27      276     336
0       0       0       0       197
0       0       0       0       175
12      0       25      123     236
0       0       7       23      177
20      6       22      160     206

best cha: 8
```

Figure 4 shows the switching channel decision with 2 associated stations

Now, node 9 associates with the AP. Now AP   gets measurement packets from three stations. Based on the measurement results, finds as best channel the frequency channel 10(Figure 5).



```
@@@@@@@@@@@@@@@ opmode 6 cur cha   8 btcount 798
~~~~~~~~~~~~received packets mac~~~~~~~~~~~~
mac: 06:1b:b1:00:26:bb  associated
mac: 06:1b:b1:00:26:82  associated
mac: 06:1b:b1:00:26:5e  associated
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
@@@@@@@@@@@@@@@ opmode 6 cur cha   8 btcount 799


AP      STA1    STA2    STA3    AVG     FINAL
8       34      30      42      285     364
0       0       6       0       15      284
18      13      42      22      237     398
0       0       2       0       5       273
0       0       0       4       10      272
30      29      29      42      325     403
0       0       0       0       0       231
0       0       0       0       0       195
12      0       24      12      120     249
0       0       6       0       15      187
25      7       23      22      192     234

best cha: 10
```

Figure 5 shows the switching channel decision with 3 associated stations

According to Figure 2, normal Mad-WiFi reaches the throughput value of 17.85Mbps.If the operating channel   is randomly selected by the user then reaches 11,54 Mbps. The modified driver with our mechanism reaches   23.927 Mbps for channel 8 and 21.477 Mbps for channel 10. We conclude that   the proposed algorithm upgrades the link quality.

15.2.2.2 Uplink Experiment 2

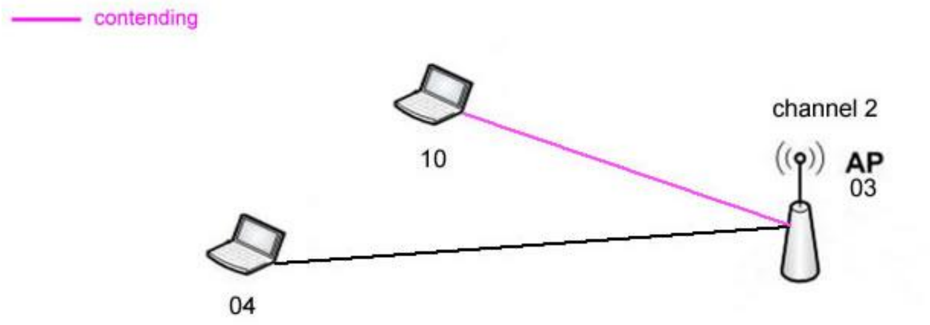


Figure 6

In this experiment, we set up a network that consists of one AP , node 3 and one STA, node 4 . The station 4 generates UDP traffic, using 1 Iperf client, while the corresponding Iperf server runs at the AP. When the STA associates with the AP, Iperf server starts receiving data and measuring the actual throughput. Figure 7 illustrates how the average throughput achieved per channel.

At the first phase , the unmodified MAD-WiFi driver selects channel 2 to operate. At the second phase , the AP based on the measurement results finds as best channel the frequency channel 4(Figure 8).As we can see from the Figure 7 the link quality is upgraded a lot.

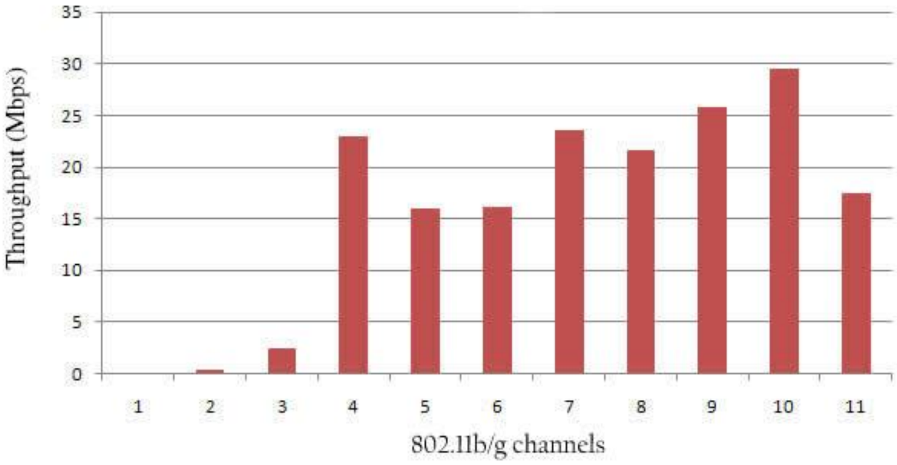
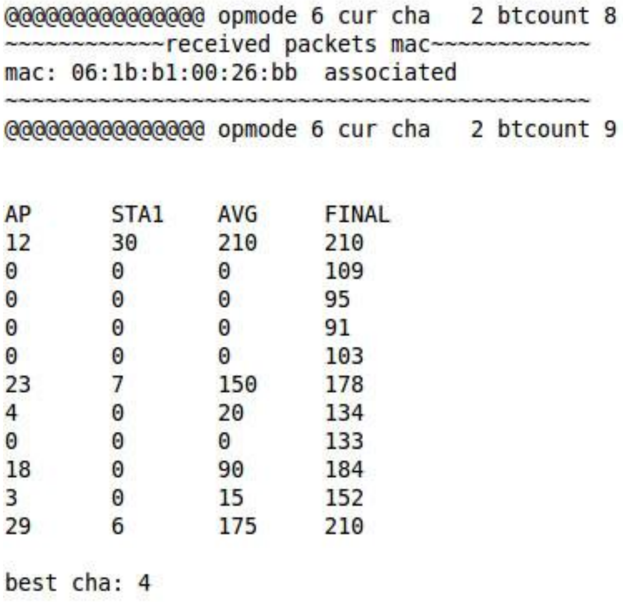


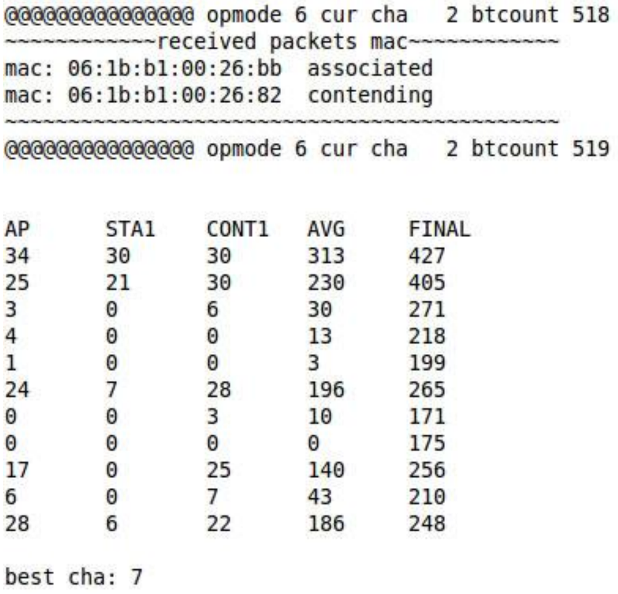
Figure 7

We repeat the first phase and the normal MAD-WiFi driver selects channel 2 again. At this point, we set up a new bss on channel 2 .In this way node 3(AP) will be able to get measurement packets from a contending station(node 10). Now , our mechanism selects channel 7 ,which has a little better link quality(Figure 9).



```
@@@@@@@@@@@@@@@ opmode 6 cur cha   2 btcount 8
~~~~~~~~~~~~received packets mac~~~~~~~~~~~~
mac: 06:1b:b1:00:26:bb  associated
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
@@@@@@@@@@@@@@@ opmode 6 cur cha   2 btcount 9


AP      STA1    AVG     FINAL
12      30      210     210
0       0       0       109
0       0       0       95
0       0       0       91
0       0       0       103
23      7       150     178
4       0       20      134
0       0       0       133
18      0       90      184
3       0       15      152
29      6       175     210

best cha: 4
```

Figure 8 shows the switching channel decision with 1 associated station



```
@@@@@@@@@@@@@@@ opmode 6 cur cha   2 btcount 518
~~~~~~~~~~~~received packets mac~~~~~~~~~~~~
mac: 06:1b:b1:00:26:bb  associated
mac: 06:1b:b1:00:26:82  contending
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
@@@@@@@@@@@@@@@ opmode 6 cur cha   2 btcount 519


AP      STA1    CONT1   AVG     FINAL
34      30      30      313     427
25      21      30      230     405
3       0       6       30      271
4       0       0       13      218
1       0       0       3       199
24      7       28      196     265
0       0       3       10      171
0       0       0       0       175
17      0       25      140     256
6       0       7       43      210
28      6       22      186     248

best cha: 7
```

Figure 9 shows the switching channel decision with associated/contending stations

According to Figure 7, normal Mad-WiFi reaches the throughput value of 323 Kbps. If the operating channel is randomly selected by the user then reaches 16Mbps .The modified driver with our mechanism reaches 23Mbps on channel 4 and 23.6Mbps on channel 7. Notice that , the proposed algorithm upgrades the link quality.

15.2.2.3 Downlink Experiment 1

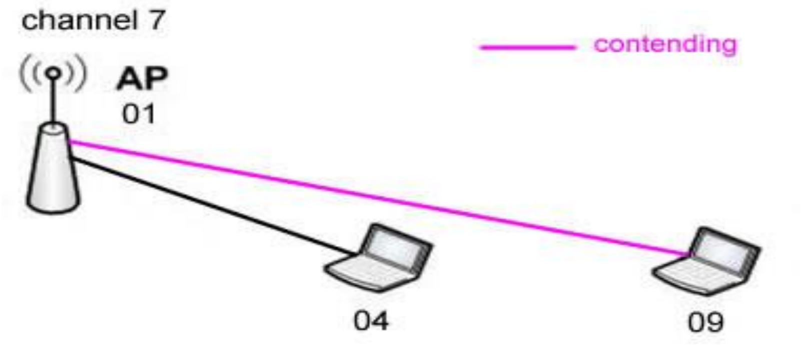


Figure 10

In this experiment, we set up a network that consists of one AP , node 1 and one STA, node 4 . The AP   generates UDP traffic, using 1 Iperf client , while the corresponding Iperf server runs at the station. When the STA associates with the AP, Iperf server starts receiving data and measuring the actual throughput. Figure 11 illustrates how the average throughput achieved per channel.

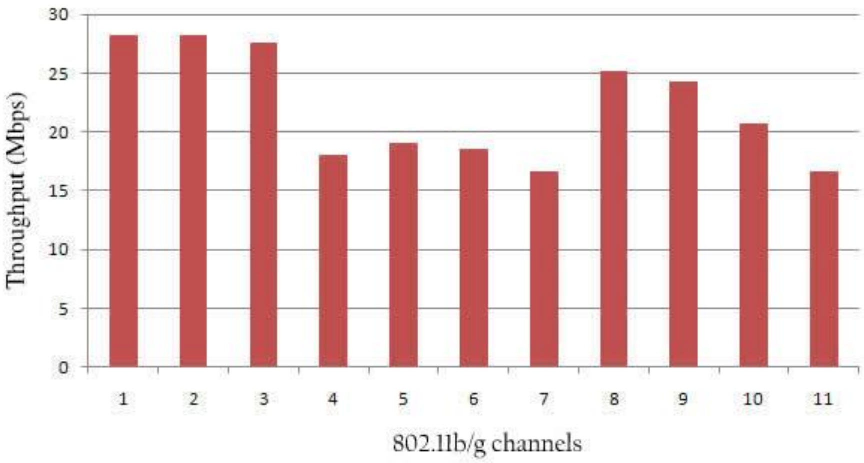
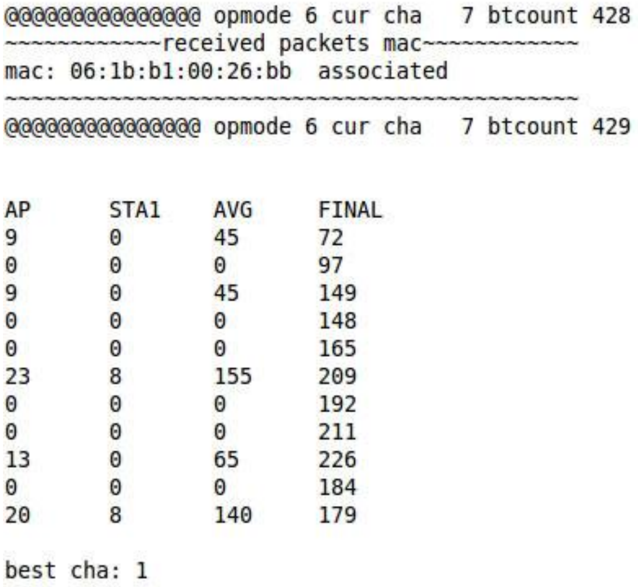


Figure 11

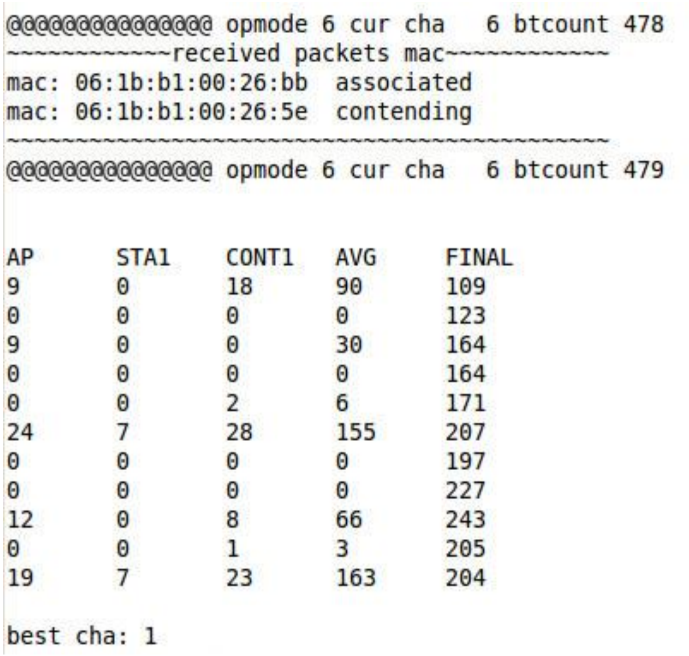
At the first phase , the unmodified MAD-Wifi driver selects channel 7 to operate. At the second phase , the AP based on the measurement results finds as best channel the frequency channel 1(Figure 12).As we can see from the Figure 11 the link quality is upgraded.



```
@@@@@@@@@@@@@@@ opmode 6 cur cha   7 btcount 428
~~~~~~~~~~~~received packets mac~~~~~~~~~~~~
mac: 06:1b:b1:00:26:bb  associated
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
@@@@@@@@@@@@@@@ opmode 6 cur cha   7 btcount 429


AP      STA1    AVG     FINAL
9       0       45      72
0       0       0       97
9       0       45      149
0       0       0       148
0       0       0       165
23      8       155     209
0       0       0       192
0       0       0       211
13      0       65      226
0       0       0       184
20      8       140     179

best cha: 1
```

Figure 12 shows the switching channel decision with 1 associated station

We repeat the first phase and the normal MAD-Wifi driver selects channel 7 again. At this point, we set up a new bss on channel 7 .In this way node 3(AP) will be able to get measurement packets from a contending station(node 9). Now , our mechanism selects channel 1 ,again(Figure 13).



```
@@@@@@@@@@@@@@@ opmode 6 cur cha   6 btcount 478
~~~~~~~~~~~~received packets mac~~~~~~~~~~~~
mac: 06:1b:b1:00:26:bb  associated
mac: 06:1b:b1:00:26:5e  contending
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
@@@@@@@@@@@@@@@ opmode 6 cur cha   6 btcount 479


AP      STA1    CONT1   AVG     FINAL
9       0       18      90      109
0       0       0       0       123
9       0       0       30      164
0       0       0       0       164
0       0       2       6       171
24      7       28      155     207
0       0       0       0       197
0       0       0       0       227
12      0       8       66      243
0       0       1       3       205
19      7       23      163     204

best cha: 1
```

Figure 13 shows the switching channel decision with associated/contending stations

According to Figure 11, normal MAD-Wifi reaches the throughput value of 16.6 Mbps .If the operating channel is randomly selected by the user then reaches 22.1Mbps .The modified driver with our mechanism reaches 28.2 Mbps on channel 1 and. Notice that , the proposed algorithm upgrades the link quality.

15.2.2.4 Uplink Experiment 3

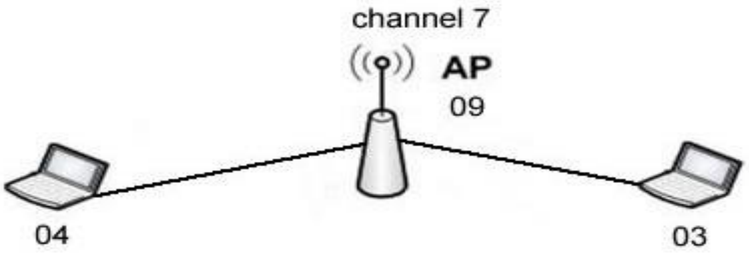


Figure 14

In this experiment, we set up a network that consists of one AP , node 9 and two STAs, node 4 , 3 . The stations 4 , 3 generate UDP traffic, using 2 Iperf clients that run simultaneously, while the corresponding Iperf server runs at the AP. When the STAs associate with the AP, Iperf server starts receiving data and measuring the actual throughput. Figure 15 illustrates how the average throughput , for each station, achieved per channel.

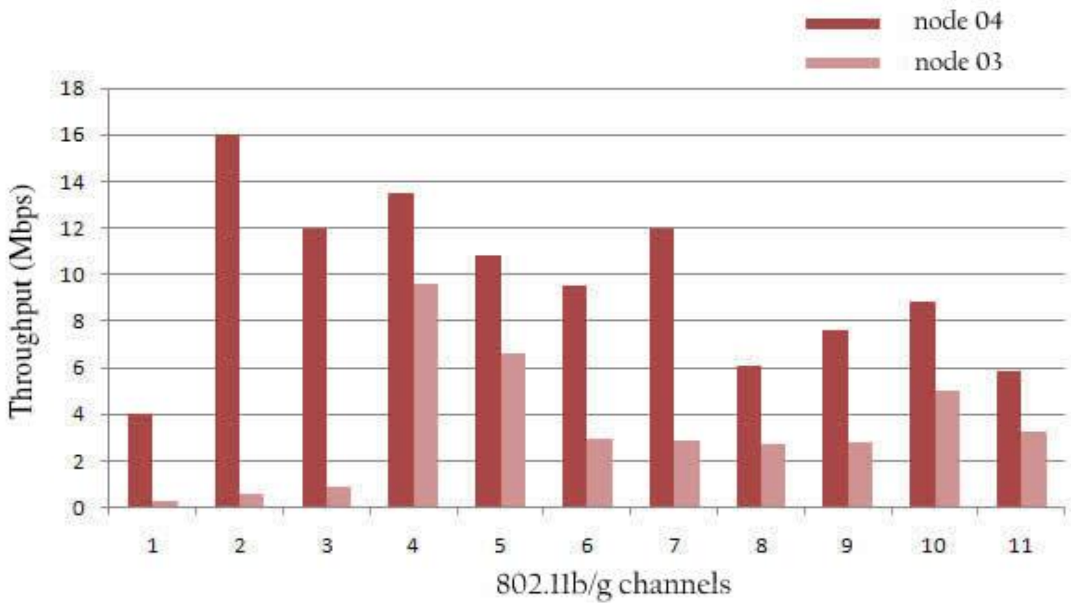
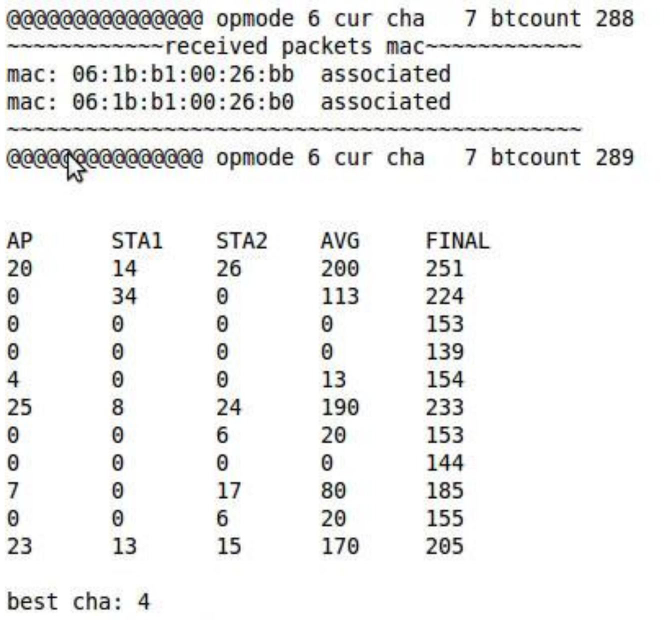


Figure 15

At the first phase , the unmodified MAD-Wifi driver selects channel 7 to operate. At the second phase ,the AP based on the measurement results finds as best channel ,the frequency channel 4.As we can see from the Figure 15 the link quality is upgraded.



```
@@@@@@@@@@@@@@@ opmode 6 cur cha   7 btcount 288
~~~~~~~~~~~~received packets mac~~~~~~~~~~~~
mac: 06:1b:b1:00:26:bb  associated
mac: 06:1b:b1:00:26:b0  associated
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
@@@@@@@@@@@@@@@ opmode 6 cur cha   7 btcount 289


AP      STA1    STA2    AVG     FINAL
20      14      26      200     251
0       34      0       113     224
0       0       0       0       153
0       0       0       0       139
4       0       0       13      154
25      8       24      190     233
0       0       6       20      153
0       0       0       0       144
7       0       17      80      185
0       0       6       20      155
23      13      15      170     205

best cha: 4
```

Figure 16 shows the switching channel decision with 2 associated stations

According to Figure 15, with normal MAD-Wifi node 4 reaches the throughput value of 9.51 Mbps and node 3 reaches 2.95 Mbps. If the operating channel is randomly selected by the user then node 4 reaches 9.96 Mbps and node 3 reaches 3.41 Mbps. With the modified driver node 4 reaches 13.5 Mbps   and node 3 reaches 9.61 Mbps ,on channel 4. Notice that , the proposed algorithm upgrades the link quality and the station 3 has the best throughput value among the available channels.

15.2.2.5 Uplink Experiment 4

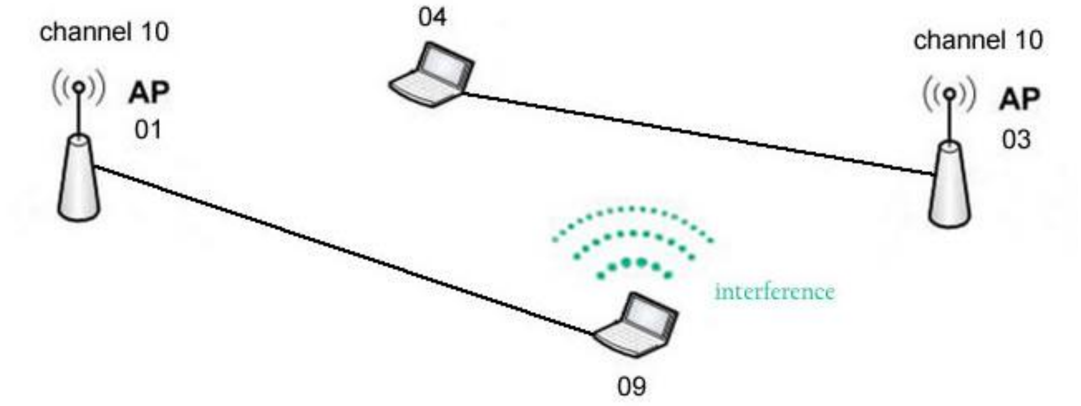


Figure 17

In this experiment, we set up a network on channel 10,that consists of two bss.Bss1 has one AP , node 3 and one STA, node 4 .Bss2 has one AP , node 1 and one STA, node 9. The stations 4 , 9 generate UDP traffic, using two Iperf clients that run simultaneously, while the corresponding Iperf servers runs at the APs. When the STAs associate with the APs, Iperf server start receiving data and measuring the actual throughput. Figure 18 illustrates how the average throughput for bss1, achieved per interference rate. As clearly shown in the Figure 18, when the interference rate is increasing the link quality is degraded.

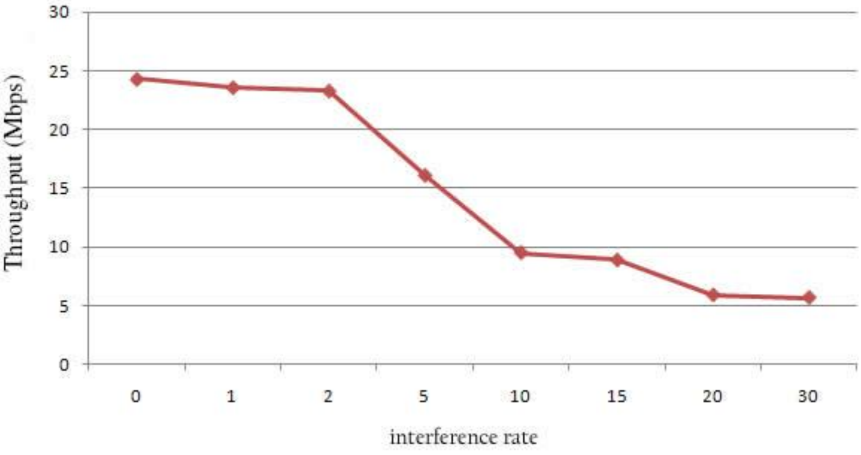


Figure 18

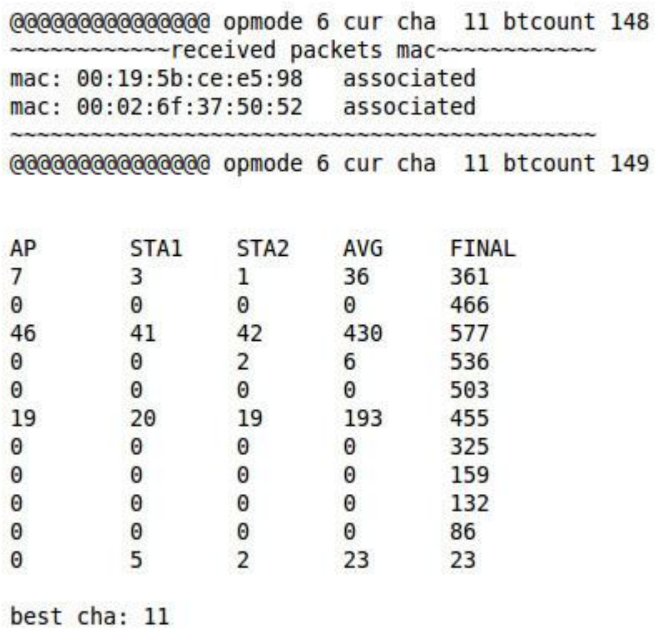
15.2.2.6 MALFUNCTION

In this experiment, we set up a network that consists of one AP , node 1 and one STA, node 4 . At this point ,a new AP, which sends only beacon frames (no traffic), appears on the operating channel. As a result, node 1 gets a high RSSI measurement for that channel and is forced to change channel.

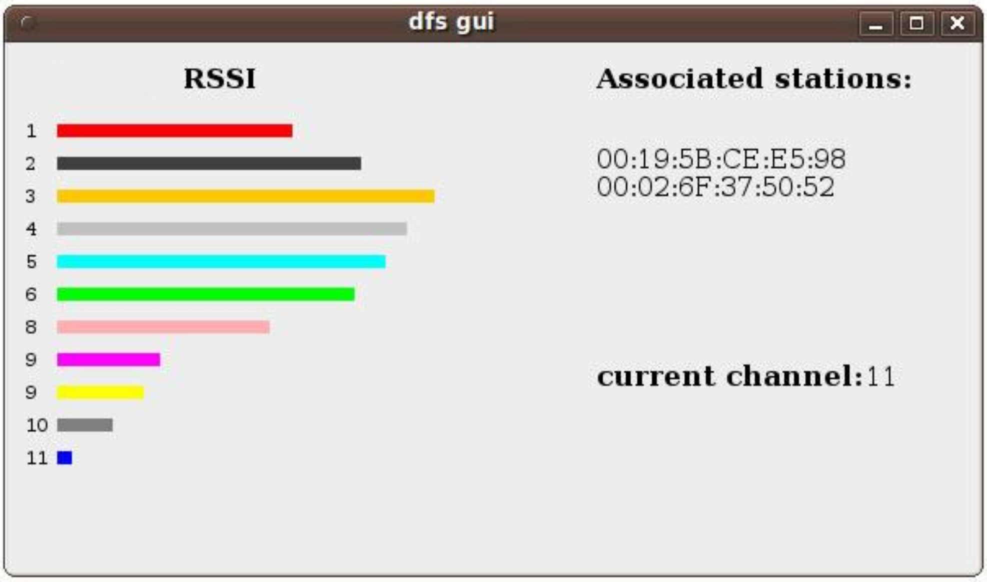
Now on the new operating channel , is possible link quality to be worse because of the traffic on that channel or on the neighboring channels. As we have said in previous chapters , a transmission on one channel becomes interference to stations on an overlapping channel.

16. DFS GUI implementation

The following graph is a Gui , written in java swing for our DFS implementations in Madwifi. The Gui reads proxies files and repaints every 5 seconds final RSSI on every channel, associated stations Mac address and the operating channel number**.**



```
@@@@@@@@@@@@@@@@ opmode 6 cur cha  11 btcount 148
~~~~~~~~~~~~received packets mac~~~~~~~~~~~~
mac: 00:19:5b:ce:e5:98   associated
mac: 00:02:6f:37:50:52   associated
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
@@@@@@@@@@@@@@@@ opmode 6 cur cha  11 btcount 149


AP       STA1    STA2    AVG     FINAL
7        3       1       36      361
0        0       0       0       466
46       41      42      430     577
0        0       2       6       536
0        0       0       0       503
19       20      19      193     455
0        0       0       0       325
0        0       0       0       159
0        0       0       0       132
0        0       0       0       86
0        5       2       23      23

best cha: 11
```



Associated stations:

00:19:5B:CE:E5:98
00:02:6F:37:50:52

current channel:11

17. Future work

There are few improvements that we have thought:

- we could use the number of AP that we found on each channel from the background scanning , for the switching channel decision.
- we could use the rssi value we received measurements packets from contending station ,to measure interference on our operating channel.
- with our mechanism through contending measurements AP discovers new stations on the operating channel. If we Broadcast measurements packets on all channels, Ap could discover all existed Stations and their interference.
- a possible extension is to add throughput information on beacon frame , so when a node receives beacon knows the link quality on the channel and decides to switch in the best channel.

REFERENCES

[1] Effect of background scan on performance of neighbouring channels in 802.11 networks, Gurpal Singh* and Ajay , Pal Singh Atwal

[2] MadWi_ Driver Summary , Wenhua Zhao

[3] Improving the IEEE 802.11 MAC Layer Handoff Latency to Support Multimedia Traffic, Yogesh Ashok Powar and Varsha Apte

[4] IEEE 802.11h: Technology and Application ,Daji Qiao , Sunghyun Choi

[5] Exploiting Partially Overlapping Channels in Wireless Networks: Turning a Peril into an Advantage,Arunesh Mishra , Eric Rozner , Suman Banerjee , William Arbaugh

[6] Converting Signal Strength Percentage to dBm Values , Joe Bardwell

[7] A capacity analysis for the IEEE802.11 MAC Protocol , Y.C Tay , K.C.Chua

[8] Compatibility between IEEE 802.11b and IEEE 802.11g networks: Impact on throughput , Izaskun Pellejero , Fernando Andreu , Asier Barbero , Amaia Lesta