Seamless Hand-over Algorithm for Wireless Enterprise Networks

by

Gjermund Hodnebrog
Qi Jin

Agder University College
Faculty of Engineering and Science

Grimstad, Norway
May 2007

Abstract

The mass deployment of the Institute of Electrical and Electronics Engineers' (IEEE) 802.11 based wireless local area networks (WLAN) and increased sales in hand-held devices supporting WLAN have resulted in an urgent need to support fast WLAN handovers or roaming. The reason for this problem arising now is that hand-held devices are more mobile than a laptop and their users actively use their hand-held equipment while moving. Laptop users are often called nomadic users in contrast to the real mobile users.

The customarily solution is that when a connection is lost with the associated access point, one tries to find a new access point and tries to connect to it. The process of finding a new access point and connecting to it takes too long time in current implementations. Some applications cannot tolerate to be interrupted or disconnected for a very long time period before the session breaks. Therefore we need mechanisms to make sure that the disconnection time is as low as possible.

Our algorithm uses an improved threshold scheme to detect the handover. The algorithm avoids many unnecessary handovers and prevents rapidly dropped signal strength or poor connection quality. In addition, we do scanning and AP selection before critical situations occur and therefore are faster in disconnecting from the current AP. As a result, the whole disconnection time is only the handover execution time, which is much shorter than the customarily one's. Furthermore, we use signal strength, hysteresis and trends to classify the candidate APs. The result allows us to choose the best one of them and then switch to it. Through thresholds and hysteresis based decisions we avoid the latent unnecessary handovers resulting in a undesired "yoyo" effect, where the client continuously jumps back and forth between APs.

Our handover algorithm is signal strength based. For technical reasons, the signal strength is the main parameter we considered. In the future, several other quality parameters can be implemented into our algorithm to make the algorithm even more efficient, e.g., by querying APs about their current load and QoS resources.

Preface

This thesis finishes a two-year Master of Science program in Information- and Communication-Technology (ICT) at Agder University College (AUC), Faculty of Engineering and Science in Grimstad, Norway. The workload of this thesis equals 30 ECTS and the project has been carried out from January till May 2007.

First of all, we would like to thank Professor Dr. Frank Reichert, our internal supervisor at Agder University College, for excellent supervision and guidance throughout the project period. As this thesis was given by Ericsson Mobile Platforms in Grimstad, Norway there are several people at EMP we would like to thank. Firstly we like to thank Svein Thorstensen who gave us the opportunity to work with this thesis and our external supervisor Sverre Vegge for arranging meetings and advices. We also owe thanks to Kjetil Asdal, Arild Løvendal, Øyvind Murberg and Kim Lilliestierna for nice and helpful technical discussions due to the challenges and solutions of the project.

Finally we would like to thank Head of Studies, Stein Bergsmark, for his contributions, and our co-students for helpful feedback on our thesis.

Grimstad, May 2007

Gjermund Hodnebrog

Qi Jin

Table of Contents

Table List

Figure List

1 Introduction

1.1 Background

The mass deployment of the Institute of Electrical and Electronics Engineers' (IEEE) 802.11 [1] based wireless local area networks (WLAN) and increased sales in hand-held devices supporting WLAN has resulted in an urgent need to support fast WLAN handovers/roaming. The reason for this problem arising now is that hand-held devices are more mobile than a laptop and their users actively use their hand-held equipment while moving. Laptop users are often called nomadic users in contrast to the real mobile users.

The customarily solution is that when a connection is lost with the associated access point (AP), one tries to find a new access point and tries to connect to it. There are no smart mechanisms to decide when to leave the current AP or how to select a new AP. The process of finding a new access point and connecting to it takes too long time in current implementations. Some applications cannot tolerate to be interrupted or disconnected for a very long time period before the session breaks. Therefore we need mechanisms to make sure the disconnection is as low as possible. There are several organisations working on this problem such as IEEE [2] and Internet Engineering Task Force (IETF) [3].

In this master thesis we will study different solutions to this problem and then choose the solution we consider best suitable and make an implementation of it. We will also study handover algorithms, compare and test the different properties of such algorithms and finally design our own fast WLAN handover algorithm for hand-held applications based on the knowledge obtained.

The idea for the thesis came from Ericsson Mobile Platforms (EMP) and is carried out as cooperation between Agder University College (AUC) and EMP.

1.2 Thesis Definition

"Background:
Historically seamless disconnectionless mobility between AP's in a WLAN network has not been a real issue. WLAN has primarily been used in conjunctions with laptops, where a temporary disconnection is at most a nuisance, and the use pattern is more of "temporarily stationary".

With the advent of WLAN in to lighter portable equipment such as mobile phones the requirement of seamless mobility between AP's is of much higher concern.

WLAN has not historically been designed for seamless transition between AP's and it is customarily solved by disconnecting from the current AP; scanning for a new AP and then trying to connect to newly found AP. This is not an acceptable procedure for various reasons:

1. *The requirement for acceptable voice transmissions does not allow for disconnections in the data stream in excess of 50 ms. The time a disconnect / reconnect procedure takes can be several hundreds of milliseconds long. When the more advanced forms of authentication and encryption is used, requiring the exchange of key and*

authentication information with a remote authentication server, the times can stretch out into the second's range.

2. *Having no knowledge of the current connection quality of the destination AP, it might turn out that the new AP offers a much lower through put than the one the station left.*
3. *The simple "scan for an AP and connect" scenario can lead to yoyo effects where the station is continuously jumping back and forth between the same two AP's.*

Some questions:
How do new standards such as Ref. IEEE 802.11r affect a faster seamless WLAN roaming solution? Can the 50 ms roaming be achieved?

What information is needed about the WLAN network in order to support a fast roaming algorithm? Based on this could a roaming algorithm be implemented and controlled by an application or is it only applicable closer to the physical level?

EMP would like a suggestion on roaming algorithms based on the previous questions and an overview indicating their different properties related to handheld applications.

Ericsson Mobile Platforms is a leading platform supplier for GSM/GPRS, EDGE and WCDMA platforms. Ericsson Mobile Platforms was one of the first companies in the world to license open-standard GSM/GPRS, EDGE and WCDMA technology platforms to manufactures of mobile phones and other mobile communication devices. Ericsson Mobile Platforms' reference designs are used in 6 of the 10 leading 3G phone manufacturers such as SonyEricsson, …"

1.3 Motivation

Firms often have an enterprise wireless network that consists of several access points that form an extended service set (ESS) as illustrated in figure 1-1. The main case is when a user is moving across the different access points. If the user is moving away from AP1, the connection with AP1 will deteriorate and at some time the wireless terminal should change to another access point.

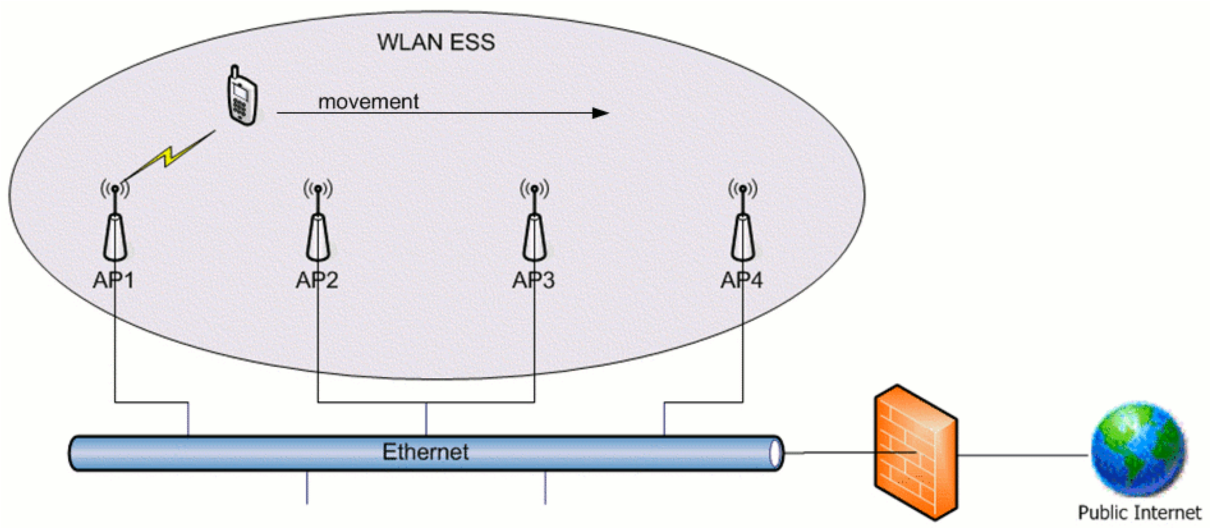


Figure 1-1: An enterprise wireless network

Packet loss due to bad wireless links, searching for other access point and the handover is the motivation. Why will be described next in the three use case scenarios for WLAN on hand-held devices.

1.3.1 VoIP

Voice-over-IP solutions based on Session Initiation Protocol (SIP) [4] and Skype [5] are now widely spread and accepted by the end-users. Today, the user can use VoIP on his or her cell phone directly instead of a laptop. VoIP solutions transfer voice over the Internet or through any other IP-based network. The quality and overall reliability of the connection is entirely dependent upon the reliability and speed of the Internet connection which is used. In our case, it means that if the handover delay it too long, the quality of the VoIP session (telephone call) will be affected. The user will experience this as a click or that there is a break in the conversation. The worst case would be that the handover delay is so long that the whole telephone call is terminated. VoIP requires relatively low bandwidth (64 kbit/s or less [6]), but because it is a real-time application VoIP is very sensitive to handover delays that lead to packet loss.

1.3.2 Video Streaming

Besides audio data transmission to a cell phone in a WLAN, video data transmission is also supported. The user can watch a movie or the news on his or her cell phone. Mobile television is a service that already exists across the 3G network and similar services across a WLAN connection are likely to appear.

Multimedia data packets usually have a large size and therefore require high bandwidth (between 200-600 kbit/s). Because video streams are one way communication and not two way like a telephone conversation, the real-time requirements are not that essential. Therefore packets are downloaded into buffers to cope with jitter and packet loss. The buffers can deal with the packet loss to some extent, but if the handover delay is too long, the buffers will drain and there will be a disruption or break in the video feed.

1.3.3 File Transfer

The last use case is file transfer which includes web browsing. All cell phones released the last couple of years come with a web browser. Because browsing has a burst type traffic pattern, the user accepts some waiting time, but packet loss may cause parts of a webpage to be missing. In this case the user has to manually refresh the webpage to get the complete content.

The reason for choosing these use cases - VoIP, video streaming and file transfer - is because they have different requirements. By considering these use cases we cover the extreme points of application requirements.

	Real-time	**Bandwidth**	**Packet loss sensitivity**
VoIP	High	Low	Low
Video streaming	Medium	High	Low
File transfer	Low	Medium	Medium

Table 1-1: Application requirements

The point of having WLAN in hand-held devices will be lost if nobody uses it. If disruptions and broken sessions are a big problem people will not be bothered to use the WLAN interface. The positive sides of WLAN are the high bandwidth (11-54 Mbit/s and ascending) and the low costs of using it. The handover issue is therefore an important problem to solve.

1.4 Problem Statement

There are several steps in a handover procedure, all of which have different difficulties. We have divided the handover into three main phases; the detection phase, selection phase and execution phase. The detection phase is to decide when to handover, the selection phase is to plan where to handover to and the execution phase is to carry out the actual handover process. When the connection quality is OK (see figure 1-2), the detection phase is active and checks the connection quality with fixed intervals. When the quality drops below a certain level something has to be done and the selection phase is activated. The execution phase is the last one and should be completed before we reach the “too late” line in figure 1-2.

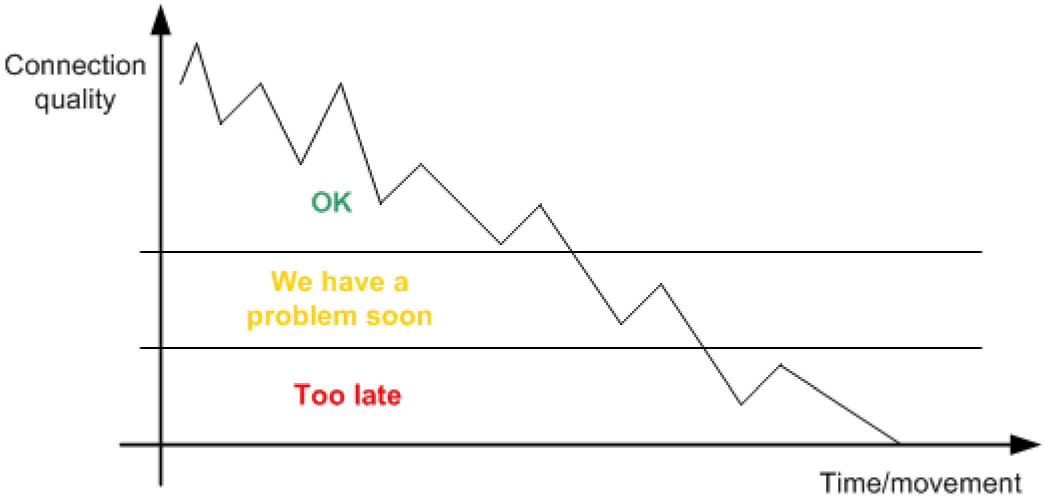


Figure 1-2: Connection quality graph

The graph of the connection quality will probably be jagged and may have to be smoothed out to avoid ping-pong effect between the two first phases. All three phases are described in more detail in the next three subchapters.

1.4.1 Detection Phase

When is the connection with the current access point too poor? This is what the detection phase is all about. In figure 1-3 the terminal is connected to AP 1, but AP 2 is actually much closer. The terminal here should be connected to AP 2.

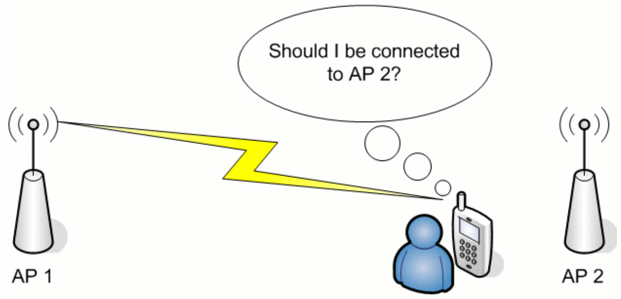


Figure 1-3: Which AP to choose 1

The physical distance is not always helpful information for the terminal. For example, if there was a thick brick wall between the terminal and AP 2 (see figure 1-4), maybe AP 1 might be the better choice.

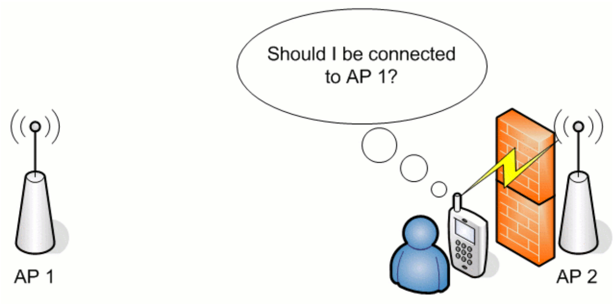


Figure 1-4: Which AP to choose 2

Many factors can be considered in evaluating the connection quality, but we have to relate to the factors that can be measured. The terminal can not see the brick wall, but it can detect the data rate, measure the signal strength, detect packet loss and retransmissions, measure noise and compute the signal over noise ratio (S/N).

The big issue is to decide when the connection is too poor. Is it when the data rate has dropped from 54 Mbit/s to 24 Mbit/s or maybe when the S/N is under a certain level? Should these levels be static or dynamic? One problem that is often discussed is the yoyo effect. For example if the signal strength of the AP you decided to leave is almost the same as the signal strength of the AP you are connecting to then the yoyo effect can occur. This effect occurs when the terminal switches forth and back between the two access points, and it results in a big waste of resources. Therefore it is very important to set the levels correctly for deciding when the connection is too poor correctly.

1.4.2 Selection Phase

The selection phase involves scanning for new access points and deciding which one to choose. Scanning for new access points is the most power consuming process in a handover procedure. It is a given that power consumption is a big concern to all vendors of mobile equipment and the scanning process must therefore not be performed more than necessary.

Because the WLAN interface might be in use by an application, the scanning must take place when the application is not usingt. Voice payload size is normally 20 or 30 ms [6]. For a VoIP session with 20 ms voice payload size a data packet is transferred every 20 ms. In a worst case scenario the voice payload size is 160 bytes [6], IP, UDP and RTP headers are 40 bytes [6] and the WLAN (802.11b) headers and tails are 70 bytes [7]. If the WLAN link is poor, the usable data rate might not be more than 0.5 Mbit/s. A total of 270 bytes is transferred on the 0.5 Mbit/s link in about 4 ms. There would now be just 16 seconds left for the scanning to run (see figure 1-5).

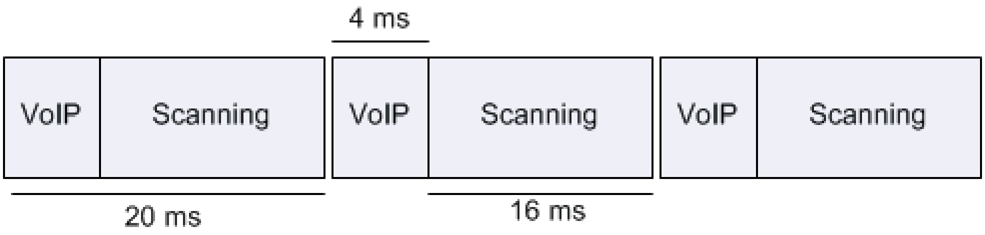


Figure 1-5: Scanning time periods

Once the scanning is done, one has to select which of the discovered access points to connect to (see figure 1-6).

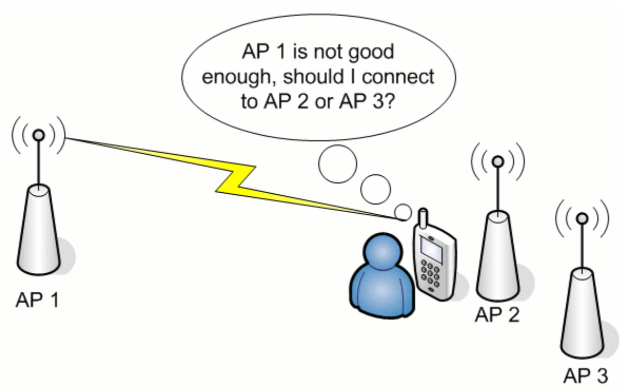


Figure 1-6: Select new AP

Which access point to choose can be based on several different parameters, such as Service Set Identifier (SSID), signal strength, the network topology, traffic load or AP features like QoS and security. The SSID is of course important for staying in the same network, and not trying to connect to a network one does not have access to. For instance, if two companies share the same building but are located on different floors (see figure 1-7).

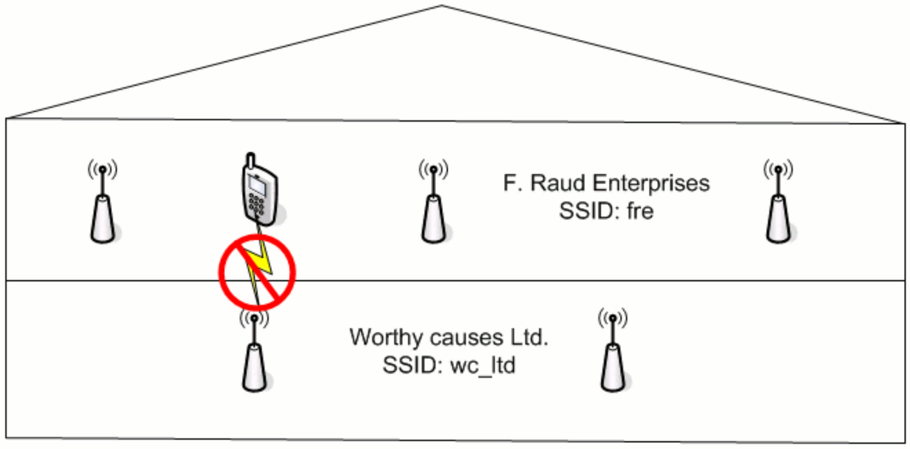


Figure 1-7: The importance of SSID check

If not all the access points are the same, it is also important to regard the features of the access point. Perhaps only some of the access points have QoS support which would be needed for VoIP, in that case an access point with QoS support will have to be preferred over an access point without. Access points with the same security setup as the current access point will also have to be preferred because it significantly helps the handover procedure.

1.4.3 Execution Phase

When the terminal has decided to leave the current AP and has chosen a new suitable one, the execution phase is initiated. The main concern here is how fast it can be done. The duration of the execution phase reflects the amount of packets that will be lost. In a VoIP session packet loss results in voice clipping and skips [8]. Depending on the voice compression method (codec) used packet loss up to 40 ms can be concealed [8]. This means that the duration of the execution phase should not exceed those 40 ms.

In enterprise networks the IEEE security and QoS enhancements 802.11i and 802.11e or the Wi-Fi alliance's corresponding standards are often used. These enhancements are an obstacle for performing fast handovers due to their configuring and negotiating routines. This will have to be worked around to achieve the 50 ms requirement.

1.4.3.1 Redirection of Streams

Redirection of data streams is another problem that occurs when changing access points. If a user is watching a video stream from the Internet while doing a handover, the video stream has to be redirected from the original AP to the new AP. In figure 1-8 the terminal has recently performed a handover from AP 1 to AP 2, but the video stream is still going to AP 1. How can this video stream be redirected to AP 2? And how fast will the Ethernet switch discover that the terminal now is connected to another access point and another physical port in the switch?

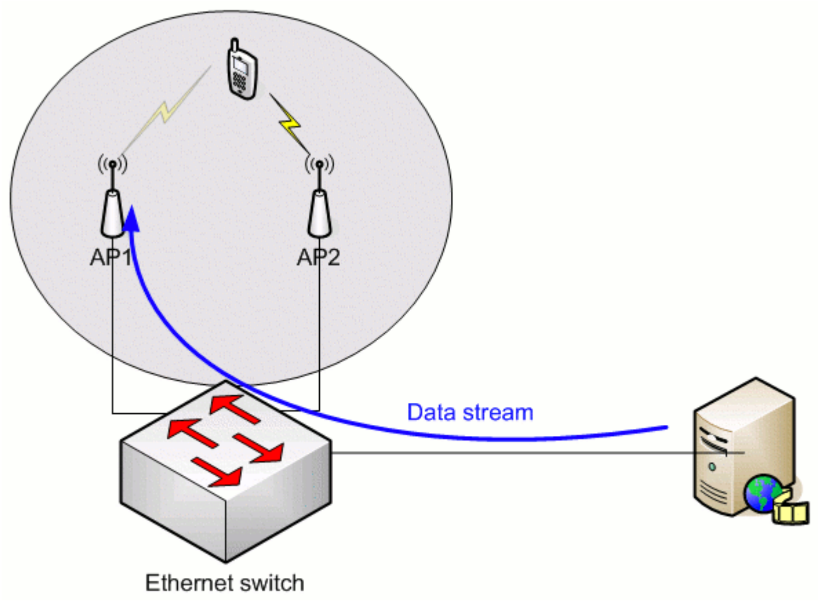


Figure 1-8: Redirection of streams

All three phases of a handover have special difficulties that need to be overcome. How other people have solved these difficulties will be described in the next main chapter "Theory and State of the Art".

1.5 Report Outline

Chapter 1 gives an introduction to the problem and why it is important to solve.

Chapter 2 presents theory and state of the art for the area concerned.

Chapter 3 describes a handover concept and our proposed handover algorithm.

Chapter 4 gives an insight to how the algorithm was implemented and the related tools.

Chapter 5 presents the results of the experiments and evaluates them.

Chapter 6 discusses both the theoretical and practical findings.

Chapter 7 concludes the results and suggests future work on the area.

2 Theory and State of the Art

To start off this chapter we want to present Ericsson mobile platforms' layered architecture. In figure 2-1 you can see the system design for WLAN on the EMP platform [9].

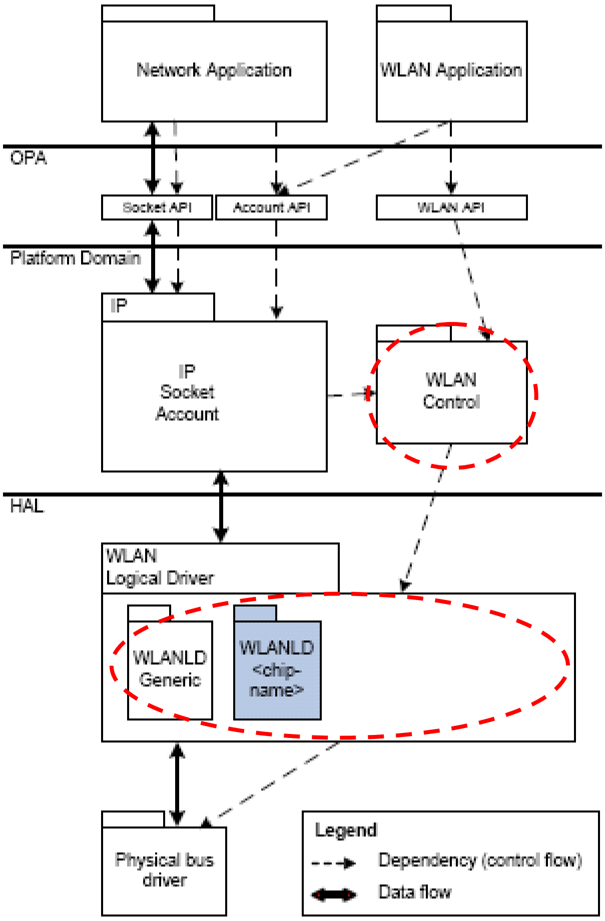


Figure 2-1: Ericsson Mobile Platforms system design [9]

The reason for introducing this figure is to establish an understanding that some parts of a fast handover implementation have to be implemented in a lower layer (WLAN Logical Driver) while the other parts can be implemented a higher layer (WLAN Control). See figure 2-2.

As presented in the problem statement, a handover process has three main phases: Detection, Selection and Execution. To try to present the handover process better, we divided it into five modules from a "modelling point of view". Before we start looking at the state of the art, the roles of these five modules will be described.

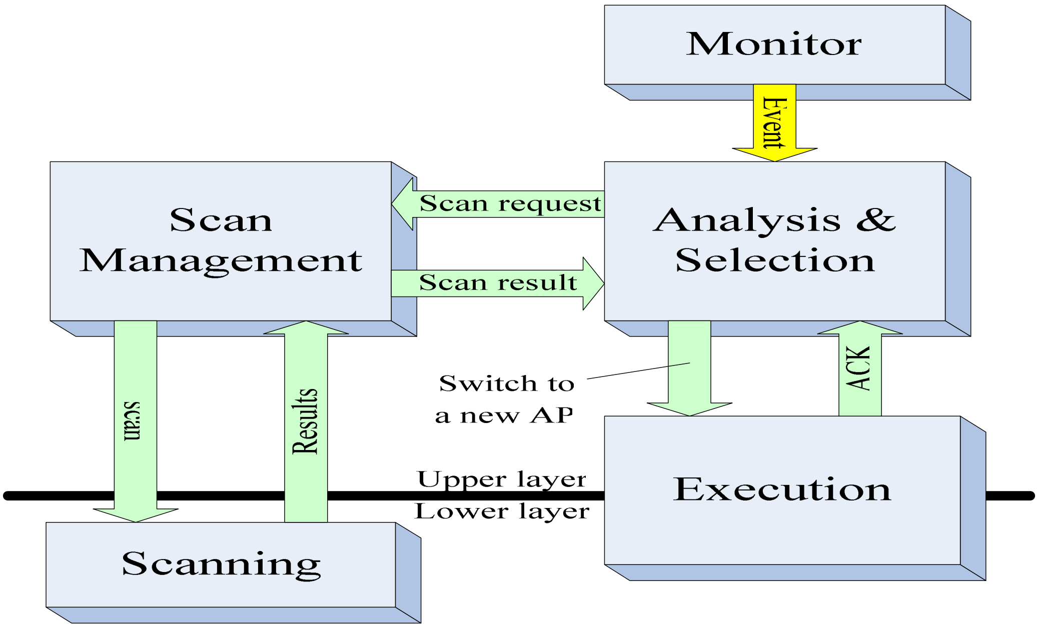


Figure 2-2: Handover process in modules

Firstly, the Scan Management module can control the Scanning module, which is a lower layer module, and can search surrounding APs and measure their physical features.

Then, the Analysis & Selection module is supposed to solve the problem "when and where to handover?". It has three main functions: ask for the current connection condition, trigger a handover when there is a problem, analyze the neighbour APs' performance and select a new AP to hand over.

Normally, the Analysis & Selection module sends a message asking the Scan Management module to trigger the Scanning module at intervals. And the Scan Management module sends scan missions to Scanning module, for example, every 500 ms. Then the Scanning module begins to measure current AP's, and sometimes the neighbour APs', signal strength, data rate, packets loss and other quality parameters. Whether the Scanning and Measure module scans and measure the neighbour APs is dependent on the type of detection methods used. The scanning results will be sent to the Scan Management module, and then be sent to Analysis & Selection module by the Scan Management module. This is because both the Analysis & Selection module and the Scan Management module are high layer modules and the interface between them is a high layer interface. In contrast, the Scanning module is a low layer module, thus the interface between it and the Scan Management module links the two layers' modules together.

The Analysis & Selection module gets this data and checks whether the current AP is good enough. It compares and measures data against the criterion it defined. When the current AP is not good enough, the Analysis & Selection module will ask the Scan Management module to trigger a scanning immediately. This time, the Scanning module scans the APs in the vicinity, and measures their performances, and then sends the data to the Analysis & Selection module.

With the data from the Scanning module, the Analysis & Selection module will analyze the data and check whether there is a better AP than the current one. If not, it will keep connection with the current AP. Otherwise, it will list all the better APs according to its criteria and choose the best one, as well as inform the Execution module to do a transition to this new AP.

Lastly, the Execution module uses the best method in our opinion to finish the fast handover action within 50ms as required. Finally, after the handover is completed, it sends an acknowledge message to the Analysis & Selection module. That is the general handover process we preconceived in our project.

In addition, the Monitor module is a background process which can also inform the Detection module of the connection quality by keeping track of packet loss and retransmissions. Compared with the Scanning and Measure module, the Monitor module is more active. It reports the connection conditions to the Detection module spontaneously without any request from the receiver. The information from the Monitor module will also be evaluated by the Detection module to see whether it is necessary to do a handover. Besides that, the Monitor module can also check the client's position and story history.

2.1 Scanning

2.1.1 Preemptive AP discovery vs. Handover-time AP discovery

According to [10], the client can figure out which AP to hand over to by scanning the medium for APs either before the decision to transit, which is a process called "Preemptive AP Discovery", or after the decision to transit, which is a process called "Handover-time AP Discovery".

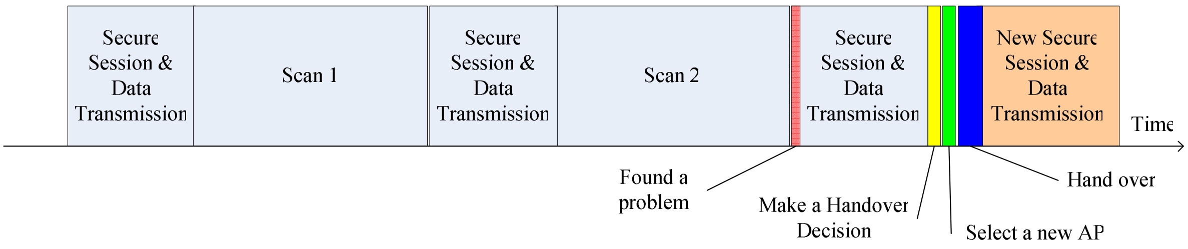

Secure Session & Data Transmission
Scan 1
Secure Session & Data Transmission
Scan 2
Secure Session & Data Transmission
New Secure Session & Data Transmission
Time
Found a problem
Make a Handover Decision
Select a new AP
Hand over

Figure 2-3: Preemptive AP Discovery

The Preemptive AP Discovery process can reduce the transition latency since the client knows which AP to hand over to. However, its drawbacks can not be ignored.

"The client cannot receive data from the currently associated AP while it is channel scanning." [10] When the client scans for new APs, it will leave its current channel to other channels. At the moment, if the AP sends data to the client, the client will miss the frames as it is on a different channel. And it has to require retransmission by the AP. Moreover, the client can not transmit data either while channel scanning. Thus, "The client application might experience throughput degradation."[10], which is not good for a fast-moving client.

Preemptive AP Discovery consumes much power for unnecessary scanning. For example, in figure 2-3, Scan 2 is useful while Scan 1 is meaningless because the current AP is good during that time and there is no need to do a handover.

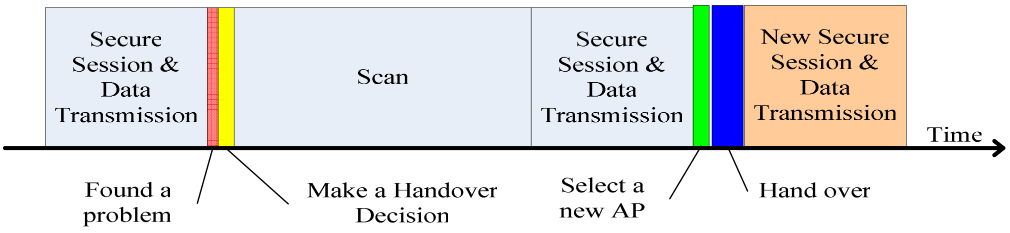

Secure Session & Data Transmission
Scan
Secure Session & Data Transmission
New Secure Session & Data Transmission
Time
Found a problem
Make a Handover Decision
Select a new AP
Hand over

Figure 2-4: Handover-time AP discovery

Handover-time AP discovery only scans after the handover decision has been made. It saves power but takes more time than preemptive AP discovery handover.

2.1.2 Active scanning VS Passive scanning

Active scanning means the client actively searches for an AP while passive scanning means the client only listens for beacon frames on each channel passively.

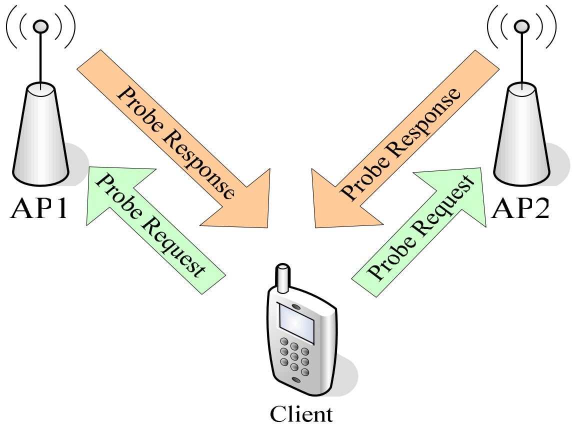


Figure 2-5: Active Scanning

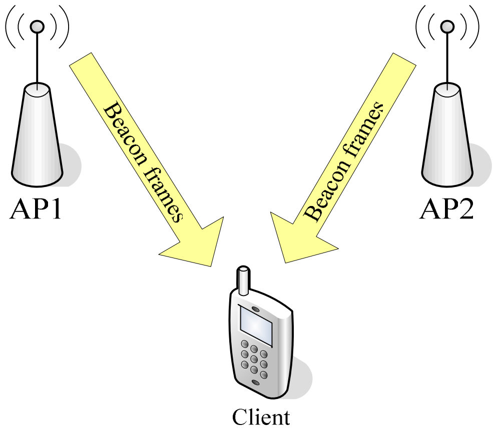


Figure 2-6: Passive Scanning

With active scanning, the client sends out 802.11 probe requests across all channels and waits for probe responses from APs. On each channel, if the client doesn't receive any probe response within a *MinChannelTime*, then it switches to the next channel. If the client received at least one response on the specific channel, then it has to wait for more responses on this channel for a *MaxChannelTime* [11]. Active scanning enables the client to get all APs' information but requires the client transmit probes actively. The total latency of an active scanning process is roughly between 50 and 360 milliseconds [12], depending on the number of active channels and APs. The active scanning delay can be reduced by refining the *MinChannelTime* and the *MaxChannelTime*. Mostly, 802.11 VoIP phones and PC client cards rely on active scanning [10].

With passive scanning, the client must stay on each channel for a longer time than active scanning because he or she has to receive beacons from as many APs as possible for a given channel. "Passive scanning has the benefit of not requiring the client to transmit probe requests but runs the risk of potentially missing an AP because it might not receive a beacon during the scanning duration."[10] The latency of a passive scanning process is about 1.1 seconds for the 11 channels of the IEEE 802.11b/g band [11]. The client has to stay on each channel for 100 ms in order to receive all periodic beacons frames of all APs. So the most direct way to reduce the passive scanning delay is to scan the selected channels rather than all channels.

2.1.3 SyncScan

SyncScan is a kind of passive scanning method. It predicts the beacons' coming time on each channel and ensures clients can passively scan by switching channels exactly when a beacon is about to arrive. Thus, SyncScan latency is equal to the double channel switching time plus the time that the client lingers waiting for beacons.

$$SynScanDelay = 2 * SwitchTime + WaitTime \qquad [13]$$

"SyncScan, replaces the large transient overhead of active access point discovery with a continuous process that passively monitors other channels for the presence of nearby access points. The potential disruption of channel switching is minimized by synchronizing short listening periods at the client with regular periodic transmissions from each access point."[13]

The most obvious benefit of SyncScan is that it can reduce the handover delay rapidly. It allows handover to be made earlier and with more confidence, thereby improving the quality of a client's connectivity and reducing unnecessary interference. And it also provides an opportunity for continuous location tracking.

However, SyncScan does add other complications. In order to be synchronized, the clock in the APs should be the same as global synchronization of beacon timings. Beacons from different APs on the same channel might come at the same time and then interference might occur. Furthermore, when the client is listening to other channels, it cannot send or receive packets from the current AP. Thus, packets might be dropped.

2.1.4 802.11k

As a proposed standard for radio resource measurement in an 802.11 wireless LAN, 802.11k exposes radio and network information to help management and maintenance of a mobile wireless LAN [14]. It aims to provide key client feedback to WLAN access points and switches. The proposed standard defines a series of measurement requests and reports that detail Layer 1 and Layer 2 client statistics. [15]

Within the IEEE 802.11k task force specification, APs will not only advertise their presence to potential wireless clients using the beacon frames, but also provide a site report which contains an ordered list of access points, from best to worst service. "An access point asks a client to go to a specific channel and report all the access point beacons it hears. The access point collects the data, and it or a WLAN switch will analyze the beacon information, looking at details such as what services and encryption types each access point supports and how strongly the client heard the access point. Then the switch or AP generates the site report." [15]

"Moreover, in certain cases the client is informed of the exact identities of APs in each channel, and can unicast Probe Request messages. Thus the client can collect information on all APs in that channel, and move to the next one without having to wait up to *MaxChannelTime*." [11]

802.11k can also be used by the Analysis and Selection phases for taking handover decisions and choosing a new AP.

2.1.5 Measurements and smoothing of data

An AP's service performance depends on the link quality between the AP and the client, as well as the load of the AP. However, most of currently implemented handover algorithms typically measure an AP's performance according to the Received Signal Strength Indicator (RSSI) on the client. In order to avoid unnecessary handover, those algorithms prefer the smoothed RSSI value to the raw RSSI value.

To smooth the RSSI value, there are two issues that should be solved: [11]

- RSSI time series are bound to have missing values due to AP's high load or queuing delays;
- The radio environment is highly time varying due to shadowing and fading.

To the first one, the time series for missing values should be pre-processed. In [11], it replaces the missing value in the time series by -80dB, a value that corresponds to no effective communication channel.

The second one can be solved by data smoothing, which removes the high frequency component (short wiggles) and emphasises the low frequency ones (longer trends) in the signal. Normally, there are two popular data smoothing forms: Moving Average and Exponential Weighted Moving Average (EWMA).

"The Moving Average Filter regards each data point in the data window to be equally important when calculating the average (filtered) value." [16] In other words, at any instant, a moving window of *n* values is used to calculate the average of the data sequence.

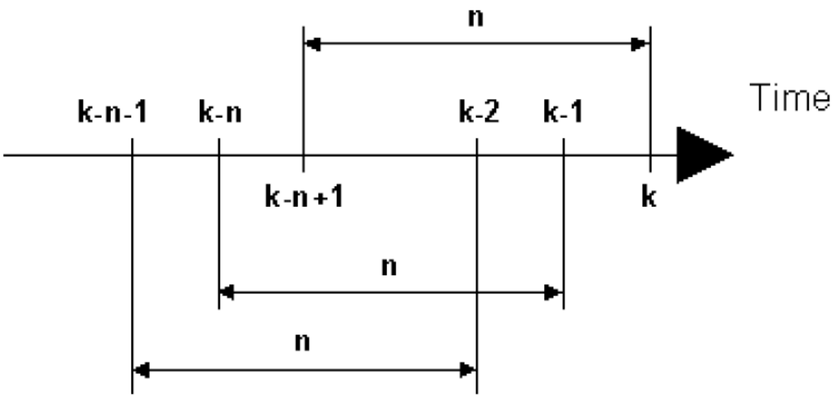


Figure 2-7: Moving windows of N data points [16]

The Moving Average Filtering is expressed as:

$$y(k) = y(k-1) + \frac{1}{n} * [x(k) - x(k-n)] \qquad [17]$$

where $y(k)$ is the value of the filter at time k, $x(k)$ is the measurement collected at time k. From the recursive expression we can find that calculating the current filtered value requires the use of the measurement n time-steps in the past. All data points are equally emphasized, so if signal has a trend, the Moving Average Filtering is not desirable.

In contrast to Moving Average Filtering, the Exponential Weighted Moving Average Filter (EWMAF) places more importance on more recent data by discounting older data in an exponential manner. Its expression is also recursive:

$$y(k) = \alpha * y(k-1) + (1-\alpha) * x(k) \qquad [17]$$

The filter constant α can control the impact that the current measurement has on the value of the filter. α should be more than 0 and less than 1. When a large number of points are being considered, $\alpha \to 1$, then $y(k) \to y(k-1)$. It means that the filter does much filtering but emphasises little on current value. On the other extreme, if $\alpha \to 0$, then $y(k) \to x(k)$, which means that the filter does little filtering but emphasises a lot on current value.

According to the expression of the EWMAF, we can see it emphasises recent events, smoothes out high frequency (short period) events, and reveals long term trends.

Consequently, both the Moving Average Filter and the EWMA Filter are implemented by means of efficient recursive formulae, and are very useful techniques for emphasising apparent slow trends in sequences of data. The moving Average Filter emphasises all measurements while EWMA Filter focuses more on recent data.

2.1.6 Power Save Mode

In the 802.11 base standard [1] power save mode is described. Power save mode is a feature that allows an embedded station to go to sleep for a short while to save battery power. Power saving is achieved by setting a bit to 1 in the MAC frame header. This bit is named "power management" and it is a subfield of the "frame control" field in the MAC frame.
When setting this bit, the access point will buffer up all packets destined for the station and when the station returns to active mode the access point will send the buffered packets. This feature could be very useful during scanning to reduce packet loss. We could tell the currently associated access point that we are going in power save mode, but instead perform scanning for 10-20 ms and then return to active mode and retrieve the buffered packets.

2.2 Analysis and Selection

2.2.1 Core issues of a handover algorithm

There are two key points that should be considered in handover decision making:

1. Who takes the handover decision?
2. What is the criterion/level for the handover decision?

The answer to the first question is not obvious since the decision to affect the handover can be made by the mobile station, the network, or in a combination of the two.

In a client based solution, the hand-held device itself has to search for and discover all access points in range by detecting the lack of radio connectivity, and then measure different parameters. These parameters can be signal strength, time and/or bit-error rate. To make it robust, averages of the input parameters must be computed to ensure a good decision. However, "the main difficulty is to determine the reason for the failure among collision, radio signal fading or the station being out of range."[18]

In a server based solution the server will collect information about signal strength and bit-error rate from all the access points in the network. Based on this information the server can decide when a handover is appropriate. The server can prepare the network for a handover and tell the hand-held device to perform the handover. The problem with a network based solution lies in compatibility issues. A standardized interface between the server and the client would be needed. Mostly, a hybrid solution is easier and more prevalent.

In a handover algorithm, normally the decision on when to affect the handover must be based on measurements of the connections made at the handheld devices or at the access points, or both. And as we mentioned in the last section, we have to define a criterion to make a handover decision. Handover algorithms are distinguished from one another by the handover criteria and the way by which handover criteria are incorporated in the algorithms. Since handovers are expensive to execute, while ensure connection quality, unnecessary handovers should be avoided. As described in [19], unnecessary handovers may pose the following problems of i) increase in the network load as each handover requires network resources to reroute the call to a new AP, and ii) shortage in channel resources, leading to call dropping.Thus, proper handover criteria are significant for a handover algorithm.

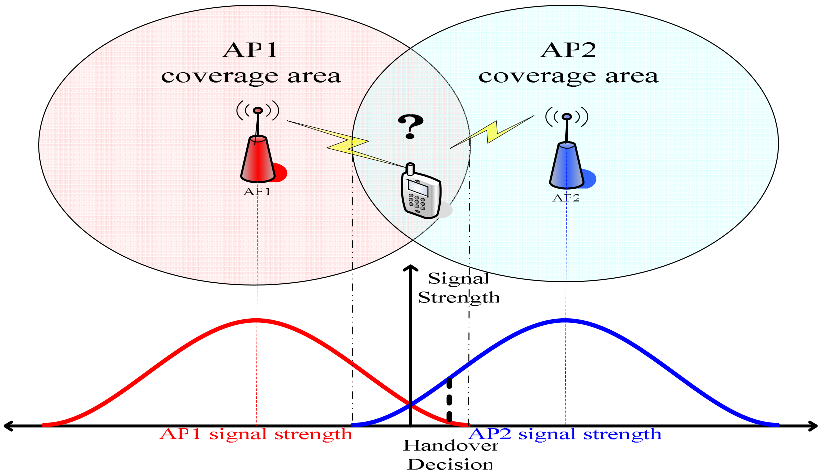


Figure 2-8: Signal strength between two access points

A good algorithm can minimize client service disruption while a bad one can lead to a disconnection. If the handover criteria are not chosen appropriately, then the yoyo effect which we don’t want might happen. The call might be handed back and forth several times between two access points when a portable in the overlapping region between their coverage area boundaries, as the figure 2-8 presented. The yoyo effect will aggravate the network traffic load, waste battery power and network resources, and even break the connection between the station and the AP. “If the criteria are too conservative, the call might be lost before the handover can take place.”[20] Unreliable and inefficient handover procedures will reduce the quality and reliability of the network.

Different handover algorithms have different handover criteria, and different criteria focus on different issues. Various factors, such as QoS, packet loss rate, data rate, S/N, and power consumption, are very important to the handover criteria. In the following sections, we will introduce a taxonomy of handover algorithms and give some typical examples of each.

2.2.2 A taxonomy of handover algorithms

As described in [11], there are 4 classification principles. The first one cares “whether they trigger transitions across the entire range of the analyzed performance metric or only when the performance falls below for instance excellent levels”. The next one considers “whether they incorporate history or not." Then the third one refers "whether they use neighbourhood knowledge or simply rely on current AP performance". Lastly, the fourth one is concerned about "whether they use simple value criteria, or use trends in the overall signal received by an AP."

1. React to bad performance VS. Pursue better performance
Most of today's handover algorithms are designed only to react to disconnection and unacceptable performance, while some other mechanisms pursue better client service, for instance, improve the clients' throughput. If there is a new AP with better performance than the currently one's, such mechanisms will initiate a handover even when the current AP is at its best performance. "Transitions across APs that both live in the 'excellent' region will have no impact on user throughput."[11]

2. Incorporate history VS. Not incorporate history
Those non-history incorporated handover schemes make their transition decision dependent only on raw instantaneous values of the performance.

As for the handover algorithms which incorporate history, they have two different categories. The first kind of algorithms capture and record the APs' long term performance, and make a decision on whether to trigger a handover according to an analyses of the records. The second class of algorithms also use their criteria to measure instant performance data. However, the measurements they use are smoothed. Since the radio environment is highly time varying due to shadowing and fading, the signal strength changes rapidly every millisecond. The smoothing operation can filter out rapid signal fluctuation and extract the overall trend in the APs' signals.

3. Care current AP's condition only VS. Consider neighbour APs' conditions
This kind of taxonomy is similar to the first one. In general, the handover algorithms which react to link disruption and poor performance only focus on the current AP and do not care for the performance of neighbouring APs. In contrast, the handover schemes which want to offer clients the best AP performance are always scanning and measuring surrounding APs. Transitions will be done continually until the station gets the best AP in the network.

4. Value criteria VS. Trend criteria
If two handover algorithms are based on the measurement of signal strength, then one can specifies a fixed strength value as its criterion while the other one can use a strength trend.

2.2.3 Typical handover algorithms

The state of art in the area of nowadays typical handover algorithms can be summarized in those schemes:

1. **Beacon/Frames based solution**
   "Beacon scheme uses the number of consecutively lost beacons (or unacknowledged MAC layer frames) to issue a trigger and is implemented in the Intel 2915ABG cards (both Linux and Windows drivers)."[11] This scheme incorporates historical information and ignores knowledge on the performance of neighbours either. It only operates a transition when the current AP has a potential disconnection. Thus, there is no chance for the beacon scheme to do a handover when the current AP is perfect.

2. Signal Strength based solution

- Best signal scheme

  The scheme allows users to choose an AP with the strongest signal strength to handover at all times. Normally, it uses the smoothed measurement of the received signal strength instead the raw one. The cost for offering the clients the best service quality is large power consumption and a lot of unnecessary handover.

- Threshold scheme

  In this kind of handover algorithm, the criterion is a specific connection quality level which can be the signal strength value or sometimes the lost frames number. The quality level specifies the worst connection condition between a station and an AP that the users can tolerate. It won't trigger a transition until the connection quality degrades substantially, worse than the level it defined.

  The threshold scheme disregards the APs nearby, only paying attention to the current AP's performance. It typically evokes a transition only when the stations have bad connection with the current AP. In this way it can save much battery power since there is no unnecessary handover when the client is experiencing good service. The scheme doesn't incorporate history if it takes the raw instantaneous value. In this way, although the scheme is easy to be implemented, it will cause an excessive number of unnecessary transitions. Therefore, a better way is using the smoothed connection quality value.

- Hysteresis (Δ) algorithm

  In this algorithm, the Scanning module scans all the APs operating on the current and overlapping channels and measures their signal strength. According to the data from the Scanning and Measure module, the Analyse and Select module will propose a handover when there is an AP whose RSSI (Received Signal Strength Indicator) value exceeds the current AP's RSSI value plus a hysteresis factor of Δ. "This scheme is in line with handover schemes implemented in cellular networks". [20] Similar to the "Best Quality Scheme", it will lead to many unnecessary transitions. But with the hysteresis, the yoyo effect can to some extent be prevented.

- Trend (L) algorithm

  To reduce excessive transitions between APs, this algorithm prefer a long-term performance to instantaneous behaviour. It defines the amount of time L and calculates the rate of change of signal strength by this formula in [11].

$$\alpha = \frac{y(t) - y(t - L + 1)}{L}$$

  In this formula, y(t) and y(t-L+1) are the smoothed RSSI value of an AP at the timeslot t and t-L+1.

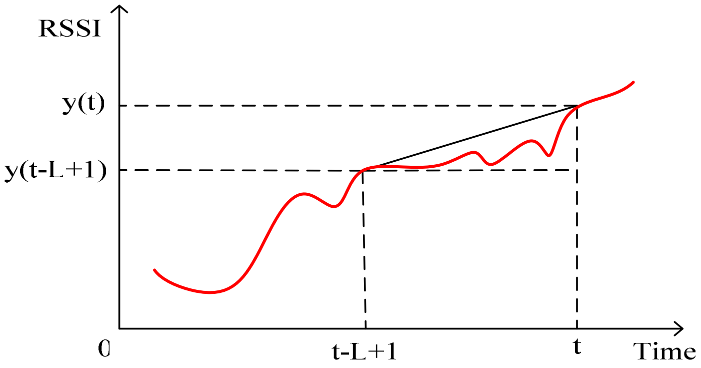


Figure 2-9: Rate of change of AP signal strength [11]

A positive α value means the AP has an increasing trend while a negative α value implies a decreasing AP performance. In this algorithm, the client scans, measures the current AP and its neighbours. Once a new AP has a positive rate of signal change and the current AP has a negative one, a handover will occur. The problem is, according to this algorithm, when a user goes from the current AP to a new AP, the user will do a handover to the new AP as soon as he or she enters the overlapping region between the two APs' coverage area boundaries, since the current AP has a downward trend while the new one has an upward trend. However, as figure 2-8 shows, we can easily find that the current AP's performance is better than the new one when a handover happens.

- Prediction algorithm

  The prediction algorithm bases the handover decision on the expected future value of the received signal strength. The LSE (Least Square Estimator) algorithm is a kind of prediction schemes which predict the value of the signal in the next time interval. It is supposed to use the smoothed measurements of signal strength. In [11], the formula to calculate the prediction value is: $y(t) = \alpha * x(t) + \beta$. "And at this point one potential triggering mechanism is to allow a transition to a new AP (z) only if the least squares estimator for the new AP minus its associated error $\delta z(t)$ still exceeds the least squares predictor of the current AP (y) plus the associated error $\delta y(t)$, i.e., $z(t) - \delta z(t) > y(t) + \delta y(t)$."[10] It means that a handover is initiated only when the lowest predicted value for the new AP is higher than the highest predicted value for the current AP. "**A** technique is proposed and shown via simulation to be better, in terms of a reduction in the number of unnecessary handovers, than Best Quality scheme, Hysteresis algorithm and Trend algorithm" [21].

3. Throughput based solution

- Load balancing algorithm

  In some hot-spots, users greedily associate APs with the best received signal strength, which leads to unbalanced network traffic load and unfair bandwidth allocation among users. To address this issue, the load balancing algorithm assists in changing user-AP associations on the fly and trades off signal strength with load by forcing a mobile user to change association from an overloaded AP with a stronger signal to a neighboring lightly loaded AP with a possibly weaker signal. [22]

In each AP, there is a Load Balancing Agent (LBA) which periodically broadcasts its AP's load condition to the common backbone, and determines whether the load is overloaded, balanced or under-loaded, according to the reports from other LBAs. The overloaded APs will force the current stations to hand over to the under-loaded APs until the traffic load is balanced among all the APs in the network. [23]

The stations which only pay attention to the received signal strength will never trigger a handover spontaneously even when they experience poor performance from a strong signal strength but seriously overloaded AP. Thus, the overloaded AP has to disconnect the station at first, and then only under-loaded APs can accept it.

The load balancing algorithm can increase the total network throughput and improve overall network utilization.

- 802.11k

  As we described in section 1.1.4, the 802.11k draft enable the AP to make a site report which lists all the available APs from the best to the worst. The site report is an analysis result of the 802.11k radio resource measurements which include beacon measurements, measurement pilot frames, frame measurements, channel load measurements, noise histogram measurements, station statistics measurements, location configuration information measurements, neighbour report measurements, link measurements and QoS metrics measurements. Then, the clients will choose the best AP in the site report. "By accessing and using this information, the stations can make decisions about the most effective way to utilize the available spectrum, power, and bandwidth for its desired communications." [24]

Consequently, most of today's implemented AP selection mechanisms typically select APs according to the signal strength. Those mechanisms are direct but will lead to unevenly distributed traffic load. While the current throughput based solutions typically require changes in the infrastructure [11]. However, the 802.11k standard draft offers us a brand-new, effective way to make handover decisions based on the general connection service quality.

2.3 Execution

2.3.1 Layer 2 vs. Layer 3 Handover

There are generally two types of handovers, the layer 2 handover and the layer 3 handover (roaming). The layer 3 handover require changing the layer 3 network address (IP-address) i.e. changing subnet, while the layer 2 handover does not involve a layer 3 router, only the layer 2 switches (see figure 2-10 and 2-11) [10].

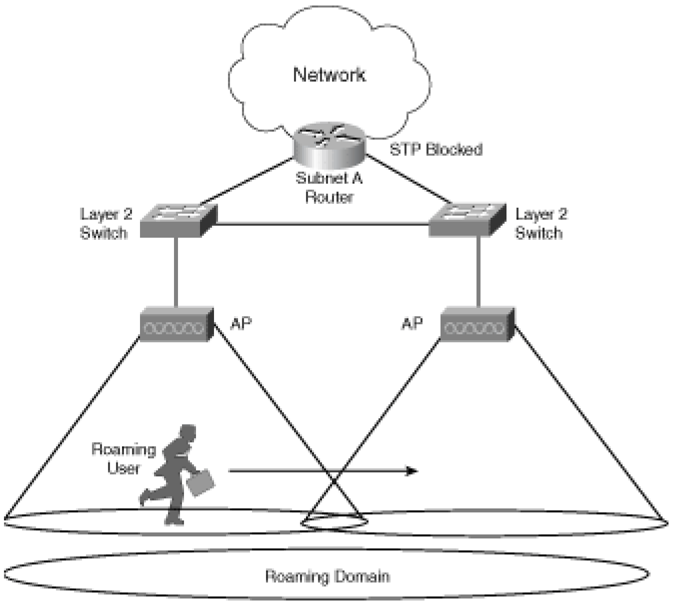


Figure 2-10: Layer 2 handover [10]

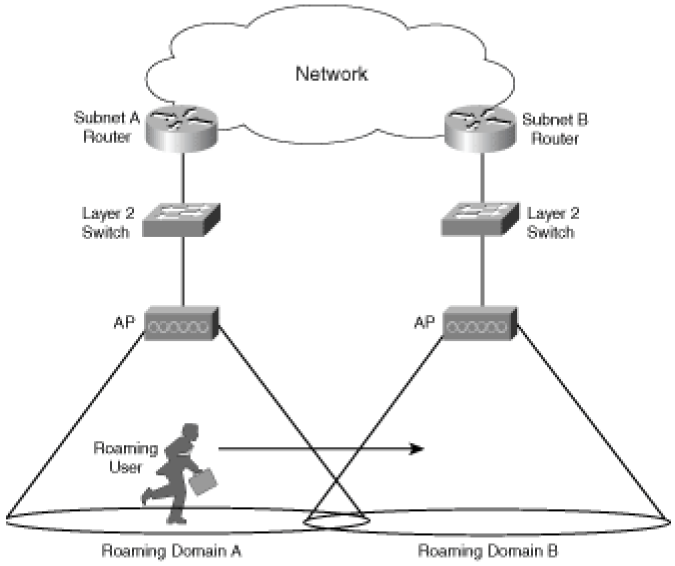


Figure 2-11: Layer 3 handover (roaming) [10]

A layer 3 handover is therefore more complex than a layer 2 handover because it involves more nodes. In this thesis we will focus on layer 2 handovers.

2.3.2 Latencies

In the problem statement it is mentioned that the main concern of the execution phase is how long it will take. This depends on the duration of the following processes [10]:

- Switch radio channel and synchronize
- 802.11 re-authentication process
- 802.11 re-association process
- 802.1X / Extensible Authentication Protocol (EAP) (re-)authentication
- 802.11e QoS re-negotiation

Of these processes it is the 802.1X / EAP authentication that takes the longest time and can save the most time.

2.3.3 Proprietary Solutions

There are solutions for fast secure handover, but these are proprietary and do not allow inter-vendor compatibility. Cisco [25] has actually incorporated a solution based on the 802.11i key caching for handovers within an enterprise network where one AP on each subnet holds the role as authenticator [26]. All the clients authenticates through the authenticator AP which then distributes the shared keys to all the other entities in the layer 2 domain [26]. Symbol technologies [27] also provide similar functionality for fast secure handover [26]. Bluesocket [28] use gateways that control simple APs to manage the handover process [26]. Proxim wireless [29] uses a slightly different technique using the current connected AP to prepare for the handover and for pre-authenticating with neighbour APs [26].

Even though these solutions may work nicely it is better to use open standards to ensure inter-vendor compatibility. In the next sections we will look at different standards that try to solve our problem.

2.3.4 802.11f Inter-Access Point Protocol

Inter-Access Point Protocol (IAPP) or 802.11F is an IEEE recommendation standard that allows communication between access points. It is designed to reduce the authentication delay during a handover by copying the security context of a mobile station from the old AP to the new AP [30] making the re-authentication delay at the new AP smaller (see figure 2-12). Most enterprises use the IEEE 802.1X [31] standard for authentication, in these cases the security context and MAC-address/IP-address mapping is distributed by a RADIUS / AAA server [30].

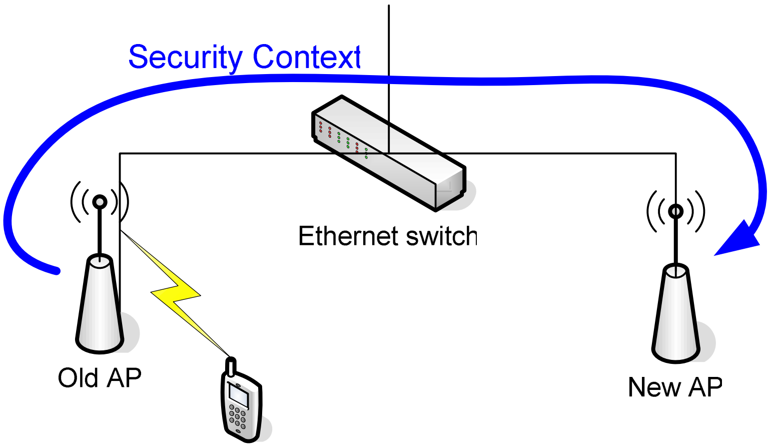


Figure 2-12: IAPP communication

In order to reduce handover delay caused by IAPP communication between the access points and RADIUS server, proactive caching was developed. With proactive caching the security context of the mobile station is distributed to all neighbour access points before a handover decision is made [32]. When the handover takes place the neighbour access points are ready to welcome the new mobile station and the handover delay can be drastically reduced [32].

802.11f specifies how the security context should be moved but does not specify the actual content structure. This lead to different implementations by different vendors, and the whole point of the amendment was lost. IEEE approved its withdrawal in the February 03, 2006 [30].

2.3.5 802.11i MAC Security Enhancements

The 802.11i amendment (also known as WPA2) was standardised in 2004 and was designed for better security in wireless LANs using the 802.1X / EAP authentication method [33]. This amendment is actually one of the biggest problems due to handover delay because this is a complex authentication process that takes a lot of time. Regular data traffic is not allowed until the authentication process is finished [33]. However, 802.11i also provides a solution to this problem.

2.3.5.1 Pairwise Master Key (PMK) caching

With 802.1X / EAP authentication a PMK that both the client and the AP know is created [33]. The PMK is used to further create encryption keys. PMK caching involves storing the PMK so that the next time the client visits that AP the PMK already is available and does not need to be created again which reduces the authentication delay [33]. The limit of PMK caching is that full 802.1X / EAP authentication is required with first association with each AP [33], which means the fast re-authentication only can be used when the AP has been visited before.

If a WLAN controller is present in the network Proactive Key Caching (PKC) can be used to distribute the PMK to all the other APs in the network [33]. This method is used in the proprietary solutions described in chapter 2.3.3.

2.3.5.2 Pre-Authentication

For those WLANs without a controller, pre-authentication can be used instead of proactive key caching. Pre-authentication is an optional part of 802.11i and might not be implemented in APs and clients [33].

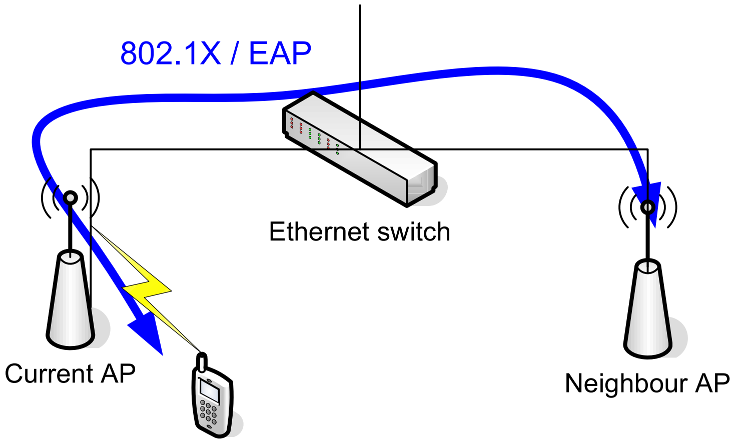


Figure 2-13: Pre-authentication through current AP

Pre-authentication is used in addition to PMK caching allowing the client to perform a full 802.1X / EAP authentication to a new AP while still connected and communicating with the currently connected AP [33]. This traffic goes over the wired network as illustrated in figure 2-13. The newly received PMK is then cached like in regular PMK caching.

The drawbacks of pre-authentication are that it adds load on the AAA server because a new

PMK is made for each new AP, it does not solve layer 3 handovers nor cope with the 802.11e QoS re-negotiation delay.

2.3.6 802.11r Fast BSS Transitions

In the original 802.11 standard family, the 802.11i and the 802.11e provided strong security and Quality of Service (QoS) features to the voice applications, but these complex mechanisms include re-authentication, and re-authorization, QoS renegotiations require longer time during handover, from hundreds of milliseconds to even seconds to complete depending upon the authentication server load and traffic conditions.

The Fast BSS Transition standard, IEEE 802.11r, was introduced by the new IEEE Task Group to minimize BSS transition time while still providing the services offered by IEEE 802.11i and IEEE 802.11e.

"The BSS transition process where a station (STA) roams from one AP to another could consist of up to 6 stages: discovery (Probe exchange), 802.11 open authentication, Re-association, Authentication method, Extensible Authentication Protocol Over LANs (EAPOL) key exchange, and QoS renegotiation." [34]

"The Fast BSS Transition mechanism allows a STA to establish security and/or QoS state at the target AP prior to or during (re)association, avoiding delays in connecting to the DS after transition."[35] According to [34] there are two ways to reduce the handover time: firstly, the Pairwise Temporal Keys (PMK) which is used to mutually authenticate between AP and station, will be generated once the station joins the network, and be distributed to all the APs that are authenticated in the subnet. It saves much time during the authentication phase while the station is roaming. Secondly, the 4-way handshake and the IEEE 802.11e traffic specification (TSPEC) negotiation are completed during re-association. Thus, the latency will be reduced strongly during the transition process. "The overall changes to the protocol will not introduce any new security vulnerabilities beyond the current IEEE 802.11 standard and its amendments. The Fast BSS Transition mechanism preserves the behaviour of legacy STAs and APs."[35]

2.3.6.1 Transition mechanisms

To simplify the authentication process, 802.11r specifies a new key management system which includes a new key hierarchy and corresponding key derivation algorithms.

According to [34], in the key management system, a Security Mobility Domain (SMD) consists of several Security Domains (SD), and a security domain consists of only one Level 0 key holder (R0KH), all Level 1 key holders (R1KH) which associated with R0KH, and all APs which associated with R1KH. In this system, the key holders such as R0KH, R1KH and PTK (Pairwise Temporal Key) key holders are logical network entities which can be APs or other physical devices. They are authorized to store keying material. The key in each key holder stores the station address, its key holder's address or network identity, and possibly some other information. The key holders in each level in the key system get their keys from their adjacent upper level key holders and devise the keys to their adjacent lower level key holders. "An R0KH within an SMD can derive PMK-R1 for any R1KH in the SMD, even those not in the same security domain."[34]

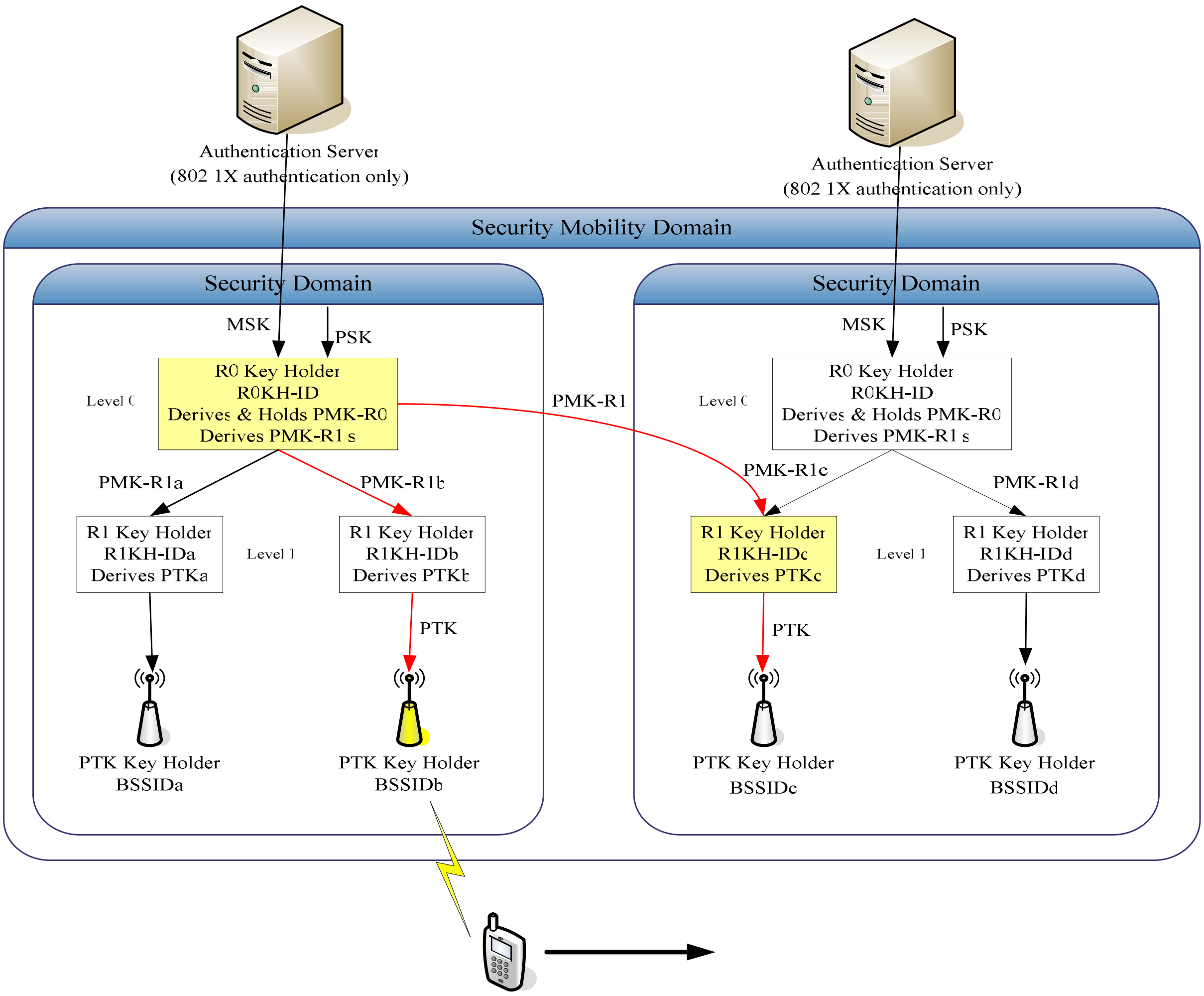


Figure 2-14: 802.11r key management system

In this way, once a station enters an SMD and associates with an AP the first time, the keys are derived step by step from the R0KH to the AP. As figure 2-14 shows, the PMK-R1 keys are derived using the current AP's R0-KHID and the R1-KHID of every AP in this SMD, and then are distributed to all APs correspondingly. When the user proposes to do a handover from the current AP (yellow one) to a new AP, the target AP's R1KH has already gotten PMK-R1 from the current AP's R0KH. Thus, IEEE 802.1x authentication is not required any more. 802.11r does not specify a new security scheme but assumes secure connections between the key holders.

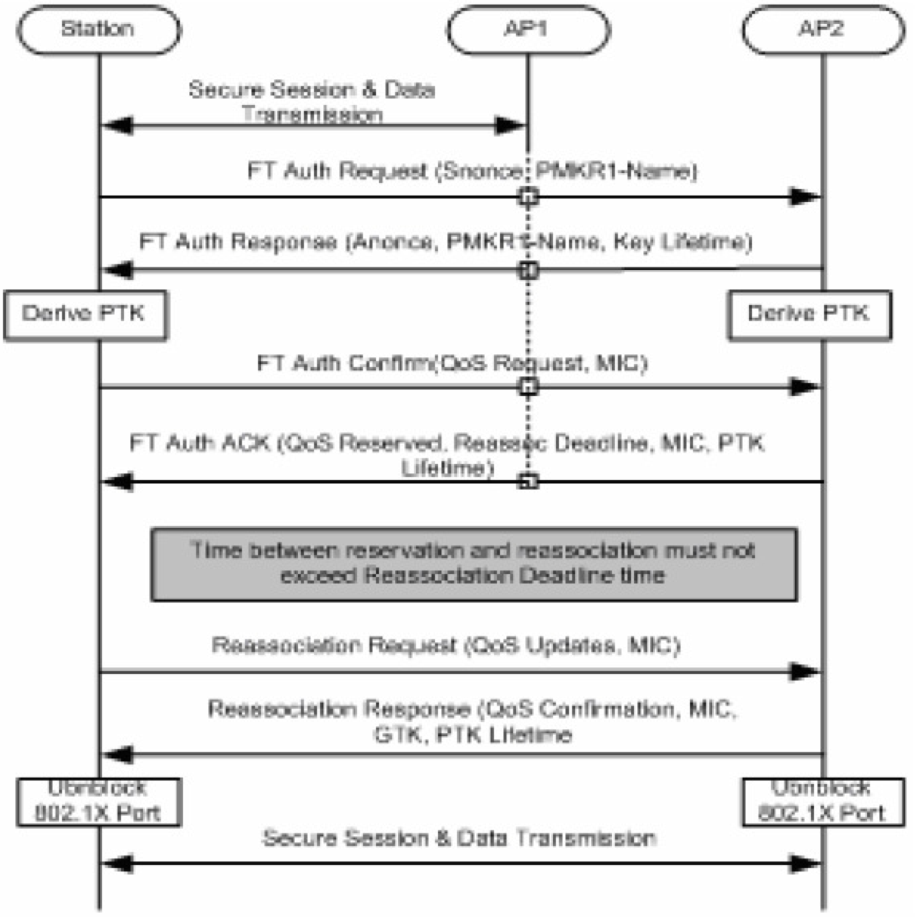

Station
AP1
AP2
Secure Session & Data Transmission
FT Auth Request (Snonce, PMKR1-Name)
FT Auth Response (Anonce, PMKR1-Name, Key Lifetime)
Derive PTK
Derive PTK
FT Auth Confirm(QoS Request, MIC)
FT Auth ACK (QoS Reserved, Reassoc Deadline, MIC, PTK Lifetime)
Time between reservation and reassociation must not exceed Reassociation Deadline time
Reassociation Request (QoS Updates, MIC)
Reassociation Response (QoS Confirmation, MIC, GTK, PTK Lifetime
Unblock 802.1X Port
Unblock 802.1X Port
Secure Session & Data Transmission

Figure 2-15: 802.11r Transition [34]

Moreover, “802.11r allows a station to request quality of service resources at the time of re-association, thus avoiding a separate message exchange to reserve the needed resources before data transfer can resume.”[34] It also allows a station reserve QoS resources prior to committing to re-association.

2.3.7 Two Radio Cards

As described in [36] two radio cards could be very convenient during a handover process. One card can be used for regular data traffic (data-card) while the other one is scanning for new APs (control card). When the handover decision is made, the control card can connect, authenticate and perform QoS negotiations before the data traffic is re-routed. Once the handover is complete, the former control card has become the data card and the former data card has become a control card (see figure 2-16).

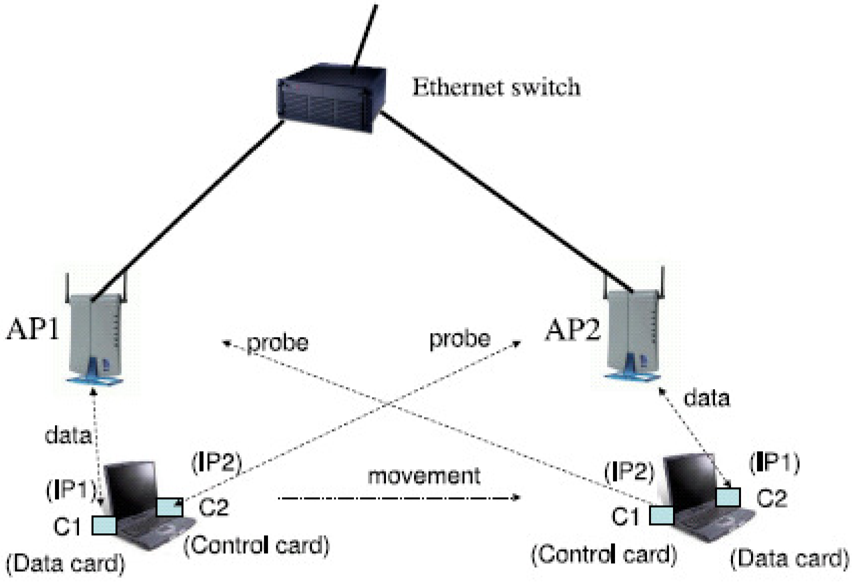


Figure 2-16: Two radio card handover [36]

The two card design has the advantages that no packets will be lost during scanning and the duration of the execution phase is very low (less than 10 ms). However, we are concerned about the power consumption of this solution as well as the cost. In hand-held devices, physical size of the components is also a big issue. Can a two card design use the same antenna and what about interference?

2.3.8 Quality of Service

Quality of service or QoS normally refers to control mechanisms for giving different types of data traffic different priority [37]. For instance it is important to give VoIP calls a higher priority than a file transfer. The two most used mechanisms for providing QoS in wireless networks are the IEEE 802.11e amendment and the WiFi Alliance's WiFi Multimedia (WMM) which is a subset of 802.11e.
Both these mechanisms classify data traffic into four classes (voice, video, background and best effort) and give them different priorities [38]. In wireless LAN all stations have to fight to get access to the channel. If the medium is busy, all stations wait a random backoff time period. The duration of this time period is a random value between zero and a value called CWmin. By giving the different traffic classes different CWmin values statistical prioritizing is achieved [39].

QoS is an obstacle for fast handovers because QoS negotiation takes place during re-association with an access point.

2.3.9 Updating Switches

When the handover is completed, the Ethernet switches have to be updated. An Ethernet switch has a MAC switching table which maps MAC-addresses to physical ports on the switch. When a station associates to a new AP it will be on another port on the switch and the data traffic destined for the station will still be sent through the old port. A solution to this would be to let the station send a broadcast packet once associated with the new AP. This packet will reach the switch and it will save the new port number for the stations MAC-address.

3 Concept

3.1 Overview

Figure 3-1 is a more detailed version of figure 2-2 in chapter 2. In this new version we have divided the execution module into two modules, one in the WLAN control layer and the other one in the WLAN logical driver layer. We have also added the High level support module that can talk with the WLAN applications through the WLAN API. In the next subsections we will give a high level description of the different modules.

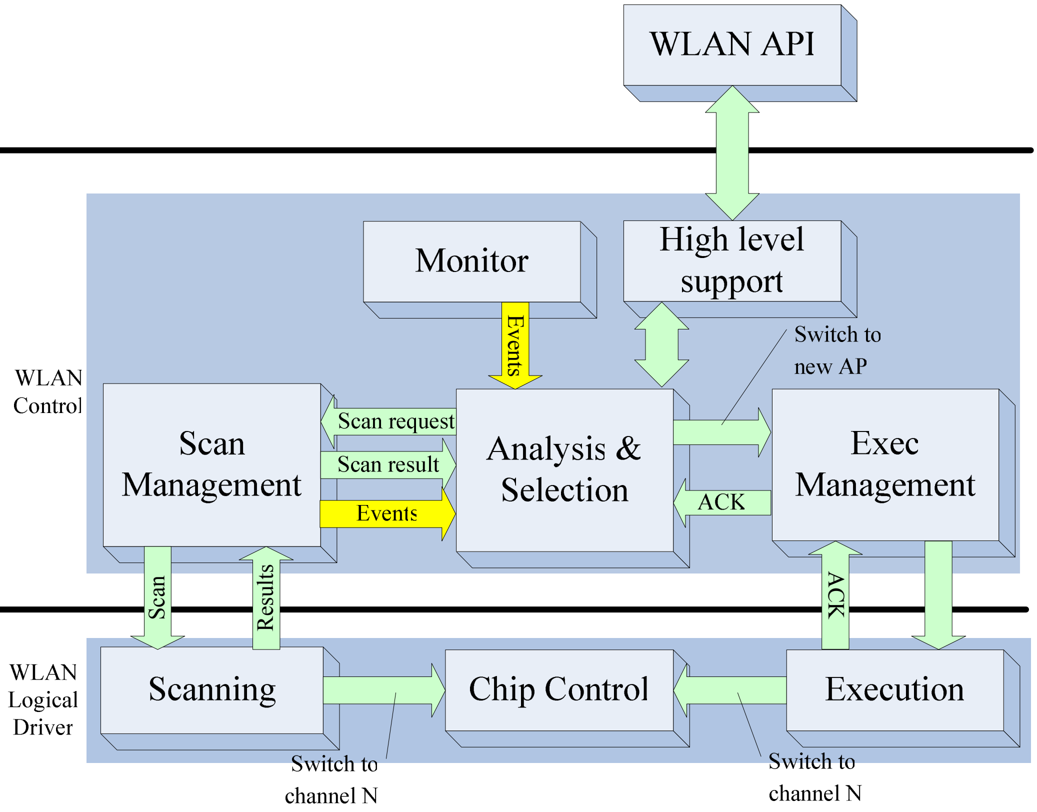


Figure 3-1: Concept modules

Chip Control

The chip control module's only task is to make the WLAN chip change to a certain radio channel. Both the scanning and the execution module will need this functionality.

Scanning

The scanning module will perform an active scan instantly after a scan request from the scan management module. It requires the number of the radio channel to be scanned from the scan management module. After sending a probe request frame on the correct channel, one or several probe response frames might be received. The essential information in these frames will be extracted and handed over to the scan management module.

Scan Management

The scan management module has several different tasks. Firstly, it is has to give scan commands to the scanning module. E.g. if we want 3 measurements of the signal strength of an AP, it has to trigger the scanning module three times.

The next task it has is to collect the results from the scanning module and translate the low level data into the data structure used on the upper level.
In addition to scan requests from the analysis & selection module it can also make events for instance if a new AP is found.

Analysis & Selection
This module analyses the scanning results provided by the scan management module and decides when a handover is required. It also controls how often the scanning is necessary and which channels should be scanned, and it selects which AP to change to and receives events from the monitor module.
Since this module is the main module it also has a relation to the application layer through the High level support module.

Monitor
The monitor module keeps track of the position in the wireless network. It incorporates history and can send events to the analysis & selection module when the history should override a regular handover decision.

High level Support
As indicated in the module name, this module provides the interface towards the application layer.

Exec Management
This module expects a command from the analysis & selection module to switch to a certain AP. It will then pre-authenticate towards this AP and use the execution module as a tool to perform the actual handover. The PMK obtained from the pre-authentication will be forwarded to the execution module.

Execution
The execution module constructs the authentication and association frames related to a handover and also participates in the 4-way handshake for fast re-authentication with PMK caching. It uses the chip control module to physically switch radio channels.

3.2 Relations

Figure 3-2 shows the relations among the modules.

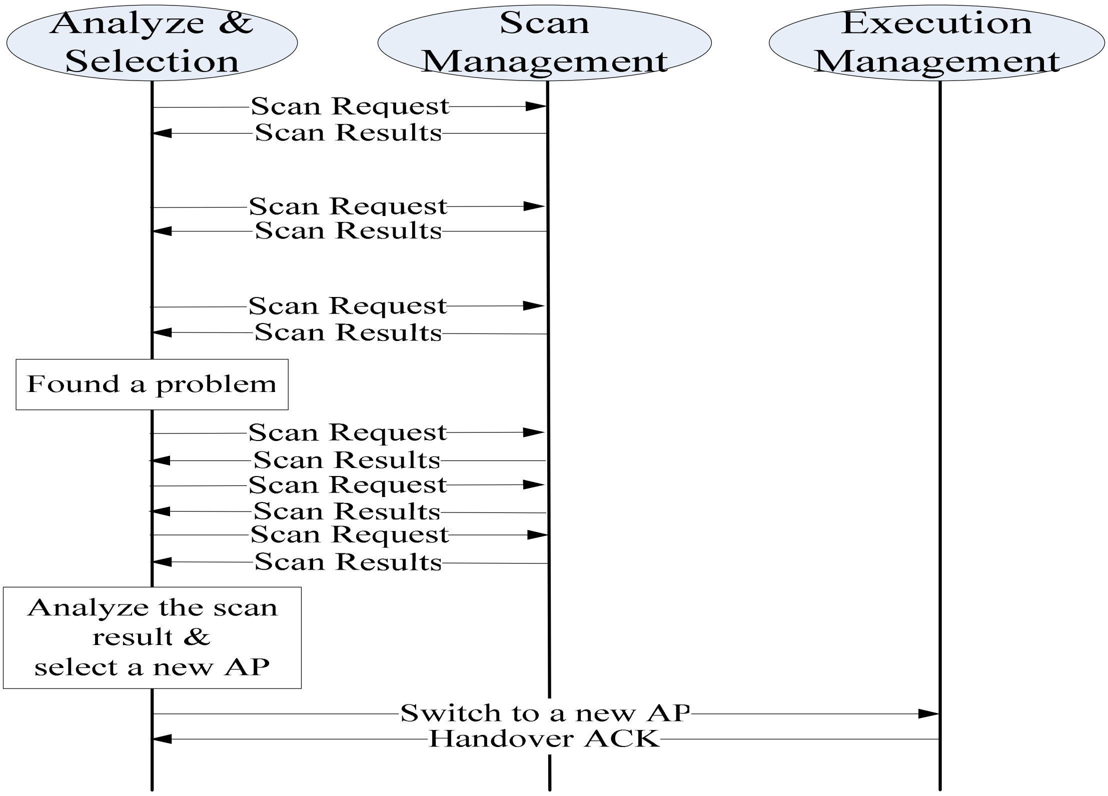


Figure 3-2: Relations overview

The first time a client enters an ESS and connects to an AP, we believe the performance is good. The Analysis & Selection module send "Scan Request" to the Scan Management module to check the connection quality at intervals, for example, 500 ms. Such a "Scan Request" message can trigger a series of scans every 500 ms. Once it gets a "Scan Request" message, the Scan Management module will ask the Scanning module to scan, and gives the scan results to the Analysis & Selection module after the scanning is finished.
Figure 3-3 presents the modules relations in a scanning process in detail.

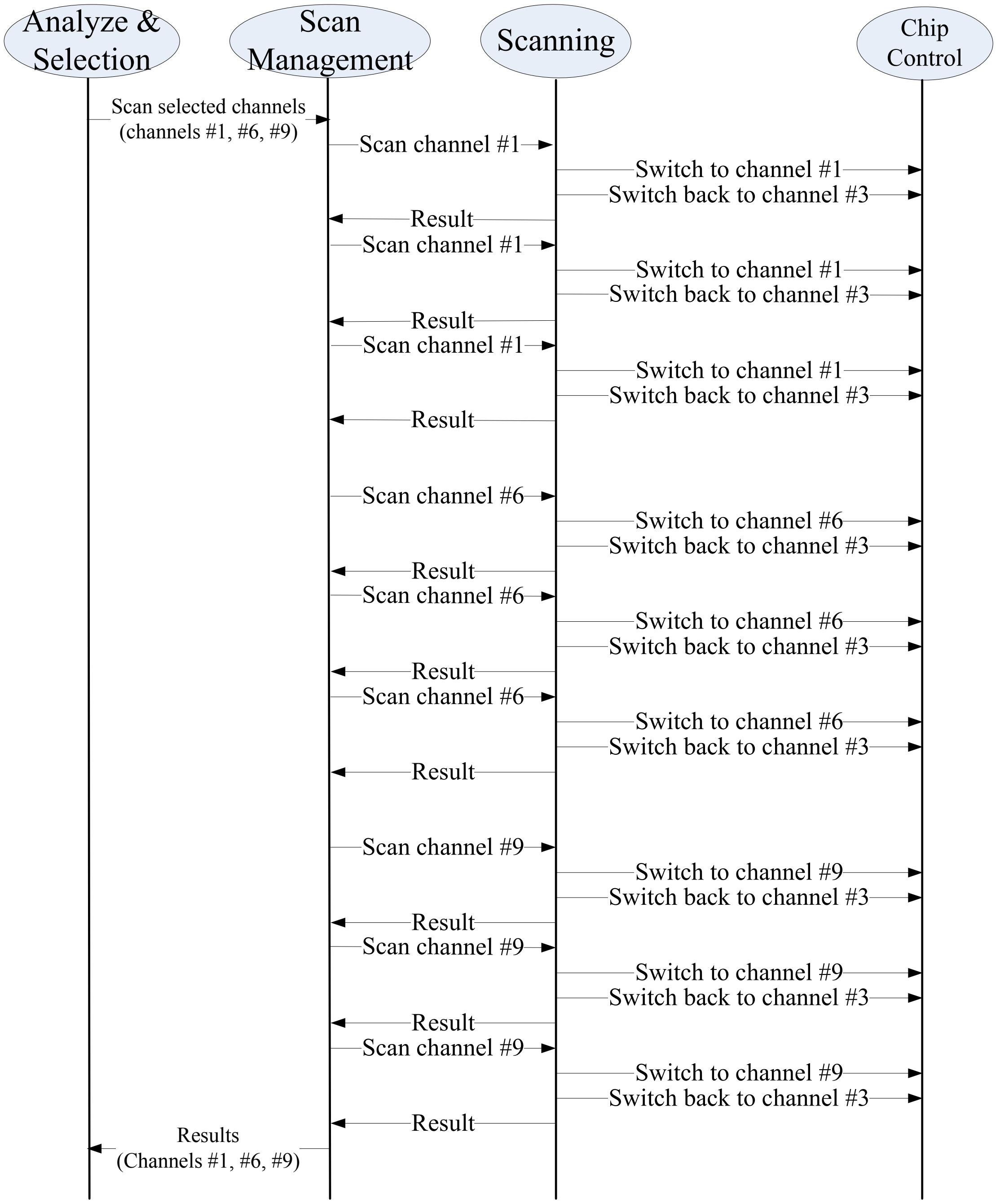


Figure 3-3: Scanning relations

Having received scan missions, the scanning module will scan all channels or all selected channels, and scan each channel many times in order to get smoothed measurements. The scanning can be active or passive. The Chip Control module is used to switch channels. Thus, as the example in figure 3-3, when the Scan Management module ask the Scanning module to scan channel #1, #6, #9, at first the Scanning module has to send message to the Chip Control module to change the client's current frequency to the frequency of channel #1. After scanning, the Scanning module sends the scanning result to Scan Management module. At last, the Chip Control module switches from channel #1 back to the client's old channel, channel #3. The number of times the scanning process will be repeated depends on the measurement smoothing methods. When the scanning on channel #1 finished, it starts to scan channel #6 in the same way, then scans channel #9. After scanning many times on all specified channels, the scanning results will be sent from the Scanning Management module to the Analysis & Selection module at once.

According to the scanning results from Scan Management module, the Analysis & Selection module will detect whether a problem occurs. If there is a problem, for instance, the current AP's performance is poor or worse than another AP nearby. Then, the Analysis & Selection module will send another "Scan Request" to Scan Management module to seek a new AP. The scanning process is the same while the period between two scans will be shorter, 100 ms perhaps.

This time, the Analysis & Selection module will analyze the scanning results and select a new AP to transfer to, and then inform the Execution Management module to switch to the new AP. The Execution Management will give a handover acknowledgement message to the Analysis & Selection module when the handover is done. The execution process will be described particularly in figure 3-4.

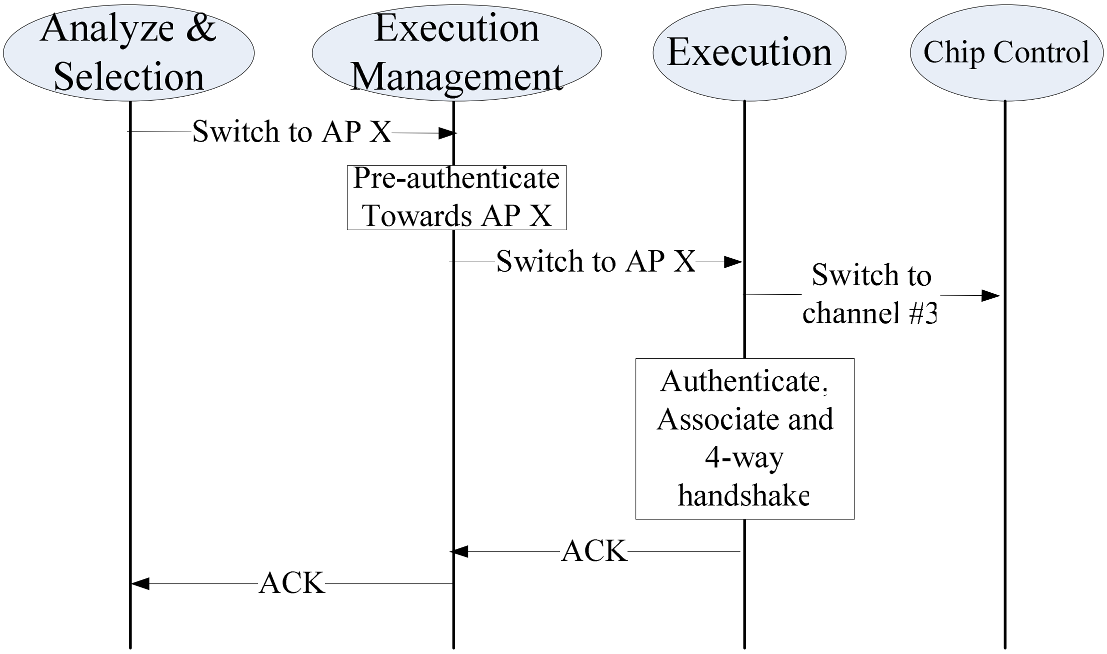


Figure 3-4: Execution relations

Actually, when the Execution Management module receives a handover message from the Analysis & Selection module, it will pre-authenticate towards the new AP first, then tell the Execution module to do the transition. In order to switch to the new AP, the Execution module has to use the Chip Control module. After switching, the Execution module will do the authentication, association and 4-way handshake according to the 802.1x standard on the new channel. In the end, the Execution module informs the Execution Management of the accomplishment of the handover, and the Execution Management module informs the Analysis & Selection module.

3.3 Analysis & Selection Module

As we described in our thesis definition, the purpose of our project is seeking a fast seamless handover algorithm, in which Ericsson is also mostly interested. Thus, the core issue we should focus on is the "Analysis & Selection" module in the concept modules architecture, not other implementation details.

Generally, the Analysis & Selection module gives scanning orders, and analysis scanning results, and decides when and where to hand over.

To introduce the intra function of the Analysis & Selection module in detail, we take the poor performance reaction handover algorithm as an example since it saves power.

When the current AP's performance is good, the Analysis & Selection module sends out a scanning request after a fixed period, only in order to monitor the current AP. After it receives the scanning result, it analyzes the result and checks whether the AP's performance is still good according to some criteria. In the state of the art chapter, we find that the signal strength is an important factor that affects an AP's connection performance, and most of today's solutions are signal strength based (see section 2.2.3). Here we mainly discuss the signal strength based handover algorithm.

- Threshold scheme
  This solution defines a signal strength value as a threshold. Once the AP's signal strength value is less than the specified value, it will trigger a handover. However, the threshold solution may lead to a mass of unnecessary handovers due to the shaky signal strength. Frequent measurement can solve the problem to some extent.

According to the threshold, the Analysis & Selection module analyzes and evaluates the scanning result. If the AP's performance is good, it will send a scanning request again another several milliseconds later. If not, the Analysis & Selection module will prepare to trigger a handover. Thus, it sends scanning requests frequently with the purpose of collecting the performance condition of the surrounding APs.

The Analysis & Selection module gets the scanning results which contain a list of the neighbour APs information, orders them form the best performance one to the worst, then the AP selection phase starts. Below are some selection schemes.

- Best signal scheme
  The scheme allows users to choose the AP with the strongest signal strength. It is simple to implement, but not thorough enough for the complex signal strength changing conditions.

- Trends scheme
  In the trends solution, when the current AP has a negative trend value while a new AP has a positive trend value, a handover will occur. It incorporates history but sometimes does a handover too early, as in the example we discussed in section 2.2.3. Here we will use this scheme only as selection criteria.

- Hysteresis scheme
  The solution initiates a handover only when there is a new AP whose signal strength value is more than the value that the current AP's signal strength value plus a hysteresis value. It prevents the yoyo effect to some extent but consumes much power to pursue better APs. Only the APs with a value, which is more than the current AP's value plus a hysteresis value, are considered. An appropriate hysteresis value is supposed to be defined during the testing period.

According to those criteria, the Analysis & Selection module filters all the APs in the surrounding APs list from the Scanning Management module, and then makes a new list which arranges all the eligible APs from the best to the worst. Normally, it chooses the best one of them as the new AP to switch to. Finally, the Analysis & Selection module will send a message together with the information of the selected AP to the Execution parts.

When the handover is done, the Analysis & Selection module will get an acknowledge message. It will then start a new circle from the beginning, sending scanning requests in intervals to monitor the new AP's performance.

3.4 Our proposed algorithm

Our proposal algorithm is a challenging performance reaction, history incorporated and power saved algorithm which combines the threshold scheme (the threshold scheme reacts to degrading performance and issues a trigger when the signal quality falls below a specific threshold), trend scheme, and the hysteresis scheme. It absorbs their merits and avoids their respective drawbacks by the combination and some improvements.

3.4.1 Parameters' setting

In the proposed algorithm, we use threshold, trend, hysteresis and variance, all of those parameters values should be fixed before the implementation.

- The interval between every two measurements in good condition
  As we explained in chapter 1, with the purpose of saving power, we measure the current AP's signal strength at intervals over a relatively longer period when the connection condition is good. We take 500 ms as an example.

- The interval between every two measurements in bad condition
  There are two conditions that we defined as bad conditions: when we found the current AP's signal strength value is below the threshold for the first time, or when we found there is no better AP than the current one to switch to.

  In these conditions, we have to wait a short period and do a current AP's signal strength measurement again. The interval should be set appropriately, neither too long nor too short. A short interval will lead to inconspicuous changes while a long one might result in the connection breaking. We will set it 200 ms in the beginning and modify its value according to testing.

- Smoothing parameter in the EWMA filter
  In order to avoid the fluctuation of the signal strength, we have to smooth the raw RSSI value, and the EWMA filter is a good method for this. However, the smoothing parameter β in the formula $y(k) = \beta * y(k-1) + (1-\beta) * x(k)$ should be fixed. Here we will use $\beta = 0.9$ as a start point and seek a better value for it in the testing.

- The threshold value
  A good threshold is significant for our algorithm. On one hand the threshold should be low adequately to avoid unnecessary handover. On the other hand the threshold should be high to ensure the client has enough time to do a handover before the connection breaks.

In order to set the threshold, we used Skype to call a client who is moving along the aisle with a laptop. When the client found the connection performance to be poor, we measured the signal strength, and set that value as the threshold. The threshold is probably around – 65 dB.

- The amount of time parameter in the trend formula

  In the trend formula $\alpha = \frac{y(t) - y(t - L + 1)}{L}$, the amount of time L should be defined. The article [11] provides a value 50 ms, but it seems be too short to present the trend. We will propose a larger value in the test, for example 3 s.

- Scanning times and intervals

  To calculate the trends of each candidate AP, we need at least two signal strength values for every AP. Thus we plan to scan 2 times with the interval of L.

- The hysteresis value Δ

  As described in our proposed algorithm, the hysteresis is used for handover detection and classifying the neighbour APs. In our experiments, the hysteresis factor Δ is 5. It means a handover will occur at once if the current AP's signal strength value falls below -70. Also, the signal strength of the 1st class APs (type A and type B), should at least higher than -60.

- The number of samples in the variance formula n

  To calculate the variance of the signal, we have to collect some signal strength value samples. The number of samples is the parameter n in the formula $Var(X) = \frac{1}{n}\sum_{i}^{n}(X_i - \mu)^2$, which $\mu = E(X) = \overline{X} = \frac{1}{n}\sum_{i}^{n} X_i$. Since we scan 3 times, we have 3 raw signal values for each candidate AP. So here we defined the n equal to 3.

3.4.2 Proposed algorithm

Our algorithm also has three parallel aspects: handover detection, candidate APs selection (classification), and the further method to compare two champion candidate APs, which are appraised according to the former criteria in the selection part in our algorithm.

1. Use threshold and hysteresis schemes for detection

We mainly use the threshold scheme to detect when to handover. An appropriate threshold value is vital to our algorithm.

The weakness of the threshold scheme is that many unnecessary handovers will occur due to the instability and fluctuation of the signal strength. To avoid this, we replace the missing value in the time series by -80dB and use the Exponential Weighted Moving Average Filter with a constant factor $\alpha = 0.9$ to smoothen the raw received signal strength value.

We use the threshold scheme frequently, and combine it with a hysteresis to detect and to reduce the times of unnecessary handover. At the beginning, we measure the current AP's signal strength every 500 ms if there is no exception. Once the signal strength drops below the threshold for the first time, we will set up a warning and measure the AP's signal strength every 200 ms for 3 seconds.

During the 3 seconds, if the signal strength rises above the threshold, a handover is avoided and a new circle monitor is started. In contrast, if the signal strength falls below the "threshold-hysteresis" in the 3 seconds, or it still below the threshold after 3 seconds, a handover is triggered. In this way, we can avoid the long time poor connection or connection breaking which is caused by rapidly decreasing signal strength. The latter condition always happens in a building, when a client suddenly moves to the place behind a brick wall.

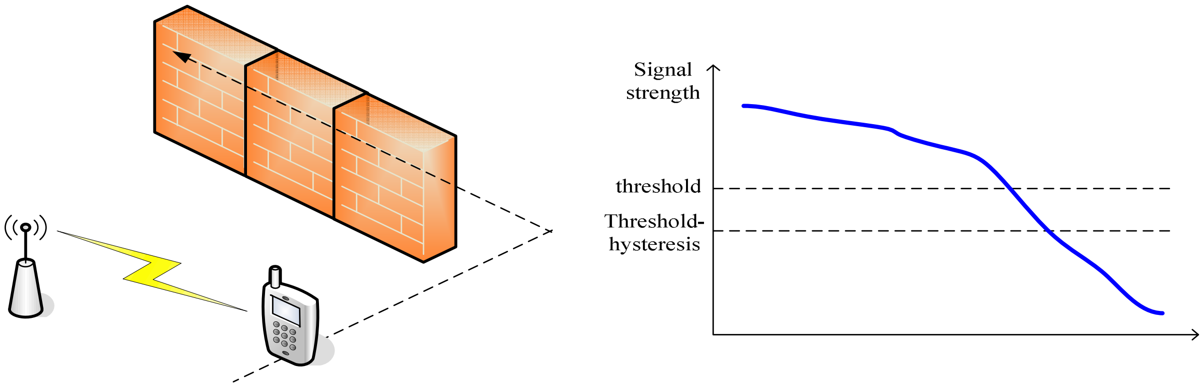


Figure 3-5: Sharply decreased signal strength

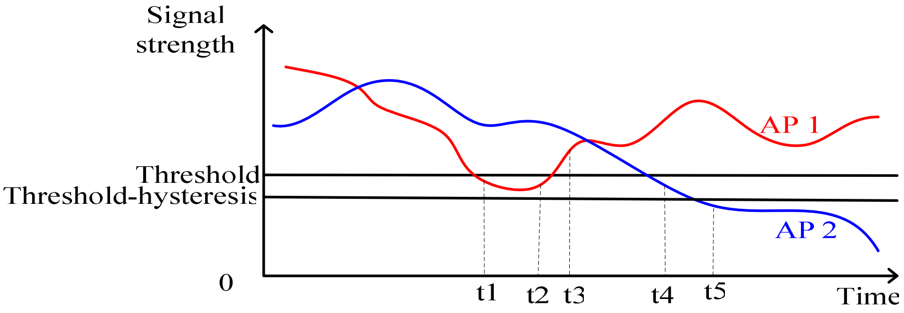


Figure 3-6: Handover detection

For example, the AP 1's signal strength drops down below the threshold at t1, and then after a while, it rises and is higher than the threshold. At t2, the signal strength is still below the threshold while at t3 it goes up and be higher than the threshold. If the period from t1 to t2 is more than 3 seconds, we execute a handover. If the period from t1 to t3 is less than 3 seconds, a handover is unnecessary. Our algorithm then considers this as a good performance and checks it again 500 ms later. But in the case of AP 2, we found its signal strength is lower than the threshold at t4, and it continues to get worse and fall below the "threshold-hysteresis" at t5. A handover will be done at the same time.

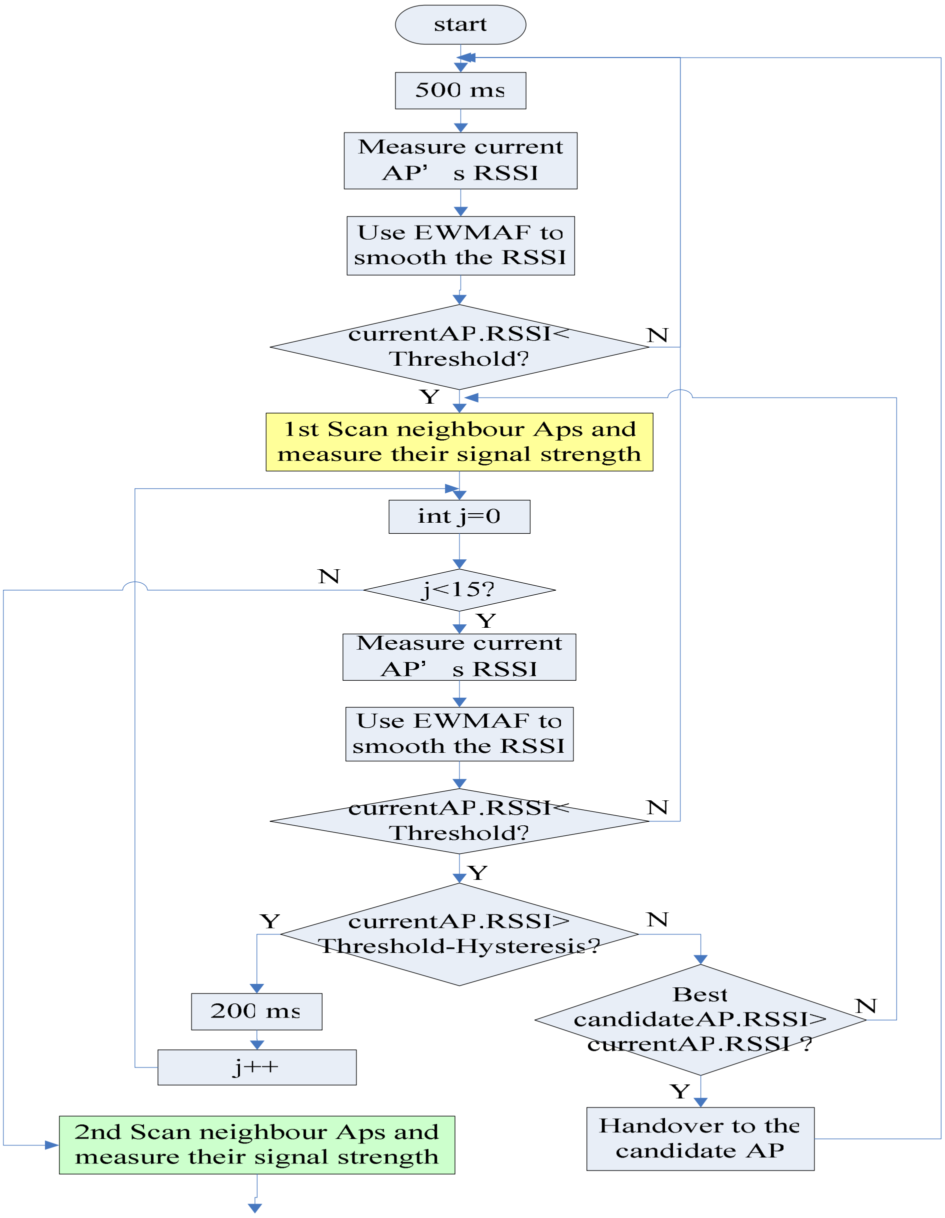

start
500 ms
Measure current AP's RSSI
Use EWMAF to smooth the RSSI
currentAP.RSSI< Threshold?
N
Y
1st Scan neighbour Aps and measure their signal strength
int j=0
N
j<15?
Y
Measure current AP's RSSI
Use EWMAF to smooth the RSSI
currentAP.RSSI< Threshold?
N
Y
Y
currentAP.RSSI> Threshold-Hysteresis?
N
200 ms
j++
Best candidateAP.RSSI> currentAP.RSSI ?
N
Y
2nd Scan neighbour Aps and measure their signal strength
Handover to the candidate AP

Figure 3-7: Flow diagram of handover detection

2. Use signal strength, trend and hysteresis for selection

Once the algorithm finds that the current AP's signal strength falls below the threshold, it scans and measures all the surrounding APs at once.

If the current AP's signal strength falls fast and below the "threshold-hysteresis" within the next 3 seconds, the algorithm will compare the current AP's signal strength with the best signal strength of the candidate APs. If the candidate AP's is better, then the client will switch to it. Otherwise, it will scan continuously until a candidate AP with better signal strength is found.

Another condition to consider: if 3 seconds later, a handover is still needed, then the algorithm will do the scanning and measurements again.

To calculate the trend of each candidate AP, we need their two signal strength values in both two scanning. If the APs can be scanned and measured in both two scanning, we can calculate their trends according to the formula. However, some APs can be found in the first scanning but not the second scanning because the client goes too far away from them, or some APs are newly detected in the second scanning since the client gets close to them. Then the trends of those APs cannot be gotten. For the former case, we ignore those APs since the client cannot connect to them any more, while for the latter case, we set the trends of such APs equal to null.

The algorithm will then classify the candidate APs into different types according to their trends and signal strength in the second scanning result.

First, according to those APs' signal strength values, we use the threshold value and a hysteresis factor to categorize the APs. Those candidate APs can be divided into three classes. We use "neighbourAP[i]" to indicate them.

- 1st class: the AP's signal strength value is more than the current AP's plus a hysteresis factor.
  neighbourAP[i].RSSI >=Threshold +hysteresis

- 2nd class: the AP's signal strength value is less than the current AP's plus a hysteresis factor, but more than the threshold value.
  Threshold <=neighbourAP[i].RSSI <Threshold +hysteresis

- 3rd class: the AP's signal strength value is less than the threshold.
  neighbourAP[i].RSSI< Threshold

We only consider the 1st and 2nd classes and ignore the 3rd class APs since their performances are too poor. Switching to the 3rd class AP will cause another urgent handover.

For both the 1st class APs and 2nd class APs, we will calculate their trends one by one. We will use a value 3 s as the parameter L in the trend formula in our implementation. The trends value can be positive or negative.

Thus, the neighbour APs can be classified into those types:

A. 1st class APs with positive trend values;
B. 1st class APs with negative or zero trend values;
C. 2nd class APs with positive trend values;
D. 2nd class APs with negative or zero trend values;
E. 3rd class APs.

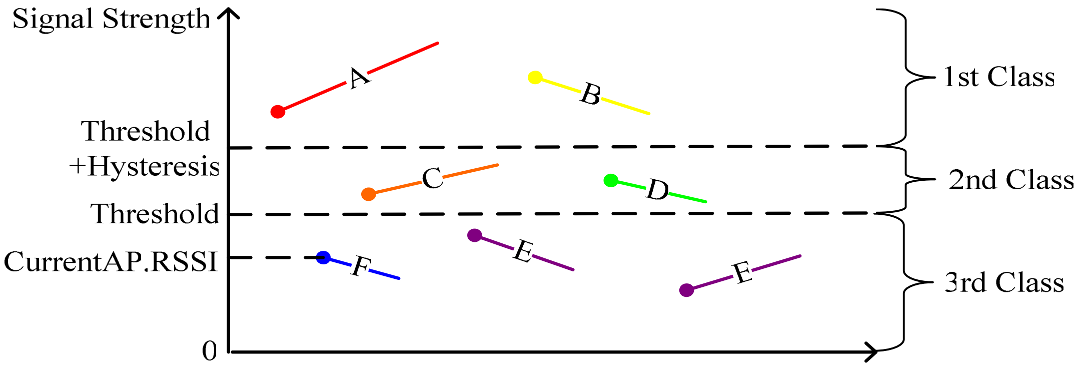


Figure 3-8: Types of candidate APs

Obviously, the A type APs are the best choice. The C type APs are also a good selection. Next, we consider B type. In a worse case, we accept D type APs. The worst option is the current AP. If there is no neighbour AP better than the current one, we have to stay in the current AP and cancel the handover this time. The priority is:

A>C>B>D>current AP (F)>E

Our algorithm will choose a new AP among the candidates from best available types. For the APs with the same priority, we choose the one with the best signal strength according to the second time scanning result.

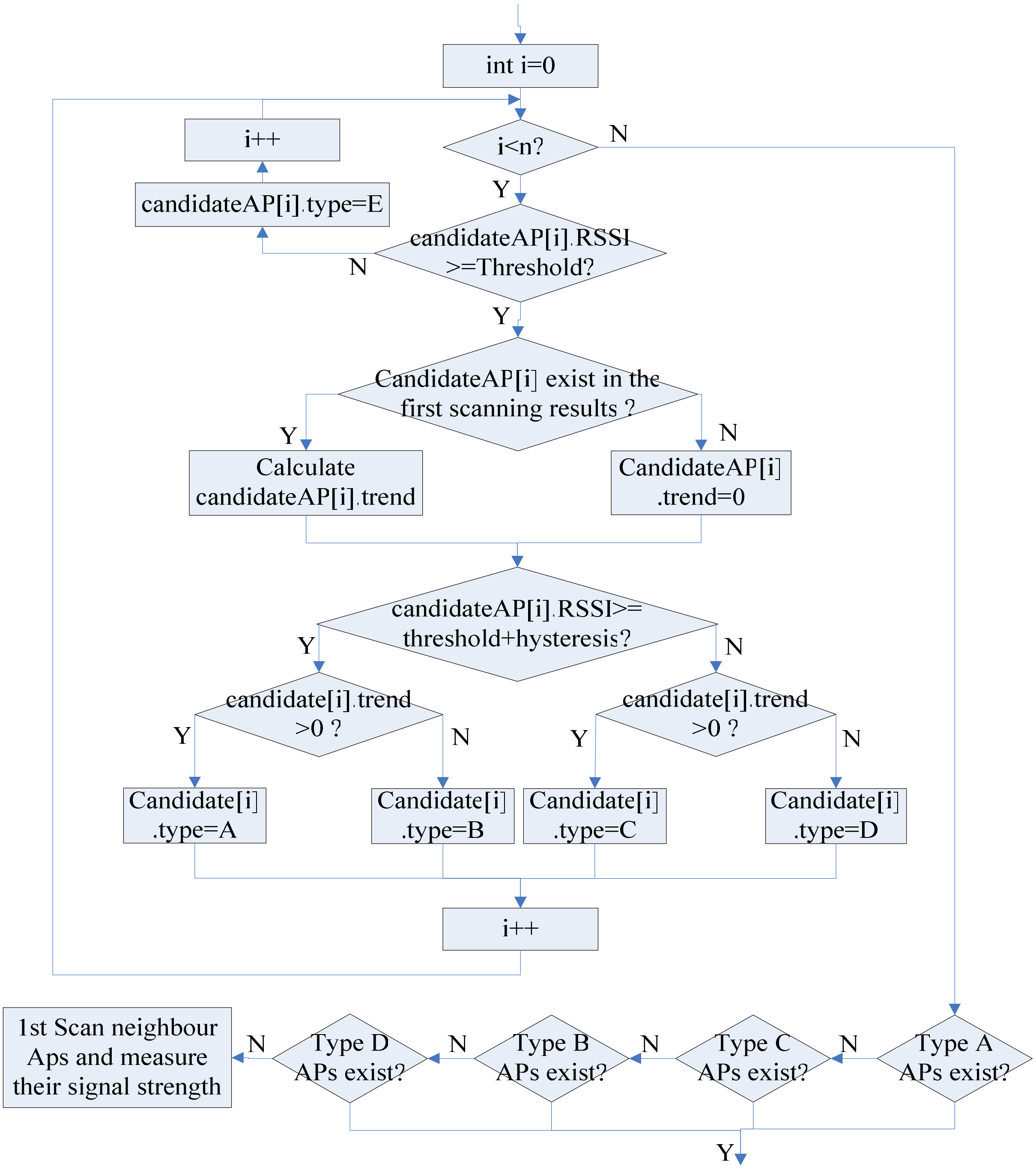


Figure 3-9: the flow diagram of candidate APs classification

3. Use signal variance in case of two best APs

How do we deal with the problem if two different APs have similar signal strengths, and both of them have the highest priority? We defined the two APs' signal strengths are similar if the absolute value of their difference is less than 5 dB.

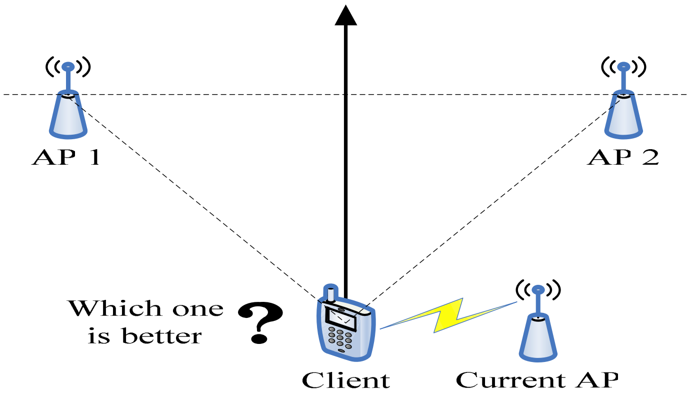


Figure 3-10: Two APs with same priority and similar signal strengths

For example, when a user moves straight, there are two APs nearby and on his or her two different sides, with equal distances to the user. Probably, both AP 1 and AP 2 have a positive trend, and the signal strengths of them are more than currentAP.RSSI + hysteresis, so they are A type APs. More coincidently, their signal strength values are similar. Which one is better?

The signal strength value is not the only criteria to evaluate a signal. We will discuss this issue in detail in the next section. Here in our proposed algorithm, we used the signal variance as the criteria to compare the two best candidate APs.

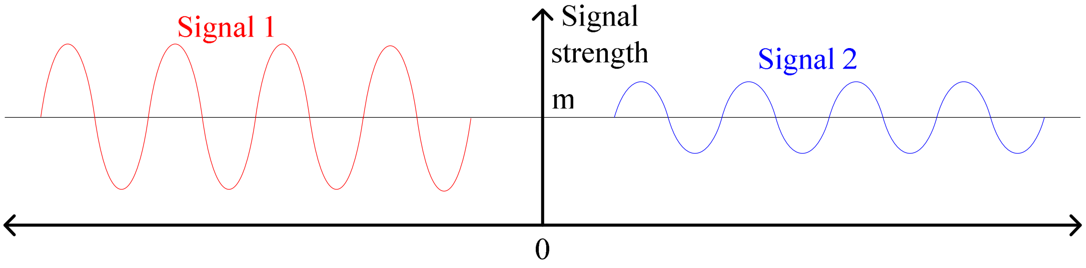


Figure 3-11: Two signals with same strength values but different variances

As we can see, in figure 3-8, the smoothened signal strength values of signal 1 and signal 2 are similar, but signal 1 fluctuates heavier than 2. Apparently, the AP with signal 2 is the preferred one since its signal strength is more stable. To compare the signal's stable extent quantitatively, we import the concept of "variance".

"In probability theory and statistics, the variance of a random variable (or somewhat more precisely, of a probability distribution) is a measure of its statistical dispersion, indicating how its possible values are spread around the expected value."[40] The more heavily a signal shakes; the lager its variance value will be.

According to [40], the computational formula for variance is $Var(X) = E(X - E(X))^2$. If $\mu = E(X) = \overline{X} = \frac{1}{n}\sum_i^n X_i$ is the expected value (mean) of the random variable X, then the variance is $Var(X) = E((X - \mu)^2)$. And if the random variable is discrete, the formula is equal to $Var(X) = \frac{1}{n}\sum_i^n (X_i - \mu)^2$.

So in the arranged candidate APs queue, if there are two best APs, we will compare them by their signal variance value. The AP with a lesser variance value will be the actual best one. Then we will consider the best AP as the new AP which the client will switch to, and execute the handover.

The figure below shows the general flow of our proposed algorithm.

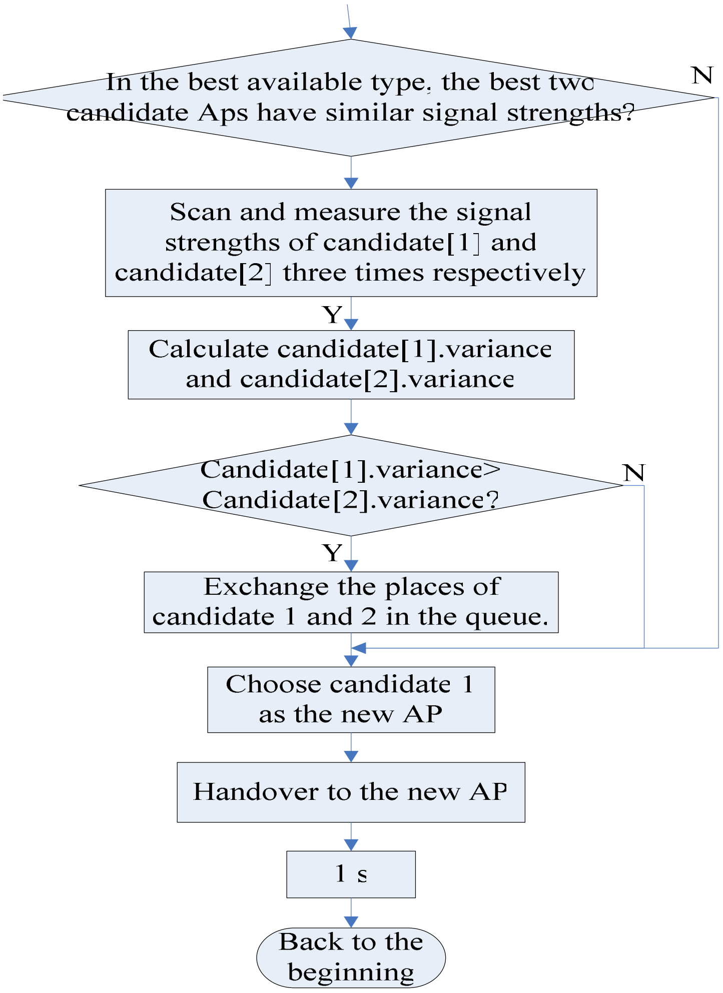

In the best available type, the best two candidate Aps have similar signal strengths?
N
Scan and measure the signal strengths of candidate[1] and candidate[2] three times respectively
Y
Calculate candidate[1].variance and candidate[2].variance
Candidate[1].variance> Candidate[2].variance?
N
Y
Exchange the places of candidate 1 and 2 in the queue.
Choose candidate 1 as the new AP
Handover to the new AP
1 s
Back to the beginning

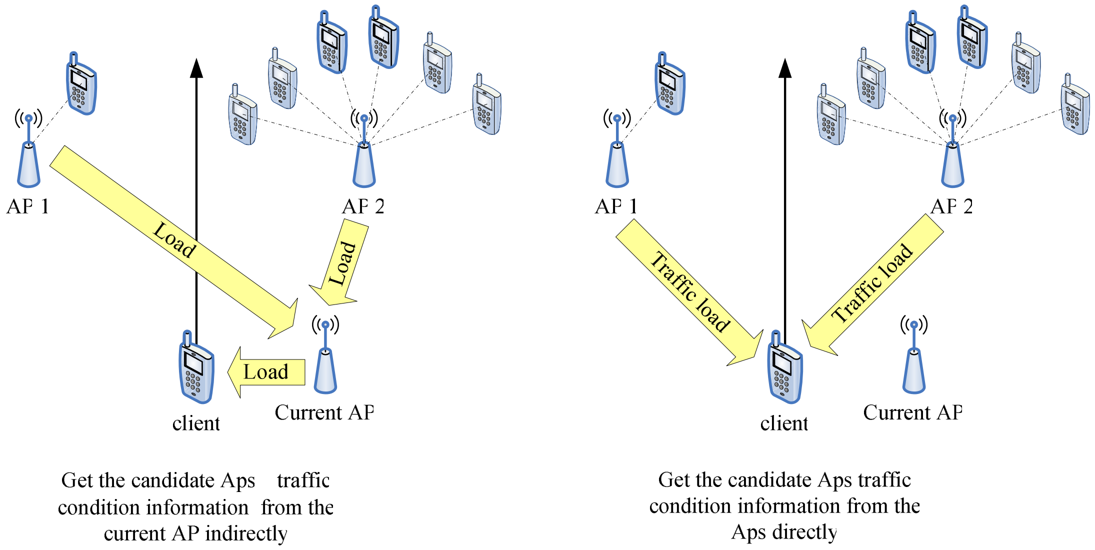
Figure 3-12: the flow diagram of candidate APs selection

3.4.3 Some more complex criteria

In the condition we discussed in figure 3-6, we choose the signal variance as the criteria to compare the two best candidate APs. The reason is that the signal variance is easily obtained. It only requires some discrete signal strength values from the AP, which we have already gotten in the early step of our proposed algorithm, and a variance calculation formula. There are also some other criteria to solve the problem, but most of them are complex and hard to be implemented.

- QoS support
  If the candidate AP can give QoS guarantee in the form of WMM or 802.11e, it will be a preferred choice.

- Traffic load of the candidate APs
  When the signal strengths are equal, the AP with a lighter traffic load will offer better performance. In those figures below, AP1 is better. The client can know the traffic load conditions of the candidate APs by two ways:



Get the candidate Aps traffic condition information from the current AP indirectly

Get the candidate Aps traffic condition information from the Aps directly

Figure 3-13: two ways to get the candidate APs' traffic conditions

-- *Ask the current AP indirectly*
The client ask the current AP for the candidate APs' traffic load, then the current AP gathers the information and sends it to the client. As far as we know, this process can be implemented practically.

As the "load balancing algorithm" (one of the throughput based solutions we discussed in section 2.2.3) indicated, in each AP, there is a Load Balancing Agent (LBA) that periodically broadcasts its AP's load condition to the others. So in this way, the current AP can get the candidate APs' traffic load condition according to their broadcasting reports, and then it informs its client. However, such mechanism is complex and is installed in the APs, not in the handheld applications. Thus it is beyond our research for this report.

-- *Ask the candidate APs directly (by air)*
The other way to get the candidate APs traffic load conditions is through the client sending requests by air to ask those APs directly. It would be efficient if the process could be done by today's techniques.

Although the station cannot ask the candidate AP directly, it can switch its channel and listen to the RTS/CTS (request to send/clear to send) messages of the candidate AP.

> "RTS/CTS (Request to Send / Clear To Send) is the mechanism used by the 802.11 wireless networking protocol to reduce frame collisions. A node wishing to send data initiates the process by sending a Request to Send frame (RTS). The destination node replies with a Clear To Send frame (CTS)." [41]
>
> Since both the resource node and the destination node can be an AP or a client, the RST/CST messages can be transferred from an AP to a client, from an AP to an AP, from a client to a client, or from a client to an AP. Thus, the amount of RST/CST messages can to some extent reflect the AP's traffic load, but not exactly. So we cannot use it as criteria to compare the traffic loads of the candidate APs.

- Lost beacon of candidate APs

  If the other factors of AP's performance are the same, the AP which loses the least beacons per second is the better one. But currently, the number of lost beacons can only be measured for the current AP and not for the candidate APs'.

4 Implementation & Tools

4.1 Linux and WLAN drivers

The reason for using Linux was to be able to easily use Linux commands to control the WLAN card. At first we tested the popular Madwifi driver which is an open source driver for Atheros based wireless network cards [40]. Due to earlier personal experiences the Linux distribution Ubuntu [41] was chosen. Ubuntu 6.10 was downloaded and installed on a standard Intel Centrino laptop.

During implementation a problem with the Madwifi driver was discovered, it uses so-called background scanning, which means that it only scans when the WLAN interface has been idle for a certain time period. We discovered that it could take several minutes before scan results were updated. We needed to scan much more often than that and were forced to find a solution. We found two parameters which controlled the background scanning interval and how long the idle time would be. Tuning these parameters helped us a little, but it was still not good enough.

IPW2200 [42] is another driver made for Intel PRO/Wireless BG2200 wireless network cards. We found out that this driver actually does a real scan on request and it was therefore better for our purpose.

4.1.1 Networking and Shell Commands

In Linux, shell commands can be used to get information about the current connection, other access points, and to control the WLAN card behaviour. All networks cards; Ethernet as well as wireless can be configured with an IP-address, submask and gateway through the shell command ifconfig.

All wireless cards also have wireless specific commands in addition to ifconfig to set different parameters such as mode, frequency, and SSID. Iwconfig and iwlist are used for these purposes.

Most WLAN drivers also support some private commands in addition to the standard commands defined in Linux Wireless Extensions. The “official command” for private commands is iwpriv, but many drivers also use other commands.

4.1.2 Important Features

By reading the User Guide provided along with the Madwifi driver we could easily find out which commands would be helpful for us. All command examples use ath0 as the wireless interface; this could be something else on a different computer.

Firstly we wanted to get the signal strength of the currently associated access point which can be done as follows:

```
# iwconfig eth1
```

To get a list of all access points/cells in range iwlist is used for scanning:

```
# iwlist eth1 scan
```

To disable automatic scanning we can use this private command:

```
# modprobe ipw2200 associate 0
```

Another private command allows us to disable automatic handover so that we can choose which AP the station should be associated with without any driver overrides.

```
# modprobe ipw2200 roaming 0
```

Finally, to manually connect to a new access point we just need its MAC address and use this command:

```
# iwconfig eth1 ap 00:60:1D:01:23:45
```

4.2 Implementation Language & Tools

Originally we planned to use the C language to implement our algorithm, but because we have no direct interface towards the driver we decided to go with Java 2 Standard Edition [43]. The reason for this decision is that the execution speed of the language is not that important as long as we use shell commands that are slow. Another reason was that development is faster in Java than in C, which is important in a time limited project like this.

The natural choice of Integrated Development Environment (IDE) to go with Java was Eclipse [44]. It is a well known and widely used open source IDE. Eclipse also has support for UML with the help of a plug-in from Omondo [45].

For testing we found two tools made by the Naval Research Laboratory in Washington DC [46]. Multi-Generator (MGEN) is a tool that can generate, send and receive UDP packets. One is the sender and another one is the receiver. The receiver logs all received packets with a timestamp to a log file. The other tool, Tcpdump Rate Plot Real-time (TRPR) can search through the MGEN log files, find packet loss and calculate statistics for it.

4.3 Model

To illustrate how the algorithm is implemented, a UML class diagram was made and is presented in this subchapter.

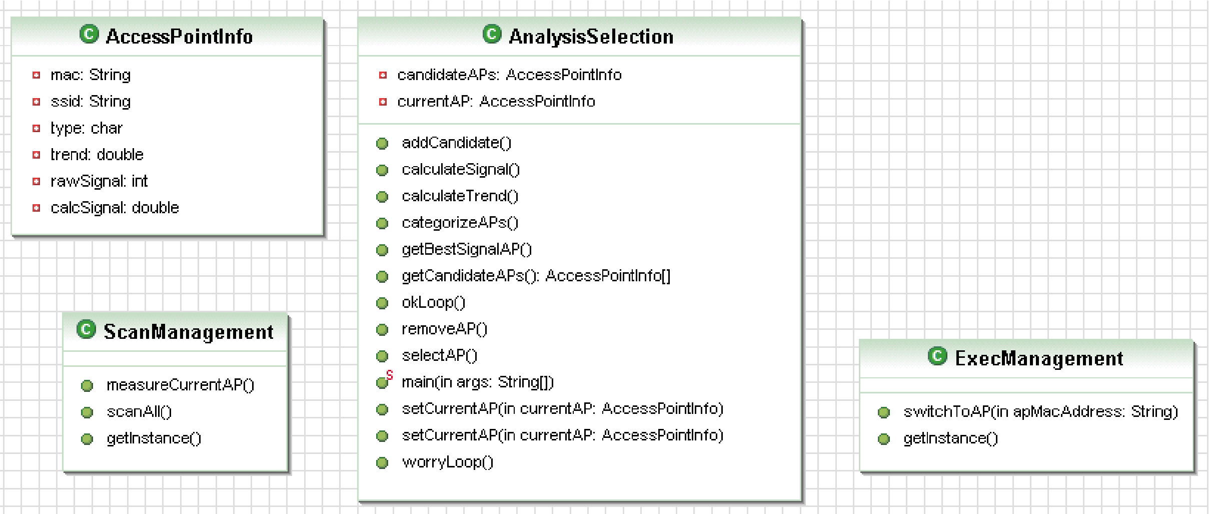


Figure 4-1: UML model

The design closely follows the concept modules except for the AccessPointInfo class which is a support class for the AnalysisSelcetion class that stores data for an access point. Data we needed to store includes MAC-address, SSID, type, trend, raw signal value and calculated signal value. As we explained in chapter 3.3 we will focus on the Analysis & Selection module and therefore the ScanManagement and ExecManagement classes are quite simple.

The ScanManagement class has two methods, one for retrieving measurements of the current AP and one method for scanning for other APs. The ExecManagement class has one method for switching to an AP and takes the MAC address of that AP as a parameter.
The AnalysisSelection class has the main method which has an infinite loop that keeps the algorithm running. It holds methods for different sub loops, methods for calculating trend and signal strength, methods for categorizing the candidate APs and methods that decide which AP to handover to.

4.4 Code Examples

In this subchapter some code examples from our algorithm implementation will be presented and explained.

4.4.1 Run Shell Commands and Catch Output

To run a shell command in Java is very simple and can be done with just one line of code.

```
Runtime.getRuntime().exec("<command>");
```

But to catch the standard output the command gives, an empty object of the class *Process* has to be made. The exec method used to execute the command then returns the process that performs the command.

```
Process shellProcess = null;
shellProcess = Runtime.getRuntime().exec("iwconfig eth1");
```

From the returned process object the inputstream can be collected with the help of the *InputStream* and *BufferedReader* classes.

```
InputStream istr = shellProcess.getInputStream();
BufferedReader br = new BufferedReader(new InputStreamReader(istr));
```

Now the standard output from the command can be read line by line in a while or for loop.

```
String line;
while ((line = br.readLine()) != null) {
      //Do something with line
}
```

4.4.2 Parsing Necessary Information

All the output that a command gives is not important to us. Therefore it is necessary to parse the information that we need to make the algorithm work. We also have to transform numbers of *String* type into *int* type.

The output of the iwconfig command looks like this.

```
eth1      IEEE 802.11g  ESSID:"HiAGroos"
          Mode:Managed  Frequency:2.472 GHz Access Point: 00:12:44:B0:EF:E0
          Bit Rate:54 Mb/s   Tx-Power=20 dBm   Sensitivity=8/0
          Retry limit:7   RTS thr:off   Fragment thr:off
          Encryption key:off
          Power Management:off
          Link Quality=89/100  Signal level=-33 dBm  Noise level=-85 dBm
          Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
          Tx excessive retries:0  Invalid misc:0   Missed beacon:4
```

The information we need is just the MAC-address, ESSID, bit rate and signal level. The word that is in front of the value we want we have called the keyword. Start and end is supposed to hold the indexes of where the value starts and ends.

```
String keyWord = "";
String value = "";
int start, end = 0;
```

Here is an example of parsing the MAC-address from an iwconfig output. The keyword is “Access Point: “ and we check if the line contains that keyword. If it does, the value starts at the end of the keyword and ends at the first blank character after the beginning of the keyword. Now the value can collected with help from the substring method.

```
keyWord = "Access Point: ";
if(line.contains(keyWord)) {
      start = str.indexOf(keyWord) + keyWord.length();
      end = str.indexOf(' ', start);
      value = str.substring(start, end);
}
```

4.4.3 Categorizing APs

To categorize the candidate APs a for loop is used to check all the candidates

```
private void categorizeAPs() {

      for(int i = 0; i < candidateAPs.length; i++) {
            if(candidateAPs[i].getRawSignal() >= (threshold + hysteresis)){
                  if(candidateAPs[i].getTrend() > 0.0) {
                              candidateAPs[i].setType('A');
                  }
            }
            if(candidateAPs[i].getRawSignal() >= (threshold + hysteresis)){
                  if(candidateAPs[i].getTrend() <= 0.0) {
                              candidateAPs[i].setType('B');
                  }
            }
            .
            .
            .
            .
      }
}
```

5 Experiments and Evaluation

5.1 Measuring Scan Delay

The scan delay was measured by saving the time in milliseconds (UNIX time) before a scan and comparing it to the time right after the scan was finished. See the expression below.

Scan Delay = Time in msec after scan – Time in msec before scan

We wanted to know if the scan delay was dependent on how many access points there are in range and if movement (walking) affects the scan delay. We did 1000 scans for each case and the interval between each scan was set to 100 ms. The results are shown in table 5-1 and the values are in milliseconds.

	Stationary 1 cell	Stationary 5 cells	Stationary 8 cells	Walking 4-11 cells
Min	252	253	253	253
Max	282	499	310	282
Average	257.056	258.629	258.502	257.636
Median	257	258	258	258
90 %	257	262	262	258

Table 5-1: Scan delay measurements

The measurements show that the scan takes roughly 250-260 milliseconds. Even though no packets are lost due to IEEE’s power save mode this delay is definitely noticeable in a VoIP scenario. As we proposed in the concept, scanning one channel at a time and letting data traffic through between the scans will make the scanning less disruptive.

5.2 Set-Up and Test-bed

The hardware we need is a handheld application, such as laptop, PDA or WLAN supported cell phone. And we also need an wireless network card which is used for AP’s performance measurements. The card we used in our project is D-Link DWL-AG600.

As for software, we installed the Linux OS on the client side. Here we used Ubuntu 6.10. Then we installed IPW2200 which is made for Inter PRO/Wireless BG2200 wireless network cards. Our proposed algorithm is coded in Java, thus the programming environment “Eclipse 3.2.1” is installed. And in order to test our algorithm in some user scenarios, we also installed some user applications like Skype, media player and FTP.

Of course, besides the client side, we also need a WLAN ESS with many APs. We did all our testing by walking along two aisles on the first and third floor in the building of HiA Grimstad.

5.3 Parameters’ Measurements

Signal strength, trend, signal variance and the number of lost packets are four main quality parameters to present the performance of an AP. We will do this testing with a portable AP from HiA and our testing is distance based. We will start from the centre of the current AP straight to its boundary, and measure all the parameters meter by meter.

Signal strength is the most important parameter in our algorithm. We can measure the APs' raw signal strength values directly via the IPW2200 WLAN card. And the smoothed RSSI values can be gained from the raw data by EWMA filter, calculating according to the EWMA formula.

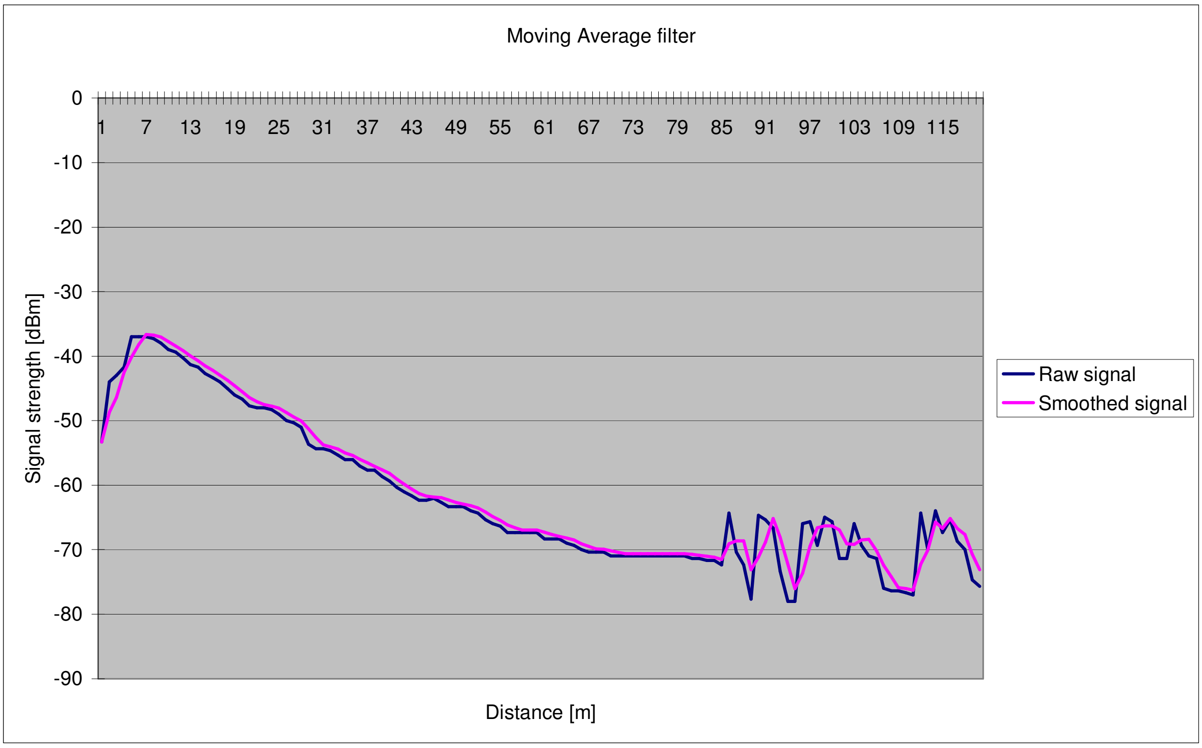


Figure 5-1: Moving Average Filter smoothed signal

For the EWMAF, we have to set a value for the factor β in its formula. The figures below show the smoothed signal strength by the EWMA filter with different β values.

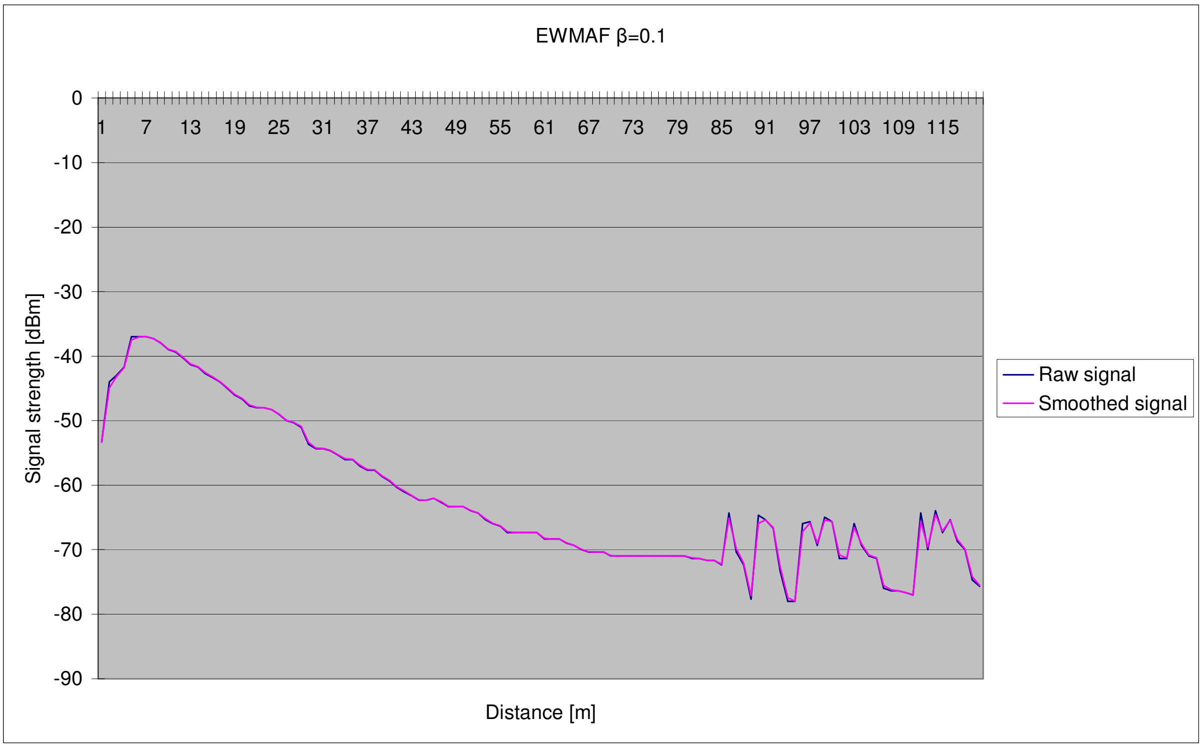


Figure 5-2: EWMAF with β =0.1

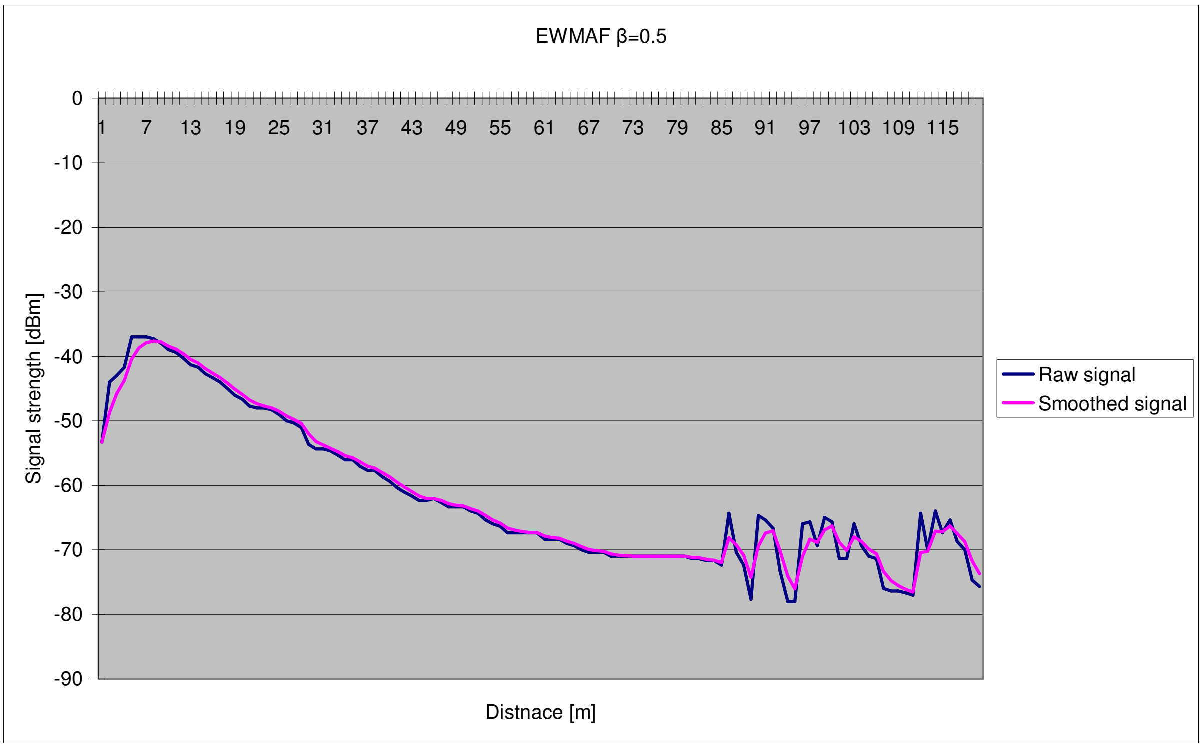


Figure 5-3: EWMAF with β =0.5

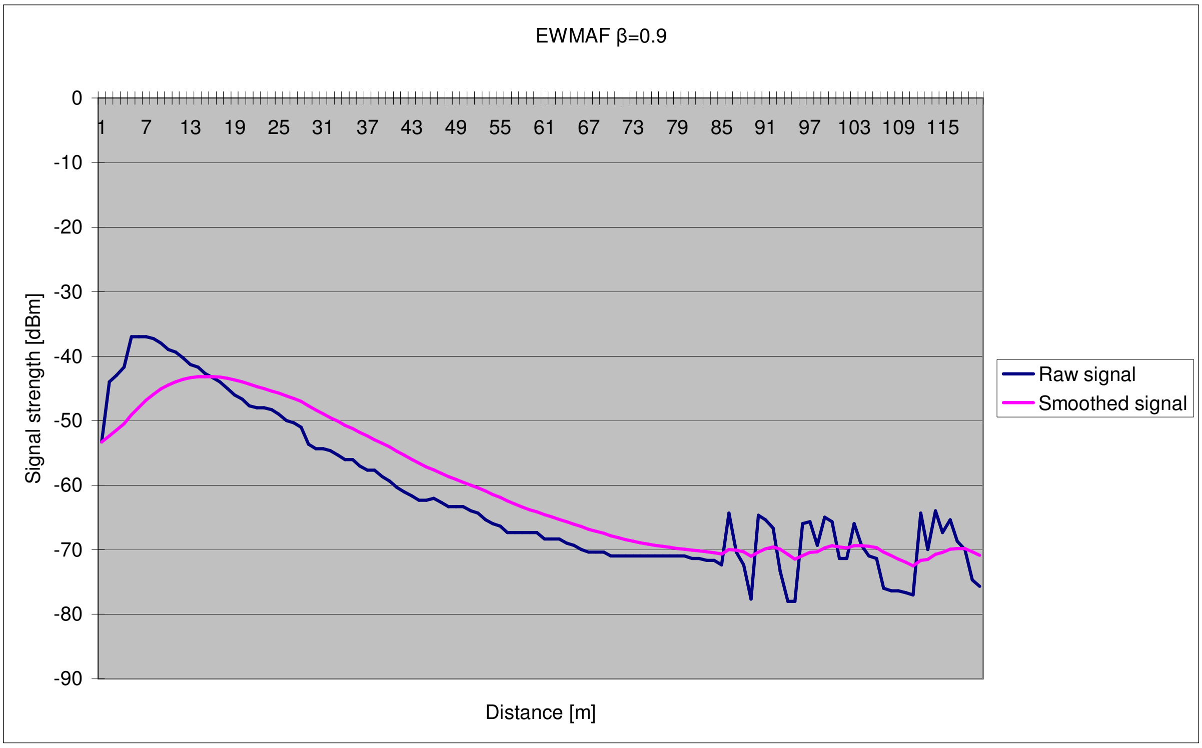


Figure 5-4: EWMAF with β =0.9

With those figures we can easily find the curve which is smoothed by EWMA filter with β =0.9 being the mildest. The more the signal strengths are smoothed, the more signal fluctuations and unnecessary handovers can be avoided. Thus we will use the EWMA filter with β =0.9 to smooth the raw signal strength in our algorithm.

With the raw signal strength values, we can calculate the trends and the signal variances according to their formulas respectively. The trend formula requires the signal strength values at time point (t) and (t+L-1), while the variance formula needs some samples of the signal strength values. Since this testing is distance based, we use 1 meter, 2 meters and 5 meters for the L. And the number of samples for variance is 3.

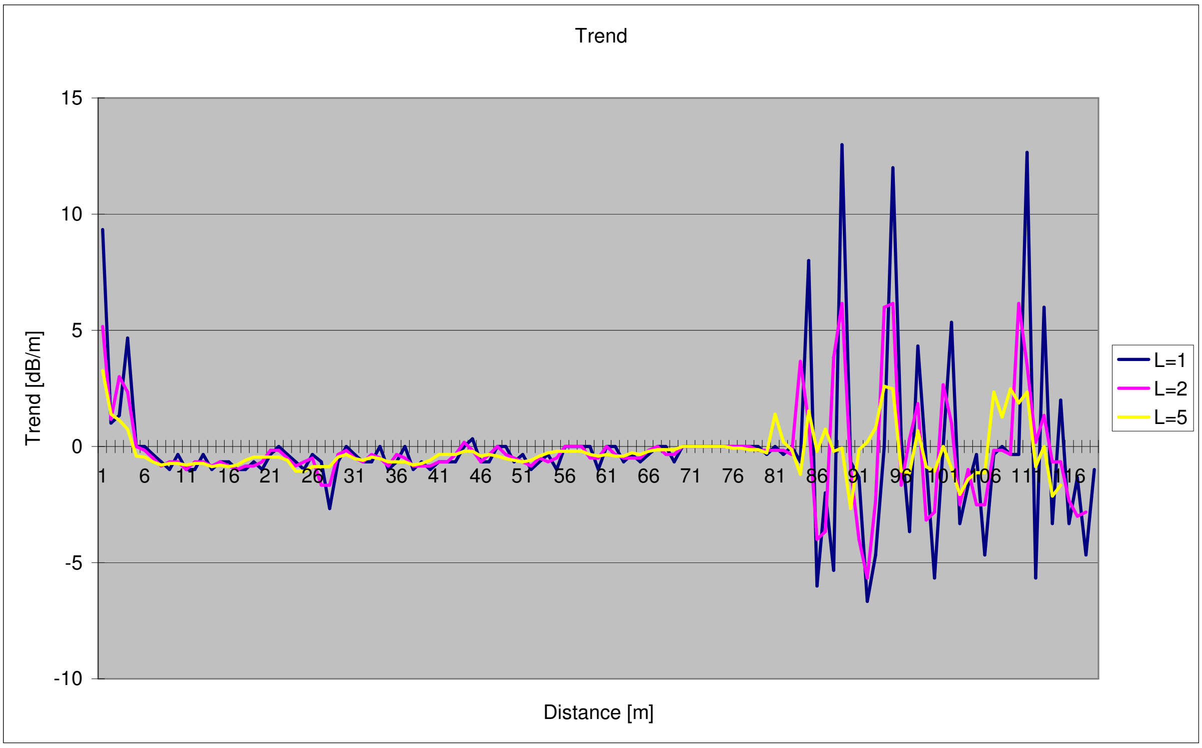


Figure 5-5: Signal trends with different L (using raw signal strengths)

As the figure shows, the longer L gives the more general trend of the signal strength.

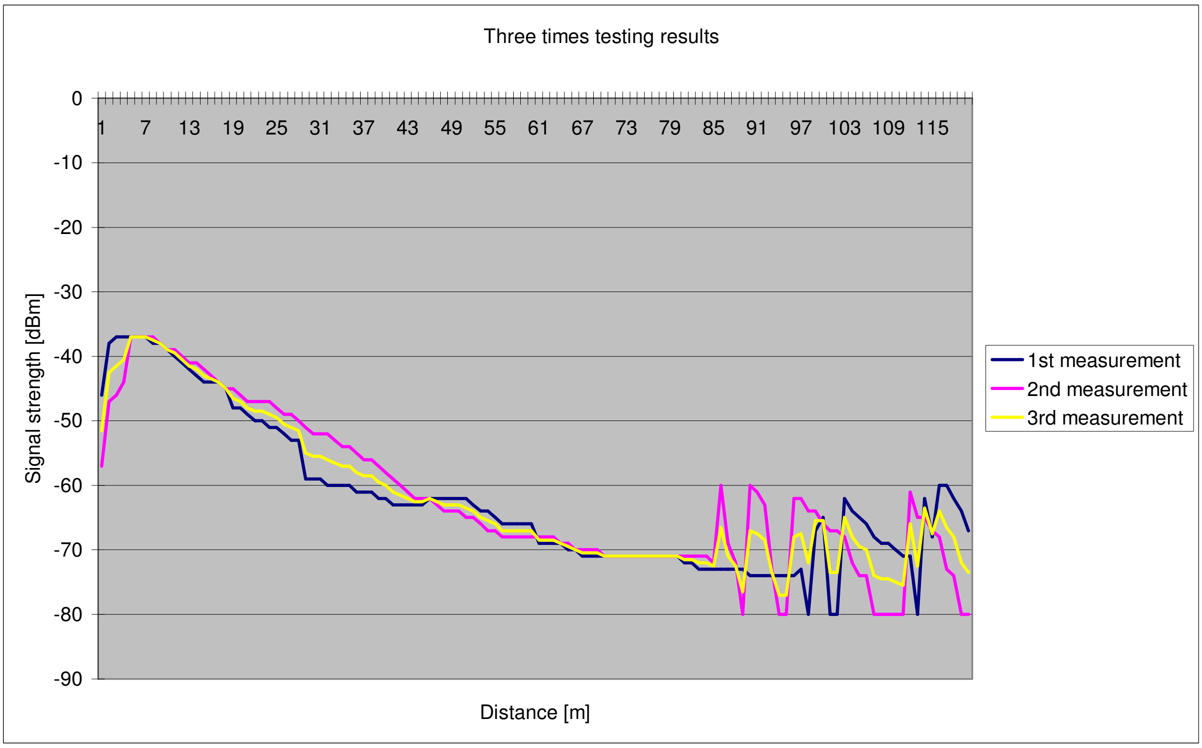


Figure 5-6: Three times signal strength measurements results

We got three signal strength values at every meter from three times measurements. At one place, the variance is little when the three signal strength values are similar, while large when the three signal strength values are absolutely distinct. Thus we can easily found in figure 5-7 that in the periods: from 0 meter to 6 meters, from 21 meters to 41 meters, from 86 meters to

120 meters, the signal strengths are dissimilar. Accordingly, in figure 5-8, the variance values are large in these three periods.

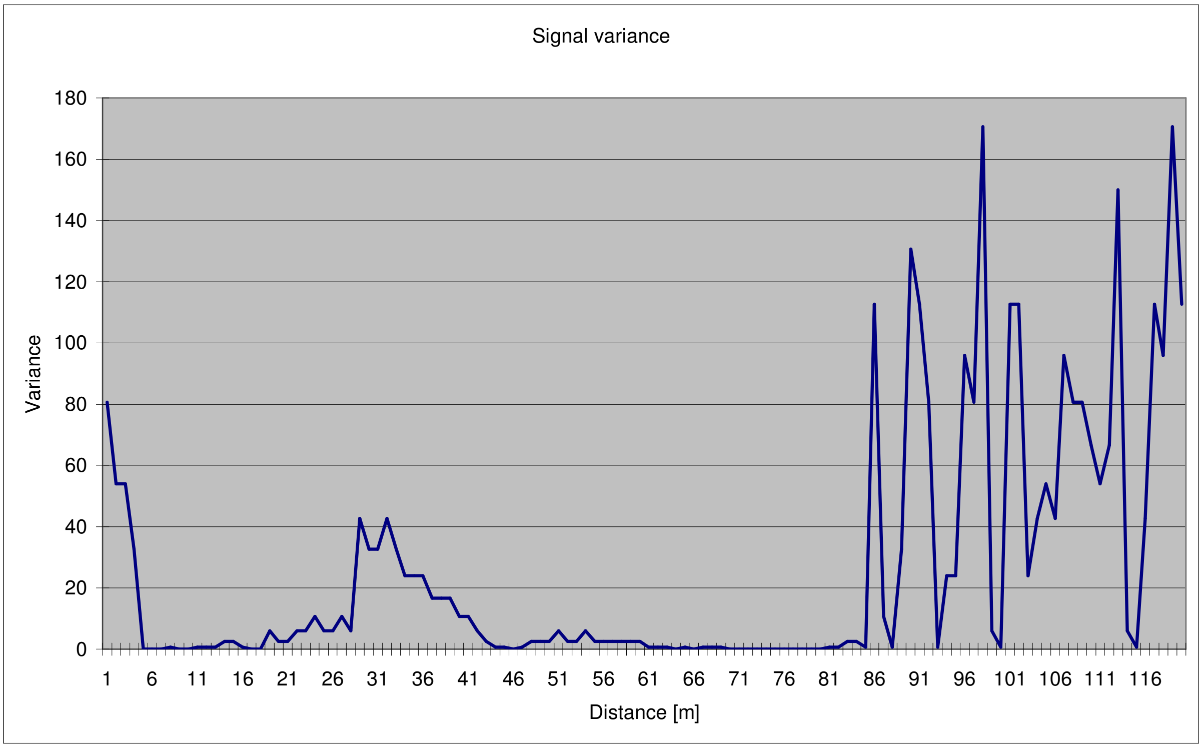


Figure 5-7: Signal variance

5.4 Test Cases and Evaluation

5.4.1 User Scenarios

As we discussed in chapter 1, different user applications require different features of the handover algorithm. Skype emphasizes the real-time, mobile TV needs wide bandwidth, and FTP has high packet loss sensitivity. We will test our algorithm in all three user scenarios to see whether it works well. Yet most of our attentions will be placed on the VoIP scenario.

5.4.2 Test Cases and Evaluation

In order to test how the algorithm works, we tried to list test cases as much as possible, covering all the possibilities in our algorithm.

1. Signal strength fluctuation

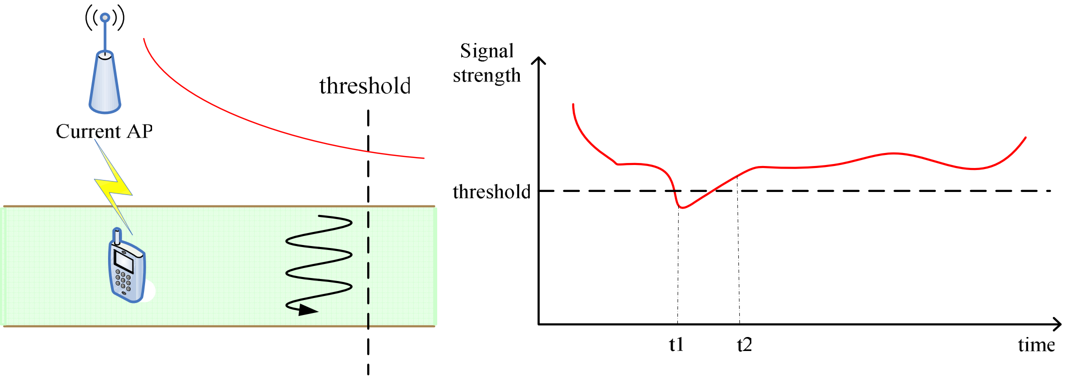


Figure 5-8: Signal strength fluctuation

When the signal strength is close to the threshold, the signal fluctuation will lead to a temporary signal strength drop which falls below the threshold. A handover here is absolutely unnecessary. If we check the signal strength a period later, we will find the signal strength is up to the threshold already. As we defined, our algorithm will not make a handover in this condition.

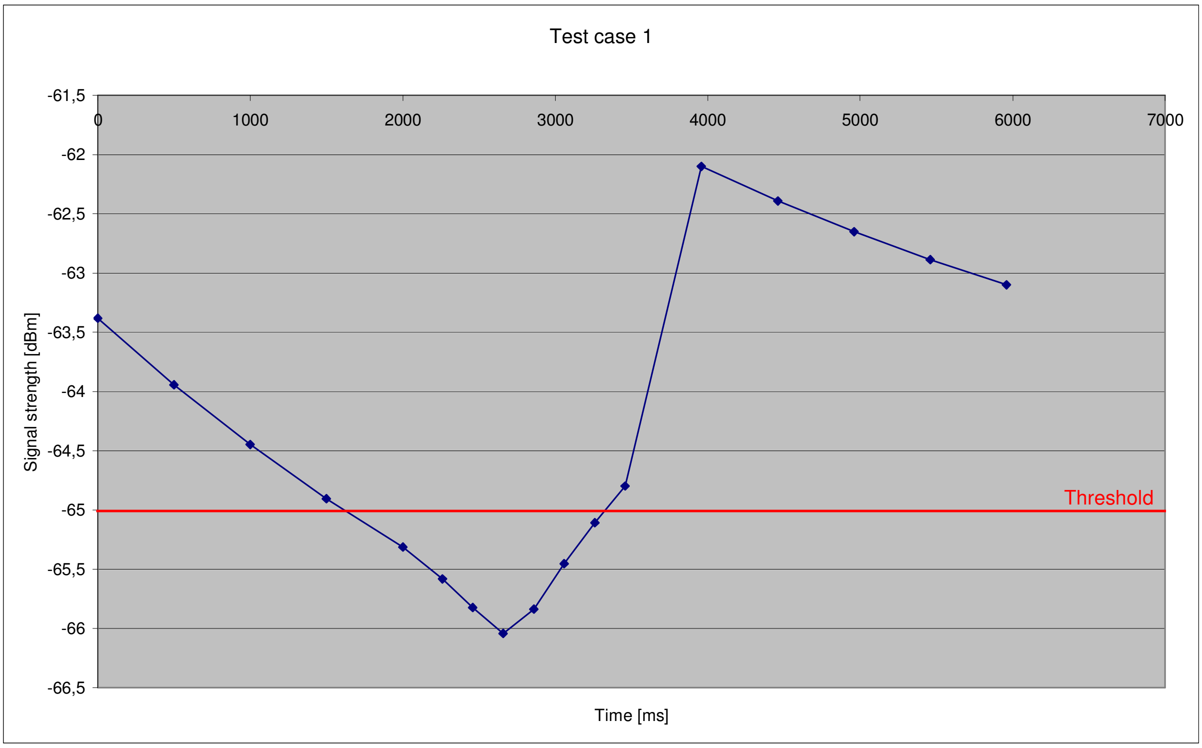


Figure 5-9: Test result of case 1

As we can see in the figure, the signal strength fell below the threshold -65 dB at 2000 ms, and then we checked it every 200 ms for 15 times. At 3459 ms the signal strength moved above the threshold again. Then a handover was avoided, and our algorithm began to check the signal strength every 500 ms. That’s what we supposed in our algorithm.

2. Signal strength drops down continually

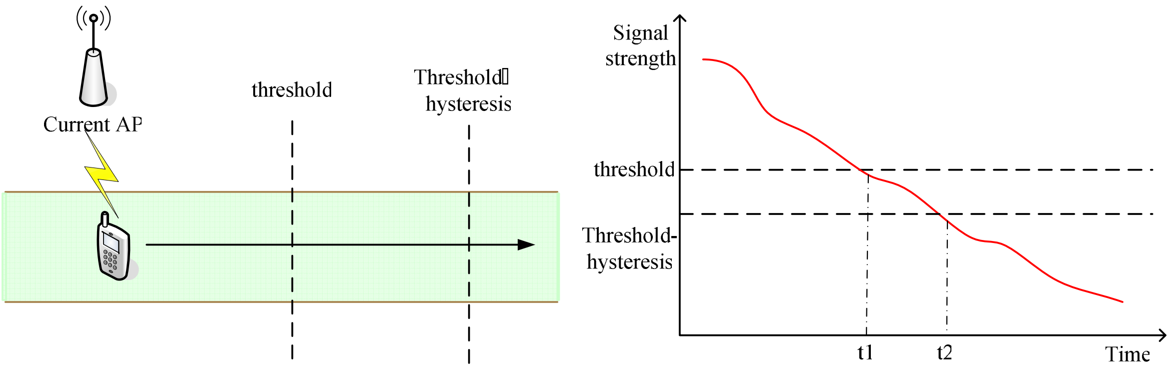


Figure 5-10: Signal strength drops down continually

As the figure shows, the client walks straight and comes further and further away from the current AP, and the signal strength falls down and down. The signal strength drops below the threshold at t1, continues to decrease, and even falls below the “threshold-hysteresis” at t2. Then a handover occurs immediately to avoid the connection break.

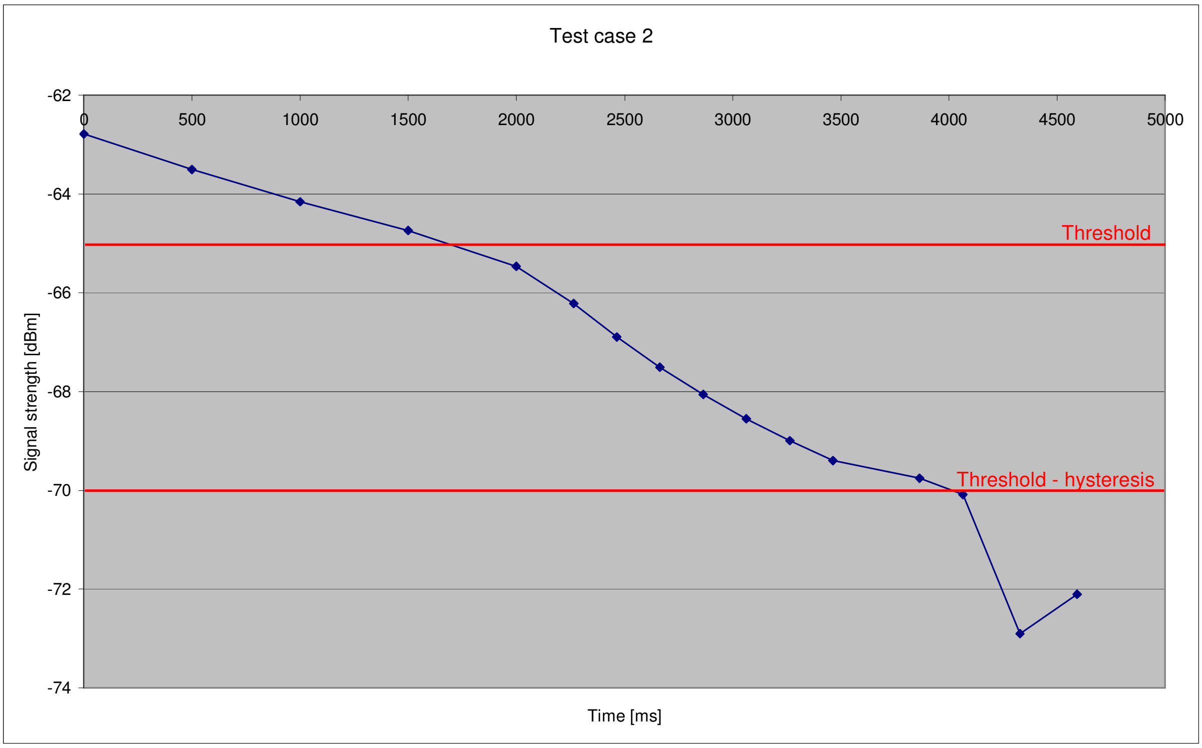


Figure 5-11: Test result of case 2

In this figure, the signal strength fell below the threshold at 2000 ms, and at 4064 ms, it even was below the “threshold-hysteresis”. Thus a handover should have done right there, and our algorithm did it.

3. Signal strength in challenging condition for more than 3 seconds

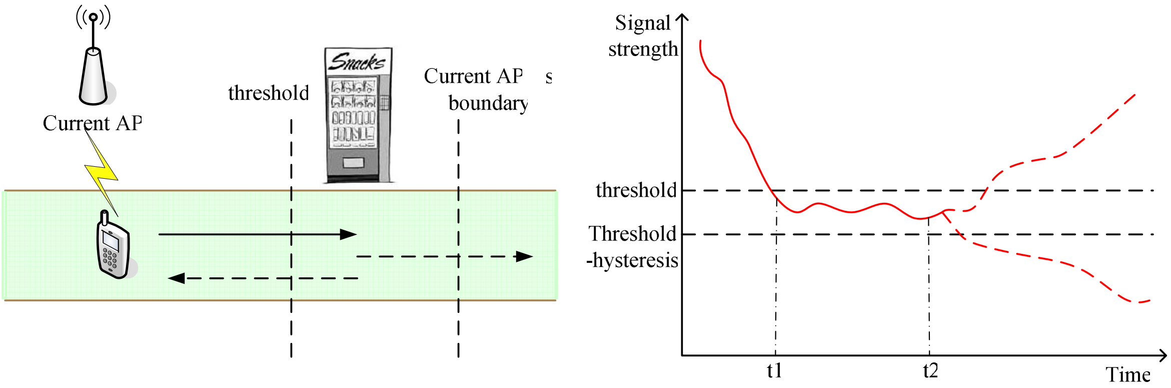


Figure 5-12: Signal strength keep challenging condition

We did some improvements based on the threshold scheme to avoid some unnecessary handovers. For example, a client with a WLAN support cellphone, walks along a corridor to a vending machine, stays there and buys some snacks or drinks, then turns back or continues walking in the original direction. Coincidently, the signal strength of the current AP at the vending machine is between the threshold and the "threshold-hysteresis". It means the connection quality is not good enough, and it should not keep a long time. Thus, if we found the client stays here for more than 3 seconds, we will trigger a handover.

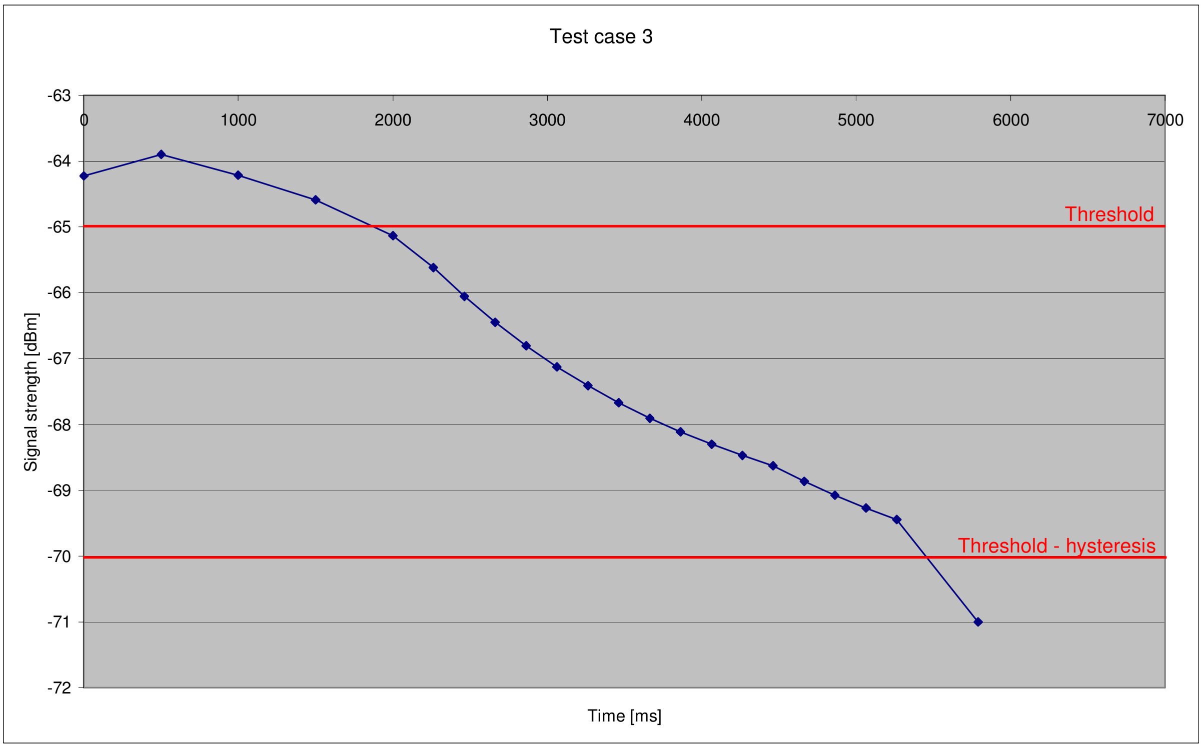


Figure 5-13: Test result of case 3

Our algorithm doesn't allow a long time poor connection. In this figure, the signal strength kept between -65 dB and -70 dB for more than 3 seconds, so a handover was triggered. Our algorithm worked as what we expected.

4. When type A APs are the best available APs

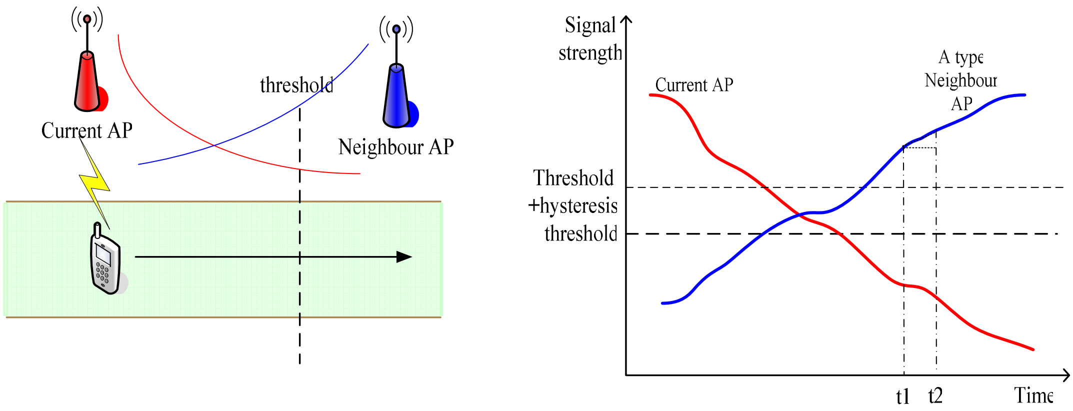


Figure 5-14: Type A APs are the best available APs

On the user's way, there are some neighbour APs which have very strong signal strengths, even higher than the "threshold + hysteresis". When the user gets closer and closer to these APs, their signal strengths become stronger and stronger. Our algorithm considers such kinds of APs as the best type of APs.

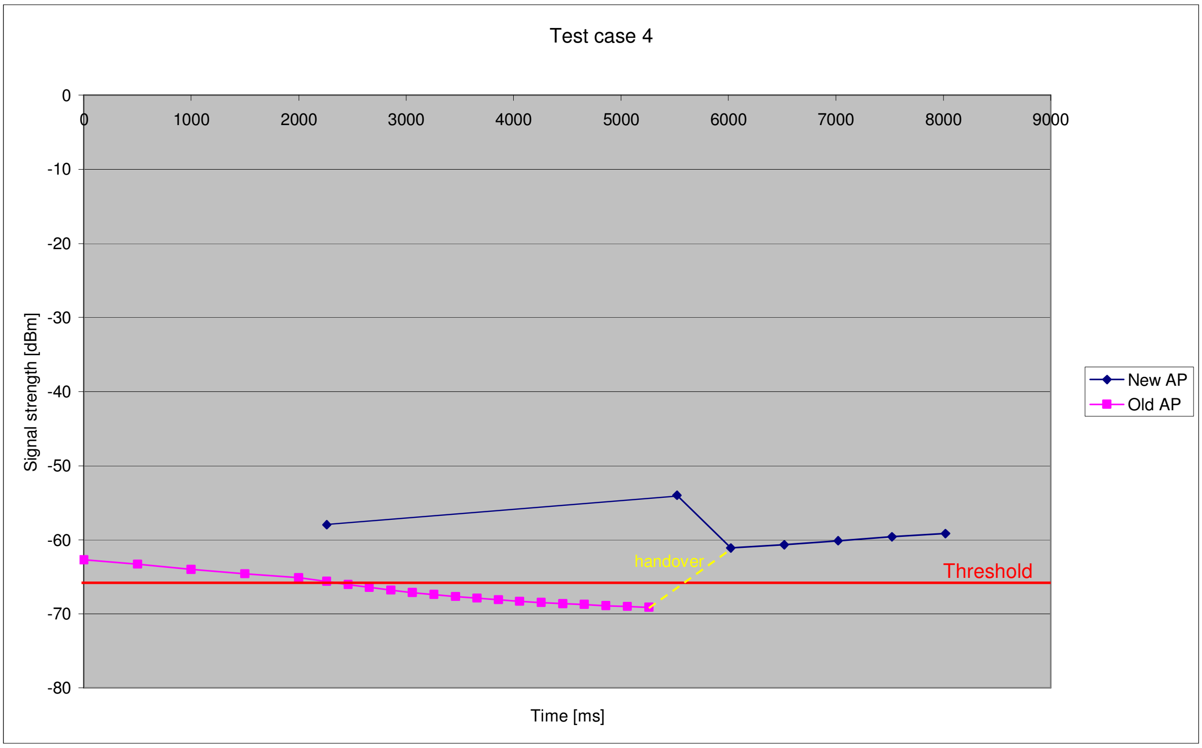


Figure 5-15: Test result of case 4

At 2000 ms, the current AP's signal strength fell below the threshold. The algorithm then took 260 ms and found a neighbour AP whose signal strength was -58 dB. And 3 seconds later, the current AP's signal strength was still below the threshold, the algorithm scanned again and found that neighbour AP had a signal strength -54 dB. The neighbour AP had a positive trend and a very good signal strength. Our algorithm considered it as A type AP and switched to it at 5521 ms.

5. When type C APs are the best available APs

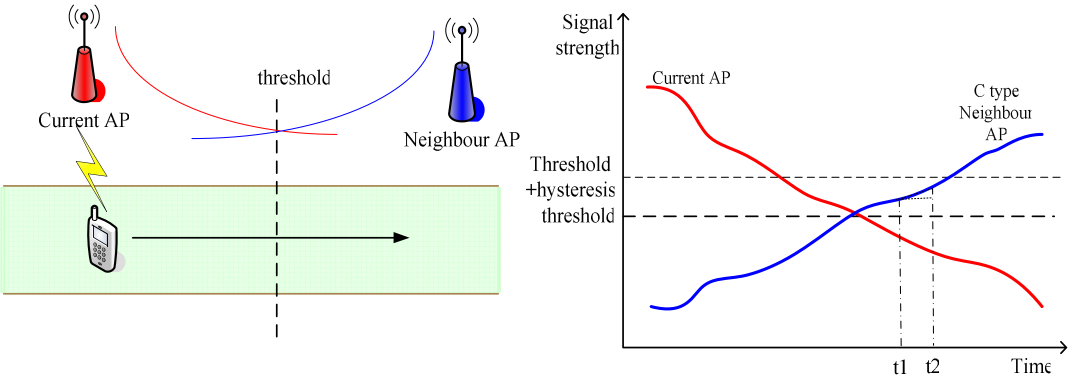


Figure 5-16: Type C APs are the best available APs

We don't offer A type neighbour APs this time. When the client walks from the current AP to some available neighbour APs, whose signal strength is below the "threshold + hysteresis" but higher than the threshold, those APs' signal strengths will increase too. If there exists no A type candidate AP, the APs of type C are the best choice.

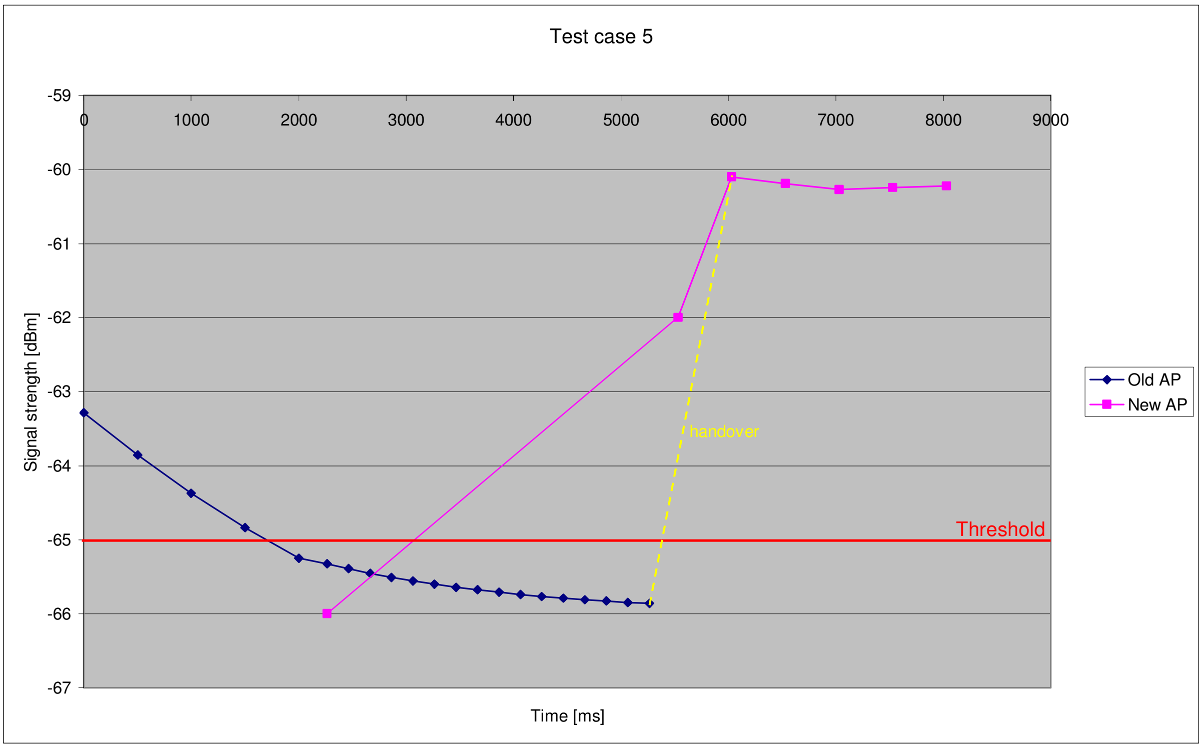


Figure 5-17: Test result of case 5

The current AP's signal strength fell below the threshold at 2000 ms, and 264 ms later our algorithm scanned and found a neighbour AP with signal strength -66 dB. 3 seconds later the current AP's signal strength still had a poor quality. Thus our algorithm measured the neighbour AP and found its signal strength was -62 dB. The candidate AP had a positive trend and a fine signal strength - it belongs to C type in our algorithm. And the client handed over to that AP at 5531 ms.

6. When type B APs are the best available APs

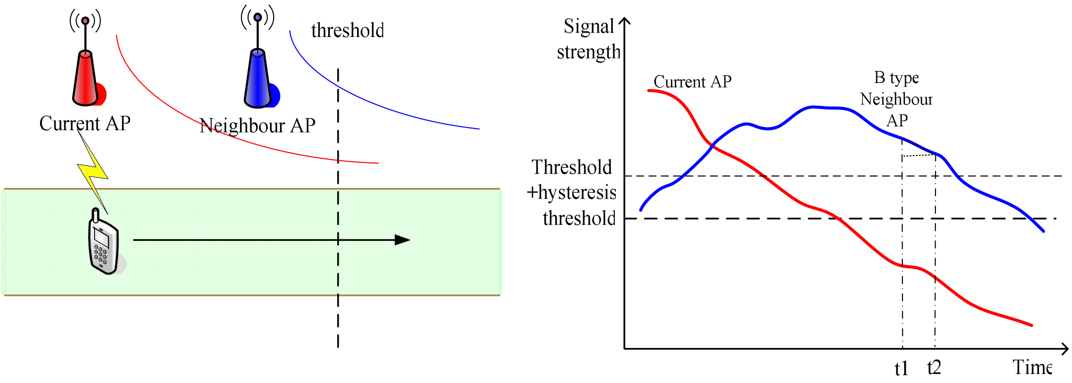


Figure 5-18: Type B APs are the best available APs

On the client's way, the best kind of AP is some AP near the current AP. However when we decide to do a handover and seek new APs, the client has just passed those APs' centre area a bit and walked to their boundaries. Then perhaps those APs still have strong signal strength, higher than the "threshold + hysteresis", but they have decreased trends. If there are neither A type APs nor C type APs, then the B type candidate APs are the best.

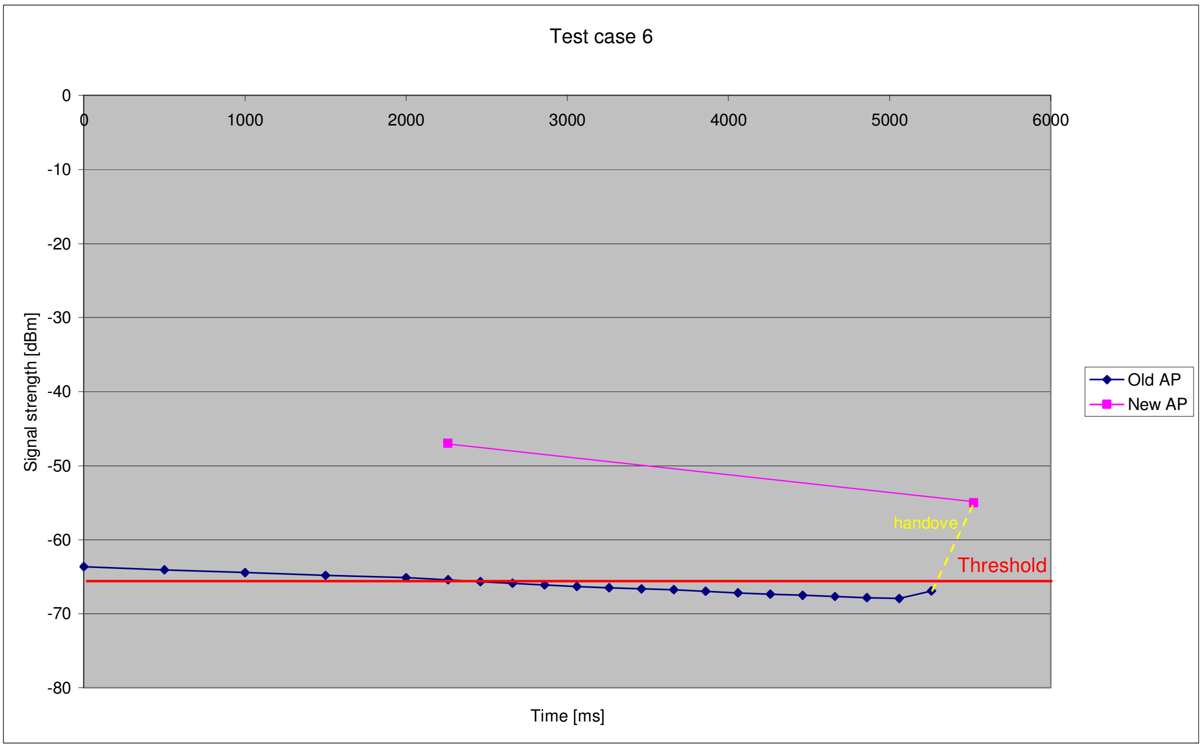


Figure 5-19: Test result of case 6

The pink line showed the neighbour AP which had a very good signal strength but a negative trend. That's the B type AP. The client switched from the current AP (showed by the blue line) to the neighbour one at 5523 ms as we wanted.

7. When type D APs are the best available APs

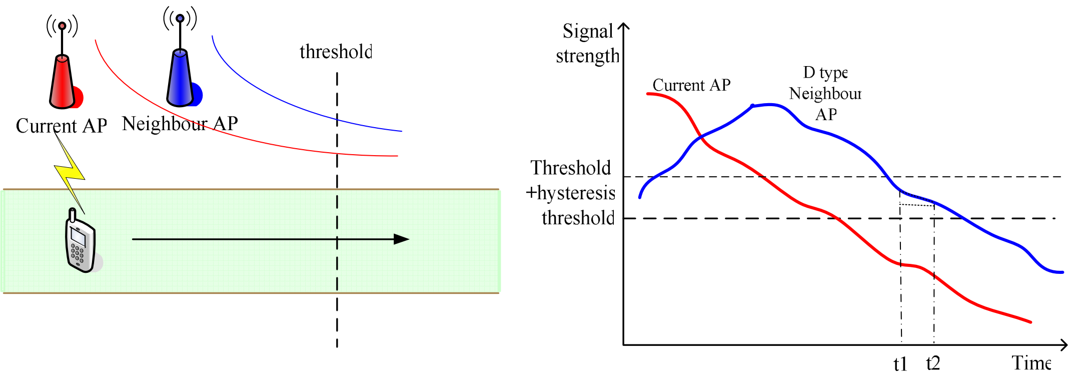


Figure 5-20: Type D APs are the best available APs

If there are only some neighbour APs which are very close to the current AP, and when we scan surrounding APs, the client has passed the centre of those APs a lot, and walked to their boundaries. However those APs' signal strengths are still better than the current AP's, actually ranging from the threshold to the "threshold + hysteresis".

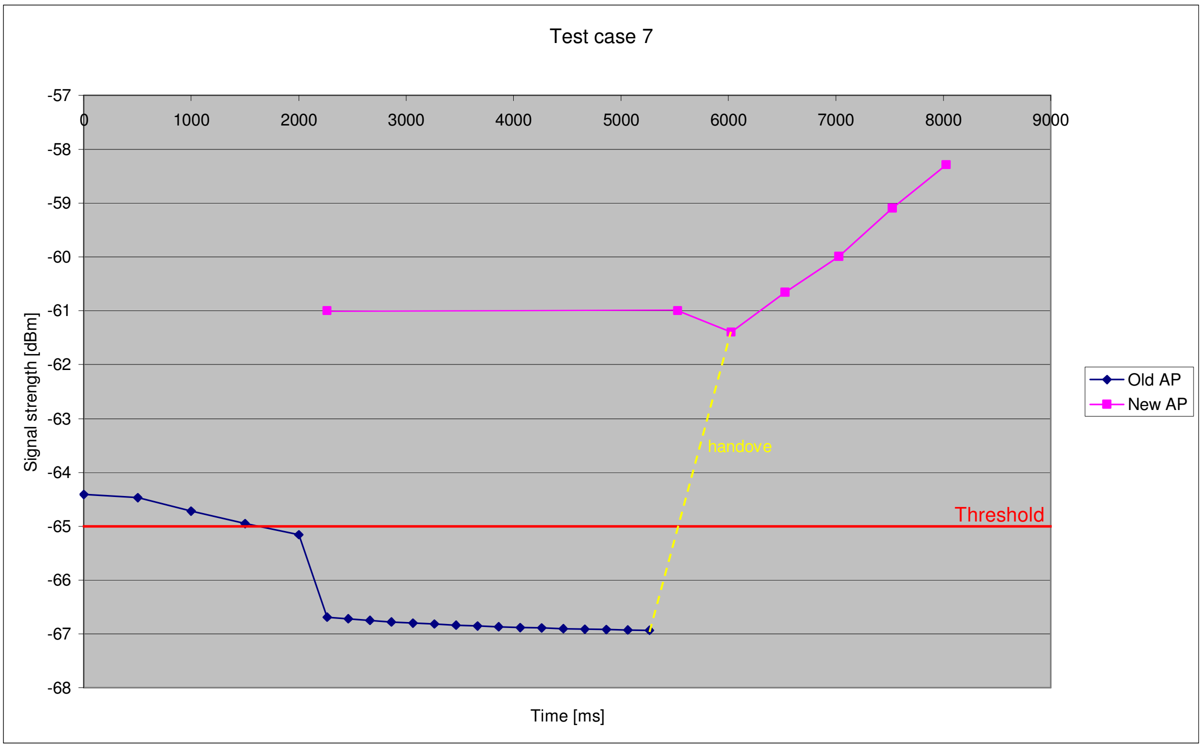


Figure 5-21: Test result of case 7

In this figure, the pink line had signal strength -61 dB at both 2263 ms and 5527 ms, so it had a nought trend. In our algorithm we consider this AP as D type. We found that even though the D type AP is not the most ideal candidate, it was still better than the current one. So at 5527 ms, the client switched to it.

8. No better AP available

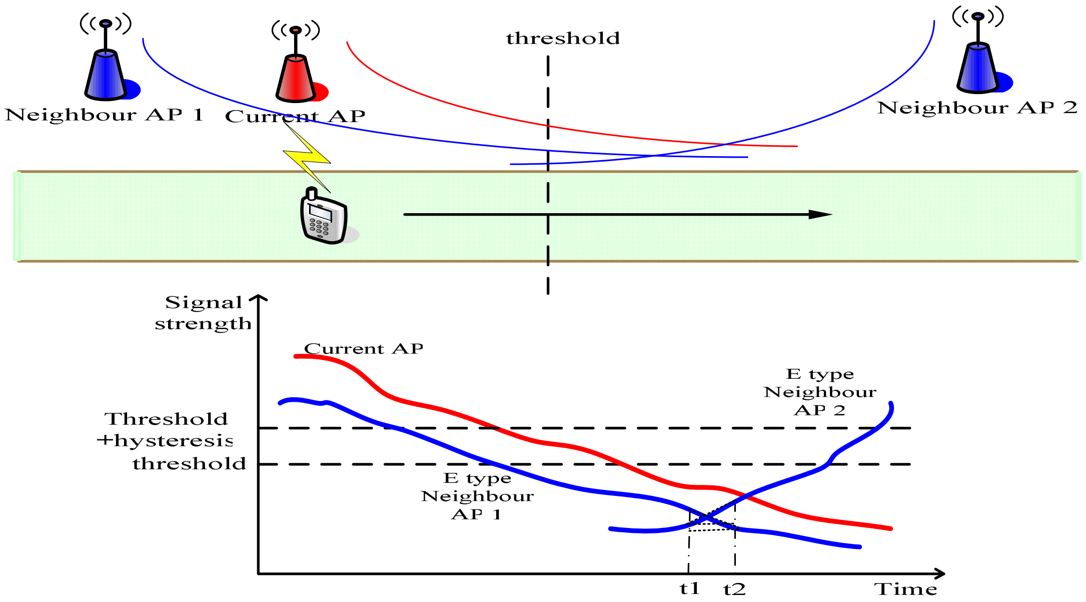


Figure 5-22: No better APs than the current one

The worst condition is when there are no other APs available besides the current one, or only some APs with very low signal strengths, lower than the threshold. Then no ideal neighbour APs can become the new AP, and a new circle of scanning is needed. We will do the testing in such conditions.

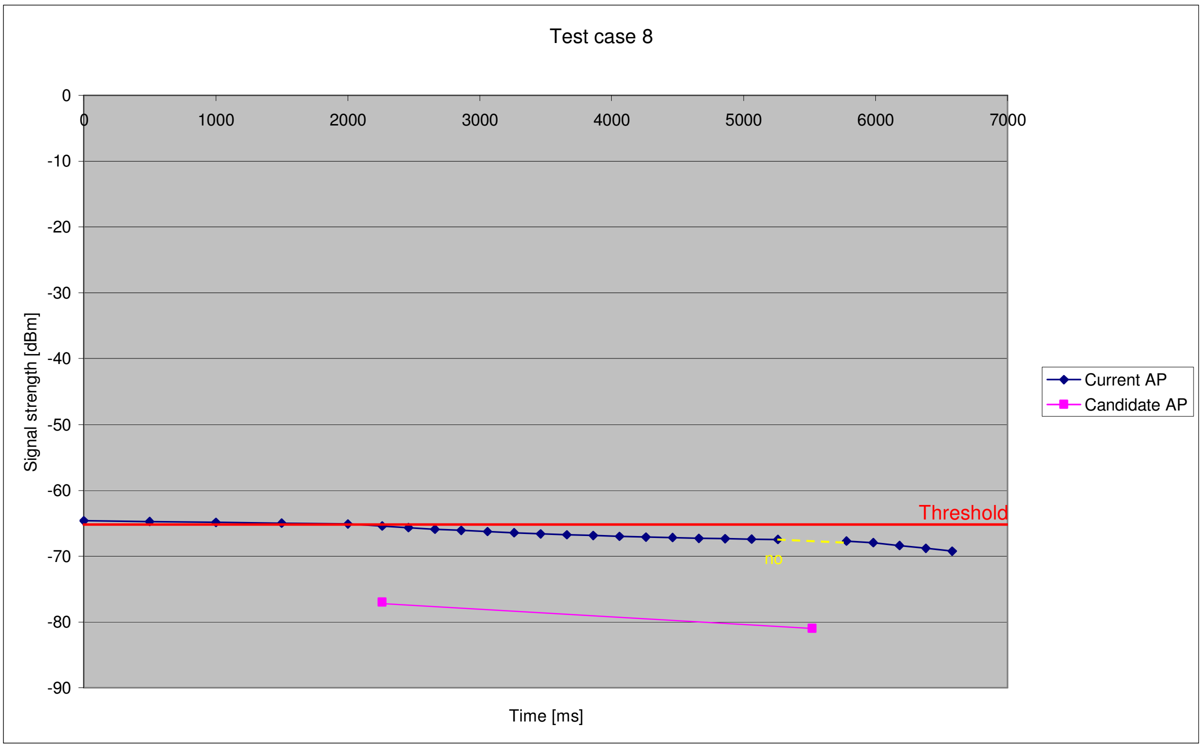


Figure 5-23: Test result of case 8

As we can see in the figure, the pink one had a worse signal strength than the current AP. Thus at 5260 ms, even though the current AP's signal strength was poor, the client didn't switch to the candidate one but keep the current connection.

5.5 Our handover algorithm VS. Windows built-in handover algorithm

In this test, we will walk twice with a PC to make some handover, using the windows built-in handover algorithm and our algorithm respectively. The windows built-in handover algorithm is the most used algorithm nowadays in our laptops. And we use another PC with MGEN to measure the number of lost packets during the handover processes. The algorithm with less lost packets is better.

Since we only did the testing once, we cannot declare that the result reflect the universal condition. We only humbly analyze the result of the condition in this testing.

In this testing, we got 2200 packets from 13:41:20 to 13:42:04 by our algorithm while 1860 packets from 13:58:33 to 13:59:10 by the win algorithm. We take the packets in complete seconds to calculate the average packets number per second, and we get 51.1667 with our algorithm while 51.0857 with the windows algorithm. This means the windows built-in handover algorithm lost more packets than our algorithm.

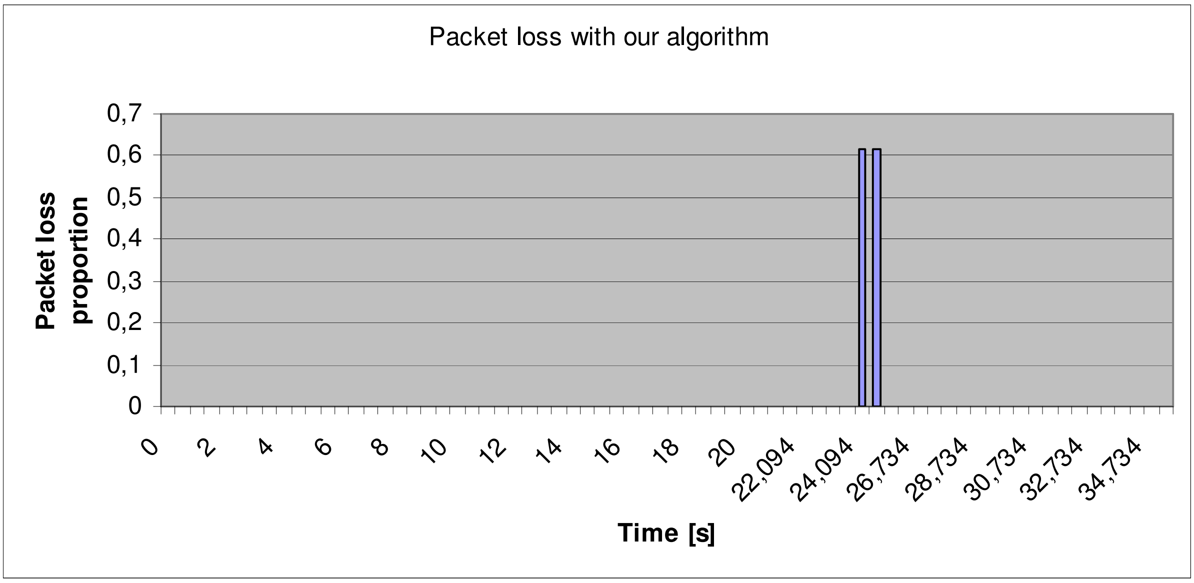


Figure 5-24: Packet loss with our algorithm

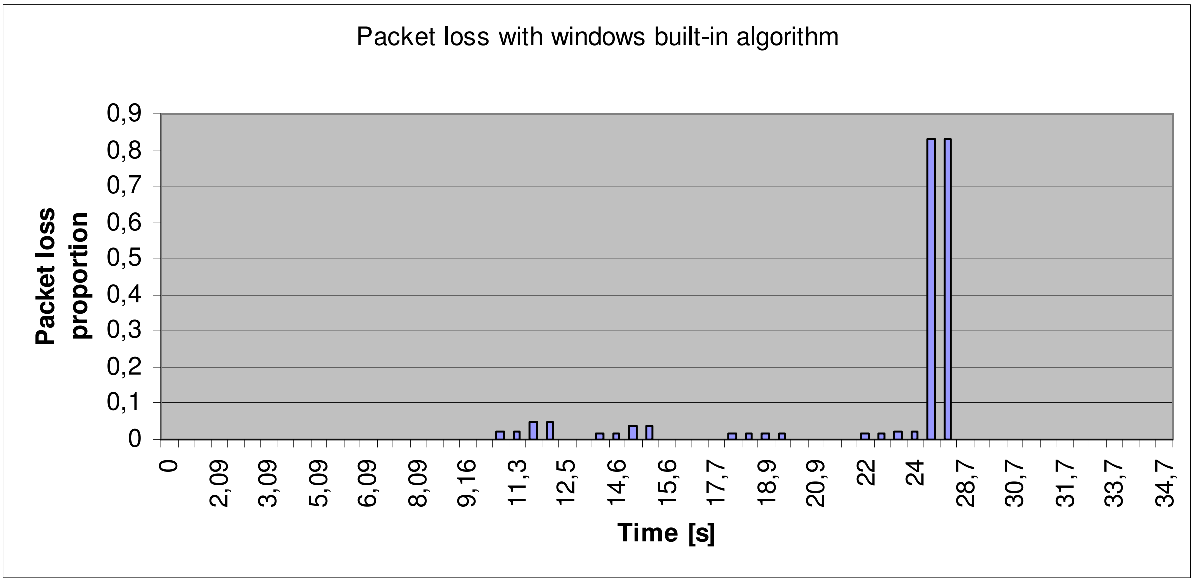


Figure 5-25: Packet loss with windows algorithm

Comparing these two figures, we can find that there are many packets lost before the handover with the windows built-in handover algorithm. That's because the window built-in handover algorithm triggers a handover when the current AP's connection quality is very poor, while our algorithm does a handover before the connection quality is too poor to loose packets. Since the packet loss influent the connection quality, so in this case, our algorithm offered better connection.

The number of loss packets increases rapidly when a client does a handover. With the two figures, we found that the handover time is shorter and the packets loss proportion is lower in the former figure than those in the later figure. Thus in this testing condition, our handover algorithm gave better performance when the client switch from the current AP to another one.

5.6 Conclusion

We tried to use the test cases to cover all the possibilities in our handover algorithm running process. According to the test results, we can declare that our program runs well and works as we defined in our algorithm exactly. With our algorithm, the client avoids mass of unnecessary handovers. And when choosing an AP from the neighbour APs as the new AP to switch to, our algorithm considers many more factors than only the signal strength parameter. Thus the client can have a stable, good performance AP as the new AP. Thereby many latent unnecessary handovers can be further avoided.

6 Discussion

A handover is a complex process and there exists many methods for detection, selection and execution. The methods we used and their parameters are discussed next.

Smoothening the signal strength is essential due to signal strength fluctuation. Exponential Weighted Moving Average Filter is the common method to smoothen signal strength and we also used this in our implementation and experiments. As presented in chapter 5.3, a smoothening factor can be chosen in many different ways. On the one side you want to smoothen very much to avoid unnecessary peaks but on the other side you want a value that is close to the real signal strength. This is an implementation decision, stability vs. accuracy. The threshold should be set in relation to the smoothening factor, with a $\beta = 0.9$ to smoothen the signal strength we found that a threshold value at -65 dBm works well.

The scanning method used in the ipw2200 driver we used for implementation scans all the channels in one process. Scanning all the channels took about 260 msec which is a too long break in the connection. Even though we never got the time to modify the driver to scan one channel at a time starting with the non overlapping channels, we are convinced this is the way to go in client based handover solutions. The best way to do scanning is of course with help from the network. As earlier explained the 802.11k standard uses this idea, but it is still just a draft and seems to be further delayed.

To classify the candidate APs we used a mix of the measured signal strength, trend and variance. As with the smoothening, the trend can be calculated with different parameters. To calculate the trend, you will need several signal strength measurements and a difference in them. To get a difference in the signal strength in the measurements you will have wait some time to be sure that the signal strength has the time to change. If you wait too long the user might have turned around, and if you wait too short you might not get a change in the signal strength. We used 3 seconds waiting time between the measurements which gave us good results. In addition to signal strength, trend and variance we think that considering the traffic load at the candidate APs are important, but we did not have the time to implement and test this in our project.

We have looked at several methods for enhancing the performance of a handover execution. The key to do a fast handover is to authenticate once and then distribute a Pair-wise master key in a secure way to the other access points in the network. Both the 802.11i proactive key caching and the 802.11r do this. A handover execution delay below 50 ms we believe can be reached in the future with 802.11r, but the total handover delay which also includes handover decision, scanning and selection will not get below 50 ms.

7 Conclusion and Future Work

7.1 Conclusion

Most of current handover algorithms disconnect the current AP, scan for APs, and then switch to the new AP. The main shortcoming is that the process takes too much time which makes the handover not seamless, even risking a broken connection.

With the study of several different kinds of current handover algorithms, we learned and absorbed their advantages into our algorithm.

Our algorithm uses an improved threshold scheme to detect the handover, which avoids many unnecessary handovers and prevents rapidly dropping signal strength or long time poor connection quality. And we do the scanning and APs selection before disconnecting the current AP. So the whole disconnect time is only the handover execution time, which is much shorter than the customary one's. Further more, we use signal strength, hysteresis and trends to classify the candidate APs, choose the best one out of them, and then switch to it. It ensures the client hand over to a good AP, which avoids the latent unnecessary handovers resulting from the yoyo effect. By testing we declare that our algorithm runs well in the real WLAN environments.

7.2 Future work

Because of the limited time, there are still many things in our algorithm that need to be improved.

First, our algorithm is signal strength based. All the parameters in our algorithm, such as threshold, trend, and variance, are signal strength related. Focus on the signal strength only makes the algorithm narrowly covered and less efficient. So in the future, we can import other quality parameters such as packet loss, traffic load and throughput, into our algorithm, as the handover detection and selection criteria.

Second, in our algorithm, we scan all the channels at one time. However what we want is to scan some selected channels one by one. In this way, a long scanning period can be divided into several short periods, which gives the client better connection quality. And since only some selected channels need to be scanned, time and power are saved.

Finally, in the project, our algorithm mainly focuses on the handover detection and selection part. We shorten the handover time by detecting, scanning and selecting a new AP before the disconnection with the current AP. However, the real handover execution time is not changed. If we have more time, we will study more about the handover execution part and shorten the execution time.

Glossary and Abbreviations

AAA	Authentication, Authorization, and Accounting
AP	Access point
AUC	Agder University College
BSS	Basic Service Set
CTS	Clear To Send
DS	Distributed System
EAP	Extensible Authentication Protocol
EAPOL	Extensible Authentication Protocol Over LANs
EDGE	Enhanced Data rates for Global Evolution, also known as EGPRS
EMP	Ericsson Mobile Platforms
ESS	Extended Service Set (consisting of several BSS)
EWMA	Exponential Weighted Moving Average
GPRS	General Packet Radio Service
GSM	Global System for Mobile communication
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IP	Internet Protocol
MAF	Moving Average Filter
MGEN	Multi-Generator
ms / msec	Milliseconds
PKC	Proactive Key Caching
PMK	Pairwise Master Key
PTK	Pairwise Temporal Key
QoS	Quality of Service
R0KH	Level 0 Key Holder
R1KH	Level 1 Key Holder
RFC	Request For Comments
RSSI	Received Signal Strength Indicator

RTS	Ready To Send
S/N	Signal over Noise ratio
SD	Security Domain
SIP	Session Initiation Protocol, VoIP signalling protocol
SMD	Security Mobility Domain
SSID	Service Set Identifier, the ID of a wireless network
STA	Station
TRPR	Tcpdump Rate Plot Real-time
TSPEC	Traffic Specification
VoIP	Voice over Internet Protocol
WLAN	Wireless Local Area Network
WMM	Wi-Fi Multimedia, QoS standard from Wi-Fi alliance
WPA/WPA2	Wi-Fi Protected Access, Security standards from Wi-Fi alliance

References

1. “IEEE Std. 802.11, “Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications,” 1999
Available at HTTP: http://standards.ieee.org/getieee802/download/802.11-1999.pdf

2. Institute of Electrical and Electronics official website
Available at HTTP: http://www.ieee.org

3. Internet Engineering Task Force official website
Available at HTTP: http://www.ietf.org

4. Wikipedia, “Session Initiation Protocol”, Visited February 2007
Available at HTTP: http://en.wikipedia.org/wiki/Session_Initiation_Protocol

5. Wikipedia, “Skype”, Visited February 2007
Available at HTTP: http://en.wikipedia.org/wiki/Skype

6. “Voice Over IP – Per Call Bandwidth Consumption”, Cisco Systems, 2005
Available at HTTP: http://www.cisco.com/warp/public/788/pkt-voice-general/bwidth_consume.pdf

7. Jardar Leira, “Throughput”, Uninett, April 15, 2005
Available at HTTP: http://forskningsnett.uninett.no/wlan/throughput.html

8. Tim Szigeti and Christina Hattingh, “Quality of Service Design Overview”, Cisco Press, December 17, 2004
Available at HTTP: http://www.ciscopress.com/articles/article.asp?p=357102&rl=1

9. Ericsson Mobile Platforms – Company presentation in course IKT501 Mobile Communication Networks at HiA, Okt 25, 2006

10. Pejman Roshan and Jonathan Leary, “802.11 Wireless LAN Fundamentals”, 1st Edition, Cisco Press, Dec 23, 2003

11. Vivek Mhatre and Konstantina Papagiannaki, ”Using Smart Triggers for Improved User Performance in 802.11 Wireless Networks”, MobiSys'06, June 2006, Uppsala, Sweden
Available at HTTP:
http://portal.acm.org/citation.cfm?id=1134706&dl=ACM&coll=&CFID=15151515&CFTOKEN=6184618

12. H. S. Kim, S. H. Park, C. S. Park, J. W. Kim, and S. J. Ko, “Select Channel Scanning for Fast Handoff in Wireless LAN using Neighbor Graph”, in International Technical Conference on Circuits Systems, Computers and Communications, Sendai/Matsusima, July 2004.
Available at HTTP: http://dali.korea.ac.kr/publication/int_pro/paper/IntPro085.pdf

13. Ishwar Ramani and Stefan Savage, “SyncScan: Practical Fast Handoff for 802.11 Infrastructure Networks”, Department of Computer Science & Engineering, University of California, San Diego
Available at HTTP: http://www.cs.ucsd.edu/~savage/papers/Infocom05.pdf

14. Wikipedia, “802.11k”, Visited February 2007
Available at HTTP: http://en.wikipedia.org/wiki/IEEE_802.11k

15. Dan Simone, “802.11k makes WLANs measure up”, Network World, March 29, 2004
Available at HTTP: http://www.networkworld.com/news/tech/2004/0329techupdate.html

16. “Dealing with measurement noise—a gentle introduction to noise filtering”, Copyright M.T. Tham (1996-1998)
Available at HTTP: http://lorien.ncl.ac.uk/ming/filter/filewma.htm

17. “Smoothing of data”, Laboratory online Computing, 1975
Available at HTTP: http://www.numberwatch.co.uk/smoothing_of_data.htm

18. H. Velayos and G. Karlsson, “Techniques to Reduce IEEE 802.11b MAC Layer Handover Time”, Tech. Rep. TRITA-IMIT-LCN R 03:02, Apr. 2003
Available at HTTP: http://web.it.kth.se/~hvelayos/papers/TRITA-IMIT-LCN%20R%2003-02%20Handover%20in%20IEEE%20802.pdf

19. P. Marichamy, S. Chakrabarti and S. L. Maskara, “Performance Evaluation of Handoff Detection Schemes”, 2003, India
Available at HTTP: http://ieeexplore.ieee.org/iel5/8975/28486/01273250.pdf

19. Anthony Noerpel and Yi-Bing Lin, “Handover Management for a PCS Network”, IEEE personal communications, December 1997
Available at HTTP: http://ieeexplore.ieee.org/iel4/98/13833/00637379.pdf?arnumber=637379

20. G. P. Pollini, “Trends in handover design”, IEEE Communications, vol. 34, pp. 82{90, Mar. 1996
Available at HTTP: http://ieeexplore.ieee.org/iel1/35/10421/00486807.pdf?arnumber=486807

21. V. Kapoor, G. Edwards, and R. Sankar, "Handoff Criteria for Personal Communication Networks", Proc. ICC '94, New Orleans, LA, May 1-5, 1994, pp. 1297-1 301
Available at HTTP:
http://ieeexplore.ieee.org/iel2/2977/8446/00368895.pdf?arnumber=368895

22. A. Balachandran, P. Bahl, and G. Voelker, “Hot-spot congestion relief and service guarantees in public-area wireless networks,” SIGCOMM Computer Communication Review, vol. 32, no. 1, 2002

23. Hector Velayos, Victor Aleo and Gunnar Karlsson, “Load Balancing in Overlapping Wireless LAN Cells”, KTH, Royal Institute of Technology, Sweden, 2004

24. IEEE 802.11kTM/D5.0 Draft Amendment to STANDARD FOR Information Technology – Telecommunications and Information Exchange Between Systems – Local and Metropolitan Area Networks – Specific Requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment 1: Radio Resource Measurement, August 2006

25. Cisco Systems official website
Available at HTTP: http://www.cisco.com

26. Louise McKeag, "WLAN Roaming – the basics", Techworld March 2004
Available at HTTP: http://www.techworld.com/mobility/features/index.cfm?FeatureID=435

27. Symbol Technologies official website
Available at HTTP: http://www.symbol.com

28. Bluesocket official website
Available at HTTP: http://www.bluesocket.com

29. Proxim Wireless official website
Available at HTTP: http://www.proxim.com

30. Wikipedia, "Inter-Access Point Protocol", Visited February 2007
Available at HTTP: http://en.wikipedia.org/wiki/Inter-Access_Point_Protocol

31. Wikipedia, "IEEE 802.1X", Visited February 2007
Available at HTTP: http://en.wikipedia.org/wiki/802.1X

32. "Implementation of IAPP with Proactive Caching", Visited February 2007
Available at HTTP: http://www.cs.umd.edu/~mhshin/iapp/

33. Benjamin Miller, "Is it the network? Solving VoIP Problems on a Wireless LAN", Global Knowledge, 2007
Available at HTTP:
http://images.globalknowledge.com/wwwimages/whitepaperpdf/WP_Miller_VoIP_LAN.pdf

34. Sangeetha Bangolae, Carol Bell and Emily Qi, "Performance Study of Fast BSS Transition using IEEE 802.11r", IWCMC'06, July 3–6, 2006, Vancouver, British Columbia, Canada, Copyright 2006 ACM 1-59593-306-9/06/0007
Available at HTTP:
http://portal.acm.org/citation.cfm?id=1143696&dl=acm&coll=&CFID=15151515&CFTOKEN=6184618

35. IEEE 802.11r^{TM}/D2.1 Draft Amendment to STANDARD FOR Information Technology – Telecommunications and Information Exchange Between Systems – Local and Metropolitan Area Networks – Specific Requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment 2: Fast BSS Transition, May 2006

36. K. Ramachandran, S. Rangarajan & J. C. Lin, "Make-Before-Break MAC Layer Handoff in 802.11 Wireless Networks", June 2006
Available at HTTP: http://www.winlab.rutgers.edu/~kishore/papers/icc-2006.pdf

37. Wikipedia, "Quality of Service", Visited April 2007
Available at HTTP: http://en.wikipedia.org/wiki/Quality_of_service

38. Wikipedia, “Wireless Multimedia Extensions”, Visited April 2007
Available at HTTP: http://en.wikipedia.org/wiki/WMM

39. Tim Szigeti and Chritina Hattingh, “End-to-end QoS Network Design: Quality of Service in LANs, WANs and VPNs”, Cisco Press, Nov 2004

40. Madwifi official website
Available at HTTP: http://madwifi.org

41. Ubuntu official website
Available at HTTP: http://www.ubuntu.com

42. IPW2200 official website
Available at HTTP: http://ipw2200.sourceforge.com

43. Java official website
Available at HTTP: http://www.sun.com/java

44. Eclipse official website
Available at HTTP: http://www.eclipse.org

45. Omondo official website
Available at HTTP: http://www.eclipsedownload.com

46. Naval Research Laboratory, Networks and Communication System Branch official website
Available at HTTP: http://cs.itd.nrl.navy.mil