NTNU – Trondheim
Norwegian University of
Science and Technology

An investigation of large scale challenges with live video streaming over Wi-Fi access networks

Sepideh Kanani

Submission date: May 2014
Responsible professor: Øivind Kure , ITEM
Supervisor: Otto J Wittner, Uninett

Norwegian University of Science and Technology
Department of Telematics

Abstract

Due to growing demand for wireless access technology (802.11 standards), high capacity multimedia transmission over wireless pose a challenge, by taking note that video streaming has gained enormous popularity and is accountable for a large fraction of current internet traffic.In order to have a satisfy video transmission wireless network especially when it comes to large scale user and roaming intensity, many requirement must be considered.

Large throughput and minimum delay are the significant requirements to keep the transaction stable and seamless, moreover jitter is another important factor that can result in degradation and quality of received video. To achieve seamless live video stream and high level of quality of service (QoS) and quality of experience (QoE) different technologies must be studied, such as video encoding/decoding platform, transmission technology (unicast/multicast) mobility models and efficient handoff method. These issues are addressed in this project.

During this work, several experiments are done in NS2 simulator to study the impact of increasing in number of users on the QoS parameters such as delay, throughput and jitter, while an H.264 SVC video stream id transmit over the wireless network in both multicast and unicast manner. Also, the effect of Random Waypoint and Gauss-Markov mobility model on the delivering performance studied. The experiments tested during movement inside an AP coverage area as well as movement to new AP domain (Handoff). Hierarchical Mobile IPv6 is the protocol used in the simulation, in order to have a faster handoff and reduction in the amount of signaling load.

Preface

This report serves as master thesis in Telematics specialized in Networks and quality of service at The Norwegian University of Science and Technology, NTNU. The assignment is given by Uninett AS.
I would like to express my deepest appreciation to my supervisor, Otto J Wittner for his great help, valuable support and weekly feedback.
I also wish to express my gratitude to Professor Øivind Kure for ideas at times where the direction of my thesis was unclear.

Trondheim, March 24, 2014
Sepideh Kanani

Contents

List of Figures

Acronyms

AP Access Point

AP Access Point

AVC Advanced Video Coding

BS Base Station

BSS Basic Service Set

CBT Core-Based Trees

CSMA Carrier Sense Multiple Access with Collision Avoidance

DCF Distributed Coordination Function

DVMRP Distance Vector Multicast Routing Protocol

EDCA Enhanced Distributed Channel Access

ESS Extended Service Set

FDDI Fiber Distributed Data Interface

FHR Frequency Handoff Region

IAPP Inter Access Point Protocol

IEEE Institute of Electrical and Electronics Engineers

IEC International Electrotechnical Commission

IETF Internet Engineering Task Force

IGMP Internet Group Multicast Protocol

ISO (International Organisation for Standardisation

ISP Internet Service Provider

ITU International Telecommunication Union

LAN Local Area Network

MIMO Multiple-Input Multiple-Output

MN Mobile Node

MOS Mean Opinion Score

MOSPF Multicast extensions to OSPF

NAL Network Abstraction Layer

NTNU Norwegian University of Science and Technology

OEFMON Open Evaluation Framework Multimedia Over Network

OSPF Open Shortest Path First

PIM Protocol-Independent Multicast

PNC Proactive Neighbor Cashing

PSNR Peak Signal to Noise Ratio

QoE Quality of Experience

QoS Quality of Service

RSS Received Signal Strength

RSVP ReSerVation Protocol

RTP Real-Time Transport Protocol

SNC Selective Neighbor Cashing

SNR Signal to Noise Ratio

SSID Service Set Identifier

TCP Transmission Control Protocol

UDP User Datagram Protocol

VoIP Voice Over Internet Protocol

WEP Wired Equivalent Privacy

WiFi Wireless Fidelity

WLAN Wireless Local Area Network

WMN Wireless Mesh Network

RDSTC Randomized Distributed Space Time Codes

MOCTS Multicast Orthogonal CTS

FEC Forward Error Correction

Chapter 1
Introduction

Currently, wireless access network technology is increasing rapidly, moreover mobile computing devices such as laptops, tablets and smart phones are growing in both quality and quantity. While video transmission is the main traffic of wired network, by developing wireless network more attention gives to video delivering over wireless networks. It is predicted that within next few years the video applications such as media streaming, interactive networked 3D games and video conferencing will be the killer application for wireless network, according to Cisco visual index white-paper mobile video traffic will increase to 65 percent by 2017 up from 53 percent In 2012.

1.1 video streaming over wireless network

Due to wireless medium nature, wireless communication is usually a broadcast transmission of data, which in the data is sent by a common source and all nodes which use the same access network as well as same modulation and coding scheme (known as MCS) are able to receive the data. However either broadcast or unicast transmission (with a large number of receivers) lead to increased bandwidth consumption and therefore, are not efficient methods. Here multicast techniques come up; a point-to-multipoint transmission, since, in multicast communication, the data is only sent to multicast group members. The other advantages of wireless multicast are the number of wireless technologies which support point-to-multipoint transmission, such as WiMAX with its the Multicast Broadcast Services (MBS), The Third Generation Partnership Project (3GPP) with using Multimedia Broadcast/Multicast Service (MBMS). IEEE 802.11 standard also use unreliable multicast service while in the IEEE 802.11aa the reliable multicast over WLAN is implemented.
Despite all these benefits, wireless multicast still suffering from some issues. Gathering the receivers' feedback is one. Since the current wireless multicast methods are not allowed to transmit the link layer feedback, can not guarantee a reliable data transaction.

1.2 Motivation

There are several reasons behind this project motivation; The first is widespread of wirelessly enabled devices. Already today practically all end users access the internet via some wireless access network, one common type being the 802.11 Wi-Fi network. Video streaming has gained enormous popularity and is accountable for a large fraction of current internet traffic. Live video streaming over Wi-Fi access networks on a small scale is very well achievable. However, when things scale up, e.g. video quality, the number of user and the roaming intensity, live video over wireless access can become a challenge.

1.3 Objective

1.3.1 Main objective

The main objective in this thesis is to investigate on the quality of service (QoS) level of live streaming through Wi-Fi networks. QoS especially arises when users are charged for the service, and hence expect video streams delivering without delay, damaged signal or explicit noise.
Many different parameters play roles in order to maintain the video streaming stable and seamless. Packet loos ratio, jitters and Signal-to-noise ratio (SNR)are such examples. Finding the most proper parameters to achieve seamless transmission is of interest.

1.3.2 Research questions

Based on main objective, three research question is identified:

- How the user mass can effect on the real-time data transmission? In which extend the number of users can increase without degrading network quality of service? Especially during handoff, does it affect handoff delay and performance?

- How user behavior and mobility models can change QoS features?

- And finally How different handoff methods may lead to different performance?

1.4 Research method

This thesis follows the research method introduced and described in [29]. In order to find the answer for research questions, the following task is done:

- Study the required background on video multicasting, video encoding platforms and IEEE 802.11 standards.
- Review the previous works related to the thesis objective.
- Design different scenarios to meet the research questions.
- Gathering simulation results and theoretical information to describe and discuss the issues.

1.5 Outline

This report is organized as follow:
In chapter 2 an overview of related background is given, includes brief description of main objectives: video multicast technologies and methods, a short review of most important encoding platforms, an overview on IEEE 802.11 architecture and standards and finally different mobility models are discussed.
Chapter 3 includes the introduction to previous works and state of the art mechanisms to overcome the existing issues. It is good to note that, after pre study and reading survey reports on the technology area, It was realized that the technology area is more mature than first expected, and much relevant work has already been done. Therefore although it was tried to explain most relevant study briefly, but since the project includes wide-spreading technologies and area (encoding video, mobility models, wireless multicast, handoff methods) still background and related work took many pages of the report.
In chapter 4 system design and simulation scenarios are described.
The simulation environment, tools and implementation are covered in chapter 5.
Chapter 6 is continued by the experimental results, discussion and analysis the results,
And eventually chapter 7 is dedicated to thesis conclusion and possible future works.

Chapter 2
Background

The background chapter is dedicated to the introduction of the basis and concepts behind the mechanisms are used during this project and within next chapters. The chapter starts with background on video multicasting (2.1.1), which is the main idea of this work. It follows by brief description of video encoding/decoding platforms (2.2). IEEE 802.11 standard that this project is defined based on it, is the next section (2.3). Background chapter ends with a short introducing on mobility models that are used in the simulation scenarios (**??**).

2.1 Video multicast

Since the first time that IP-multicast was proposed in Steve Deering's ph.D. thesis in 1988 and later tested during an audio cast at 1992, so far multicast technology developed into an important and paramount internet service. By advent of video, mobility and cloud technologies, it has turned into a greatly desired feature of many vendors' network products and now is highly used by the companies that offer large-scale internet services and application. According to definition IP multicasting is "the transmission of an IP datagram to a *"host group", a set of zero or more hosts identified by a single IP destination"*[RFC 1112]. Today,the main concern of ISPs and telecom industry is a traffic load and bandwidth usage. At the same time, use of bandwidth-intensive applications is increased. The most popular application benefits from IP multicasting. Multimedia application such as video streaming, IPTV and video on demand, training and corporate applications such as e Learning application, video and audio conferencing, VoIP, data warehousing and any applications push data in one to many manner are such examples. Currently, during evening hours 50 percent of all Internet traffics belong to video streams in wired networks.[26]
Multicast optimizes performance and enhances efficiency by eliminating redundant traffic and controlling network traffic and consequently reduces server load and CPU consumption. It also makes using of distributed application possible. The 2.1 illustrates the traffic load for same audio stream in both multicast and unicast order.

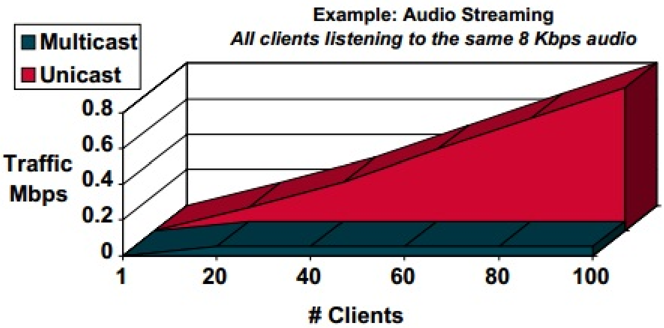

Figure 2.1: Multicast VS Unicast traffic from cisco.com

Multicast is a UDP based protocol, and it suffers from UDP's intrinsic weaknesses. Such as::

- Reliability issues: multicast follows Best Effort Delivery mechanism; hence packet dropping is expected, and it is not able to deliver data reliably.
- Congestion: due to lack of TCP windowing, there is no avoidance mechanism; multicast application should benefits from congestion detection condition.
- Duplication: it caused by some multicast protocols mechanism such as SPT; therefore, the application should design appropriately to deal with duplicated packets.
- Out of order delivery: the multicast packet may be delivered different from the order that they were sent.

However another major problem with multicast is transmission of same data to the different receivers in a heterogeneous group which requires different rates.

2.1.1 Multicast technology

Multicast transmission is done at both layer 2 (data link layer) and layer 3 (network layer) depends on network technologies. If the multicast is transmitted over a single LAN, it will carry out at link layer, While, across autonomous systems, multicasting is implemented at network layer. Multicast in layer 3 carries out through following process: [27]

- Addressing: A multicast address belongs to a group of users (multicast group) and not every single receiver. Class D(1110) is used for IP multicast addressing.

An ethernet address is allocated as the destination address, the 32 low bits of class D indicate to these ethernet addresses.

– Dynamic Registration:is a mechanism to inform the network about new multicast group's member. The internet group multicast protocol(IGMP)is responsible for updates the multicast list between host nodes and routers. A host can join a group by sending IGMP join request or "report". The router periodically sends a "query" to nodes to keep the multicast list update, if the router has received no report,after sending the query to nodes, the node will be pruned from the group list.In IGMP v2, a host sends a leaving message leaving the multicast group, by this way the wasted transmission time is minimized.

– Multicast Forwarding: as mentioned already, multicast transmission is based on UDP packets and compare with TCP has more packet lost, since UDPs follow the "best effort delivering" and also can not benefit avoidance mechanisms (such is available in TCP). therefore the quality degradation in multicast is more than unicast, especially in real time application which re-transmission request is impossible, in order to overcome UDP quality of service's drawbacks, ReSerVation Protocol (RSVP), the Real-Time Transport Protocol (RTP) and 802.1p are used.

– Multicast Routing: multicast routing refers to the distribution tree rooted at multicast source subnet, and its branches are ended at the subnet of multicast groups members. A spanning tree rooted forwards a copy of multicast packets on each branch and makes the connectivity possible.The IETF proposed several protocols to accomplish multicast routing, such protocols are Multicast extensions to OSPF (MOSPF),the Distance Vector Multicast Routing Protocol (DVMRP), Protocol-Independent Multicast (PIM), and Core-Based Trees (CBT). These protocols examine unicast routing table in order to find distribution tree. For example, MOSPF uses its link state data, and DVMRP uses its distance vector routing protocol while PIM and CBT use unicast forwarding table in order to implement the distribution tree.

2.1.2 Multicast distribution trees

Distribution trees are categorized in two basic groups: the source tree and shared tree(Figure 2.2)

Source tree: in the source tree or shortest path tree (SPT), source forms the root and branches are forwarding path to receivers. A source tree with source IP address S and multicast group address G are denoted as (S,G). Forwarding is based on the shortest path algorithm; therefore, this tree have the minimum network latency, but this optimization costs more resource and memory consumption since each router

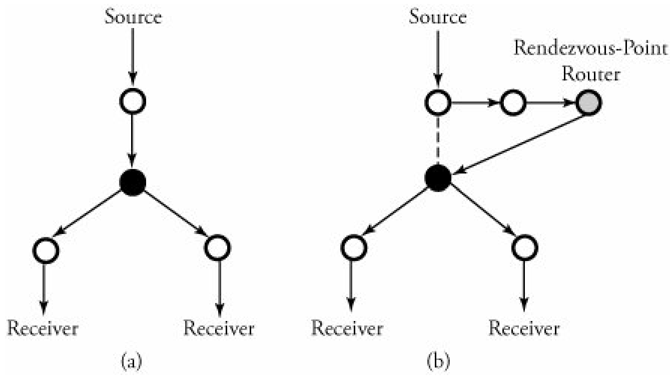

Figure 2.2: Source tree(a) VS Shared tree(b)[37]

must keep path information.
Shared tree: uses a rendezvous point(RP) as a source, the RP can be located at any chosen point in the network.The shared tree is unidirectional; traffic is sent to RP using s source tree then is forwarded down from RP to all receivers. A (*,G) notation indicates a shared tree which in, * represents all sources and G points out to the multicast group. However shared tree requires lower memory (compare with the source tree) but the built path might not be optimized.

2.1.3 Multicast routing protocols

Multicast protocols are belonged to two main categories: Dense-mode(DM) and Sparse-mode (SM), each of which follows different assumption.
In dense-mode It is assumed that all routers need to receive multicast traffic to forward to all multicast group, in fact, the assumption is that almost all nodes in the networks belongs to a multicast group. Therefore, the protocol initially flood the whole network and due to received report makes the distribution tree. DM protocols such as DVMRP, MOSPF and PIM-DM are well suited for LAN when there are dense receivers and also enough bandwidth is available.
In sparse-mode, the protocols, assume there are only few routers which should forward the multicast data and the group member are dispersed. SM protocols such as CBT and PIM-SM are appropriate for internet and WLAN.

2.2 Video encoding/decoding platforms

Video encoding is the technology of compressing video streams at source generator, store and transmit them by lower capacity and bandwidth to receivers. Using this technology is necessary for applications such as mobile TV, video conferencing and Internet video streaming. The standardization of video encoding is essential since

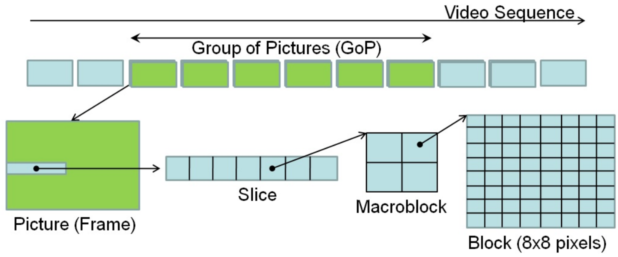

Figure 2.3: What is video? [63]

makes different manufacturing products able to coordinate and cooperate. In this section protocols, standards, structure of streams and their architecture will be briefly discuss. Short describes of three most recent and pervasive encoding standards are present at the next parts.

2.2.1 Video streaming architecture and protocols

As figure 2.3 Shows, a video stream is nothing but the groups of consecutive pictures. Each picture refers to as "frame", every frame consists of numbers of slices while each slice itself includes macroblocks that built by gathering of four 8*8 blocks.

In general encoding standards such as H.264, utilize three frame type:

- I-frame: intra-frame is the most important frame since it is independent of the other frames.
- P-frame: predicted frame,depends on previous P-frame or I-frame.
- B-frame: bidirectionally predicted frame, decodes the frame by using past frames, as well as future frames.

Video streaming architecture: The 2.4 illustrates the video streaming architecture. The video server is implemented by source video generator in order to send the video with an adequate bit-stream rate.If the stream is going to be used in rel time application such as video conferencing or live video streaming, the encoding process is done one line, otherwise video pre-encoding is performed and encoded video is stored for on demand request.

Streaming protocols and standards: There are several protocols for video streaming over Internet; the most foremost of this protocol are set as following. figure 2.5 illustrate these protocols:

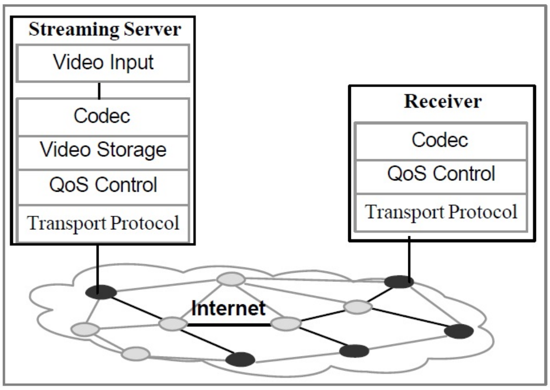

Figure 2.4: Video streaming architecture [39]

- **RTP:** Real time protocol is proposed by IETF to support streaming transmission, it also can support the real-time services provided by underlying network, since RTP is based on best effort delivery, is not able to guarantee QoS parameters. RTP uses the advantages of functionality such as sequence numbering and time stamps to deliver real time packets, these functionality also help RTP to detect lost packets.[RFC 3550]
- **RTCP:** Real time control protocol is designed to control media stream by providing quality feedback includes jitters, delay time and number of lost packets. It also can provide inter stream synchronization and round trip time's value. The feedback can occupy maximum 5 percent of bandwidth and take place every 5 seconds. The most used approaches for media delivering is RTP/UDP and RTCP/UDP.[RFC 3605]
- **SIP:** Session initiation protocol is a control protocol works on the application layer. Its main roles are establishing, modifying and terminating multimedia session.[RFC 3261]It also can invite new participants to an ongoing session.
- **RTSP:** real time streaming protocol which is known as "network remote control" works as application layer. It can control both single and several time synchronized streams. In order to control multiple session, RTSP chooses delivering channels such as UDP and TCP. [RFC 2326]
- **SDP:** According to [RFC 4566] the session description protocol is intended to describe multimedia session in terms of video/audio, bit rates and code methods. SDP using this information to perform session announcement, invitation and monitoring.
- **SAP:** Session announcement protocol [RFC 2974] is a multicast session announcement protocol that assists the advertisement of multicast multimedia

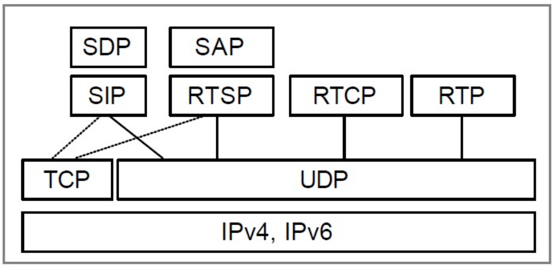

Figure 2.5: Video streaming protocols [39]

sessions. It also sends information about a session setup to other relevant participants to make possible the communication.

2.2.2 H.264/AVC video coding standard

H.264 Advanced Video Coding [66] is a syntax for video compression approved and published by ITU-T and ISO/IEC first in 2003. It benefits some features that makes it more efficient compare whit earlier standards such as MPEG-2 and MPEG-4, features such as error resilience and network abstraction layer (NAL) that helps H.264/AVC data To be mapped to packets over IEEE 802.11 [27]. conceptually H.264/AVC stands on two basis: the Video Coding Layer (VCL) and the Network Abstraction Layer (NAL).[55] defines them as *"VCL creates a coded representation of the source Content, the NAL formats these data and provides header information in a way that enables simple and effective customization of the use of VCL data for a broad variety of systems."* In fact VCL contains compression engine that performs some functions to accomplish coefficient transform coding and motion compensation, the VCL is transport-unaware, VCL uses the higher data structure refers to "video slice". [1] NAL encapsulates video slices and forms them as transport entities.

As 2.6 Shows an H.264 AVC code includes three main processes: prediction, transform and encode, and an inverse transform done in order to reconstruct video stream. A frame is processed in a macroblock [figure 2.3](16 * 16 pixels). macroblocks forms two modes: intra mode or inter mode,In Intra mode *"a prediction macroblock P formed from samples in the current frame n that have previously encoded, decoded and reconstructed".* in inter mode a prediction macroblock *"is formed by motion compensated prediction from one or more reference frames)."[32]* The residual is resulted by subtracting of prediction from the current macroblock, then block transform, and quantization will be performed, then the coefficients will be achieved. At the next

[1] video slice refers to a collection of coded macrocosms.

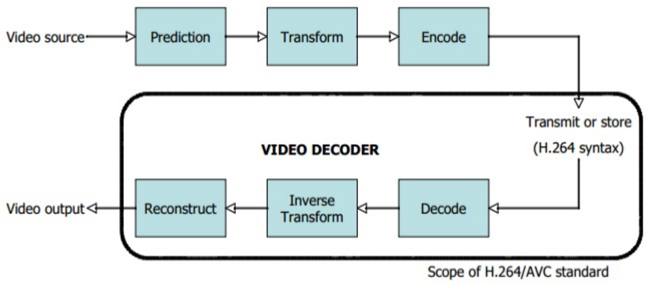

Figure 2.6: Encoding/decoding process in H.264/AVC [31]

step re-ordering is done, then reordered coefficient along with other side information will form the compression stream.

2.2.3 H.264/SVC video coding standard

The H.264 SVC (Scalable Video Coding)is an extension to H.264 AVC standard. It designed in order to deliver scalable streams. Scalable contents are achieved through a layered structure. Layered structure consists of a base layer (with minimum bandwidth requirement and lowest performance) and several enhancement layers that increase video quality, and, therefore, require high bandwidth. the scalability defines in terms of temporally, spatially and video quality: Item:

- Temporal Scalability: The frame rate of an encoded bit stream cab be reduced by dropping packets.[55] defines Temporal scalability as *"the set of corresponding access units can be partitioned into a temporal base layer and one or more temporal enhancement layers with the following property"*

- spatial scalability: stream is coded at multiple spatial (picture size) resolution and coded stream represent original stream with reduced resolution or picture size, in fact, the decoded sample resolution can be increased by means of adding enhancement layer to decoded stream.

- quality scalability: is also known as SNR quality, the ability of improving SNR quality by more layer acquisition. *"With quality scalability, the substream provides the same spatio–temporal resolution as the complete bitstream, but with a lower fidelity, where fidelity is often informally referred to as signal-to-noise ratio (SNR)."*[55]

2.2.4 VP8 Video coding standard

VP8 [12] is a recent open source video compression format. It developed as the core of Google "WebM" project in May 2010. The project aims to fulfill high compression efficiency and low computational complexity. the main focus of VP8 is Internet/web based video, since the following features are addressed :

- Low bandwidth requirement: Bandwidth consumption has high importance in today's networks and will have an even higher importance in the future, due to bandwidth limitation. VP8 operates in a quality range from "watchable video" (30dB in the PSNR metric) to "visually lossless" (45dB)." [12]
- Heterogeneous client hardware: By increasing the hardware technology diversity, today there are a wide range of devices connected to the web. From quiet simple mobile to PADs and more advanced devices embedded in laptops, therefore, implementing an efficient technology to adapt and support all wide range of devices look necessary.
- Web video format:VP8 can support most image formats; "420 color sampling, 8 bit per channel color depth, progressive scan (not interlaced), and image dimensions up to a maximum of 16383x16383 pixels."[12]

VP8 includes two inter-frame and intra-frame and also makes advantage of using three reference frame buffers: the last frame, golden frame and alternative reference frame. Golden frame is a buffer used to store a video frame from previous and alternative (constructed) reference frame is decoded but may or may not be shown in decoder, it is used to improve other coded inter prediction frames.[12] Intra-frame which is corresponded to I-frame of H.264 AVC standard, is the most important frame hence is independent of other frames, while inter-frame is corresponded of P-frame and depends on previous frame included recent intra-frame.[27]

2.3 IEEE 802.11 standard

The family of IEEE 802.11 standards released by Institute of Electrical and Electronics Engineers(IEEE), first in 1997. It provides wireless data transmission in Local Area Network(LAN). *"An IEEE 802® LAN is a peer-to-peer communication network that enables stations to communicate directly on a point-to-point, or point-to-multipoint, basis without requiring them to communicate with any intermediate switching nodes.And independent of device location."*[1] in fact IEEE 802.11 standards make available access network between end users, independent of their location and through radio waves rather than cable connection. Usually WLAN or WiFi is the final link between a wired network (consists of resources and services) and wireless access

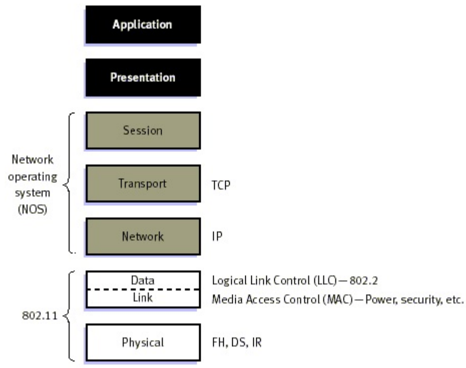

Figure 2.7: IEEE 802.11 on the ISO Model [25]

to clients in a limited area. IEEE 802.11 focuses on physical layer and link layer, and all application, protocols and operating systems are run on 802.11 compliant WLAN.[2.7] There is many motivation behind WLAN, such as cost effectiveness setup and ownership in the dynamic environment, since network interface card and access point as fundamental implementation device are inexpensive; also deployment can be done much more easy than wired network and also cheaper. But the main is "mobility"; user ability to move around while access to the network remains ideally uninterrupted. the freedom in roaming bring several advantages, such that mentioned in [25]:

– Immediate bedside access to patient information for doctors and hospital staff

– Easy, real-time network access for on-site consultants or auditors

– Improved database access for roving supervisors such as production line managers, warehouse auditors, or construction engineers

– Simplified network configuration with minimal MIS involvement for temporary setups such as trade shows or conference rooms

– Faster access to customer information for service vendors and retailers, resulting in better service and improved customer satisfaction

– Location-independent access for network administrators, for easier on-site troubleshooting and support

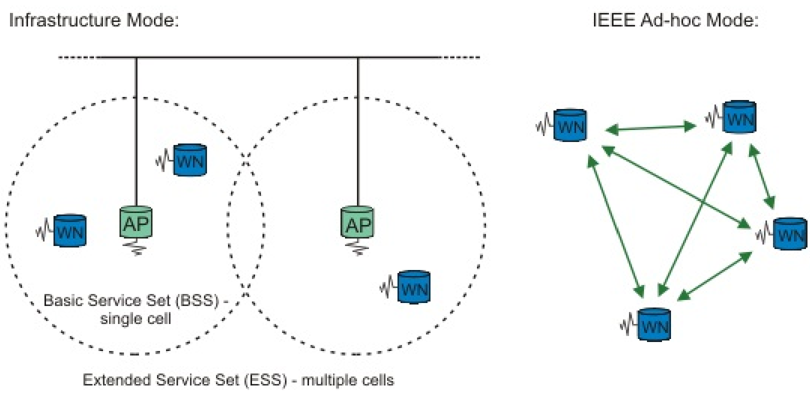

Figure 2.8: Adhoc and Infrastructure Modes [59]

2.3.1 IEEE 802.11 Architecture

The main building block of WLAN named Basic Service Set(BSS) and consists of two or more station that communicating together. In this context station refers to end-user device such as PCs, tablets, smartphones. WLANs include two different modes [Figure 2.8]:

1. Adhoc Mode: which in the devices send data and communicate with each other directly without needing an AP.

2. Infrastructure Mode: this mode benefits of using the existing wired network. All data is passed from wired LAN to 802.11 through "Portal", which acts as a bridge between wired and wireless networks. this mode supports two services:

   – Basic Service Set(BSS): in order to make a WLAN one AP is used.

   – Extended Service Set(ESS): more than one AP play role in creating a WLAN, as a result, users have more liberty in moving area. In order to support the roaming between two adjacent access point, an overlap is is necessary. This is must be from 10 to 15 percent to prevent connection lost. The APs also "should use non-overlapping channels to avoid interference. The most popular non-overlapping channels are channels 1, 6 and 11". [7]

In order to support roaming, all APs within an ESS must use the same Service Set Identifier(SSID). This unique name is used by all devices that are located in the same WLAN. The maximum length of SSID is 32 characters.

2.3.2 Coordination function in IEEE 802.11

According to IEEE 802.11e documentation [5], the coordination function is the logic function which determines when a station (node) is allowed to send or receive a protocol data unit (PDUs) via wireless medium.
Distributed Coordination Function (DCF): the original method for sharing the medium by 802.11, this method employs CSMA/CA and, therefore, many collision may occur when the number of node's increase. in this method includes no QoS grantees.
Point Coordination Function (PCF): is another basic model for 802.11 and only works at infrastructure method. APs send beacon frame every 0.1 second, the Contention Free Period (CFP) and the Contention Period (CP)are defined between beacons, in CP, the DCF is used and in CFP, AP sends a Contention-Free-Poll (CF-Poll) packets to all nodes, this packet indicates if the node can send a packet or not. This method provides a degree of QoS but still is not sufficient enough to guarantee QoS parameters.
In order to achieve QoS in wireless communication, The DCF and PCF improved and made a new coordination function referred as Hybrid Coordination Function (HCF). This new method takes advantage of two following method to get access to the channel, they set different traffic categories (TC) and prioritized different traffics by assigning them to these categories.
Enhanced distributed channel access (EDCA): In this method the node with higher priority traffic waits less than other nodes to send the traffic packets. EDCA achieves this through TCMA protocol. TCMA follows the same logic as CSMA/CA but using shorter arbitration inter-frame space (AIFS) when it comes to high priority traffic.
HCF Controlled Channel Access (HCCA): Is the most advanced coordination function,It works similar to PCF; the difference is using of a controlled access phase (CAP) instead of CFP(Contention Free Period)in PCF. CAP can be initiated at any time during CP. This method can provide a great level of QoS precision, since HCCA makes nodes able to request their intended QoS parameters such as jitter or data rate, is well suited for interactive application such as VoIP and video streaming over Wi-Fi network. This method is used optionally for 802.11e APs.

2.3.3 IEEE 802.11 Standards

In 802.11 Standards unlicensed radio frequencies are managed to different WLAN channels; "The 2.4 GHz band is broken down into 11 channels for North America and 13 channels for Europe. These channels have a center frequency separation of only 5 MHz and an overall channel bandwidth (or frequency occupation) of 22 MHz."[27] The most critical issue in WLAN was limited throughput [25],since IEEE 802.11 only supports low rate transmission, IEEE ratified new standards known as IEEE

802.11a/b/g and n.WLAN covers distance less than 150 m; 802.11a/b/g use 1Mbps to 54Mbps bit-rates while 802.11n advantages 400Mbps.

– IEEE 802.11b: this standard aims to support higher bit-rate than original 802.11 and in order to achieve more robust and higher connectivity, some modification on the physical layer is done. As a result of this changes, physical layer would be able to support 5.5 Mbps and 11Mbps. both bit rates are achieved by taking advantages of QPSK modulation technique.[25] Dynamic rate shifting is another significant advantage of these standards that make it well suited for noisy environments. User devices tend to connect at full 11 Mbps rate but beyond this range, device automatically is adjusted to speed down on lower speed 5.5, 2 or 1 Mbps. The drawback of 802.11b includes: Interference from household appliances such as 2.4 cordless phone, Bluetooth and microwave oven, lack of interoperability with voice devices, crowded frequency bands, and lack of QoS provision for multimedia and support few users simultaneously, are examples of such disadvantages.

– IEEE 802.11a: works up to 54 Mbps data rates and allows 8 non overlapping 20 MHz channels that can work simultaneously across the link layer and physical layer.The standard also allows physical layer to adjust bit rate while higher bit rate is not available, then it uses seven lower data rates: 48, 36, 24, 18, 12, 9 and 6 Mbps. The frequency of 5Ghz is used by 802.11b and helps to avoid interference with other household devices. There are few disadvantages; absorption problem is one, radio waves with higher frequency have lower performance because they absorbed by obstacles such as walls. It also suffers of poorer range in compare with other standards 802.11b or 802.11g.it also is not compatible with 802.11 network components such as routers and APs.

– IEEE 802.11g: this standard developed in order to solve the compatibility issues of 802.11a and now is the most popular in use. It uses up to 54 Mbps and can be uses by simultaneous users. 802.11g has the best signal range, is compatible with 802.11 routers and APs and so on and also is more in terms of obstruction. but it still suffers of 2.4 GHz frequency and interference problem has remained as 802.11b

– IEEE 802.11n: can provides 250 Mbps bit-rates. 802.11n adds Multiple-Input Multiple-Output(MIMO) in order to improve bit-rates and throughput, without additional power and only by using multiple transmitter and receiver antennas (up to 8 antennas). [39] "The multiple input/multiple output technology splits a high data-rate stream into multiple lower rate streams and broadcasts, meanwhile over the available radios and antennae. This allows for a speculative maximum data rate of 248 Mb/s using two streams."[27]

802.11 protocol	Release[6]	Freq. (GHz)	Bandwidth (MHz)	Data rate per stream (Mbit/s)[7]	Allowable MIMO streams	Modulation	Approximate indoor range (m)
—	Jun 1997	2.4	20	1, 2	1	DSSS, FHSS	20
a	Sep 1999	5	20	6, 9, 12, 18, 24, 36, 48, 54	1	OFDM	35
		3.7[A]					—
b	Sep 1999	2.4	20	1, 2, 5.5, 11	1	DSSS	35
g	Jun 2003	2.4	20	6, 9, 12, 18, 24, 36, 48, 54	1	OFDM, DSSS	38
n	Oct 2009	2.4/5	20	7.2, 14.4, 21.7, 28.9, 43.3, 57.8, 65, 72.2[B]	4		70
			40	15, 30, 45, 60, 90, 120, 135, 150[B]			70

Figure 2.9: 802.11a/b/g/n standards Specification

- IEEE 802.11e: 802.11 e is an improved standard to 802.11b and q, which provides QoS features to 802.11. It defines four different access categories(ACs)for traffic: video, voice, best efforts and background.[5] In this standard MAC layer is enhanced to use coordinate time division multiple access and also takes advantage of error correction mechanisms; hence it is well suited for high speed internet services such as VoIP and motion videos. 802.11 works between 2.4 GHz to 2.4835 GHz radio frequency or between 5.75 GHz and 5.850 GHz. The higher frequency offers less interference and faster data transfer. The stations that using this standard are called enhanced station, they work as a central controller for another station. IEEE 802.11 e supports enhanced distribution coordination function (EDCF) and has backward compatibility with HCF.

The figure 2.9 indicates to the most significant features of 802.11a/b/g/n standards:

2.3.4 Handoff in 802.11 networks

Today, Wi-Fi delivers high-speed connections between millions of public places, homes, offices and airports. The main concern is providing seamless connectivity. It is a more important and critical issue in emerging interactive applications such as voice over IP or live video communications and can easily deteriorate the QoS as well as the QoE which user perceived. Since the APs can cover only limited distance(typically under 100 meter), in order to provide good radio coverage of a campus or building, large scale deployments are necessary. Great coordination and management of access points can improve the seamless connectivity, and this can be achieved through a sophisticated hand off method.

Handoff occurs when a client moves outside of the AP coverage area and enters to a new BSS. at this moment it must "handoff" or "handover" to another BS. During this period client connection is interrupted. The handover procedure defined as "the sequence of actions and messages exchanged by access points and a mobile station, resulting in the transfer of a connection from the origin-AP to the destination-AP." [19] Ideally users shouldn't be aware of this change, but in real environment when a

802.11 user moves to the boundary of coverage area, it stops the communication with current AP and sends prob request in order to discover the best AP, this process make a gap in ongoing connection which may last up to a second and leads to packet loss. If the handoff occurs in the same network technology, is called horizontal handoff and otherwise if it takes place across different wireless technologies is known as a vertical handover. for instance from Wi-Fi to Wimax.
Handoff take place in both layer 2 and layer 3. If a mobile node moves from its point of attachment in one network to an adjacent AP on a different network, handoff performs in layer3. If the movement is among two APs but from the same IP network, handoff can be handled in layer 2, this handoff is transparent for layer 3 (network layer). The handoff in layer 2 includes discovery and re-authentication, whilst handoff in layer 3 uses protocols such as Mobile IP(MIP) or fast handover.

The handoff process in layer 2: also known as link-layer handoff is divided to two steps; Discovery and Re-authentication.
Discovery: when a mobile station moves away from AP, the SNR or signal to noise ratio degrades, when the degradation is less than threshold value, the station looks for a new AP with strong power to keep connectivity, mobile nodes find this by using scanning which is a function of MAC layer. APs spread out "beacon" frames periodically. these frames contain some advertising and timing information which based on these information , mobile node can choose the best AP. according to IEEE 802.11 standard there are two scanning methods:

- Passive scanning: mobile nodes listen to each channel, but they are only able to receive the beacons from the AP in the same channel range. As it mentioned already, IEEE 802.11 b/g work in 2.4 GHz and use 11 channel (out of 14 possible channel), while IEEE 802.11 a operates in 5 GHz with 32 channel.

- Active scanning: describes the method in which, mobile node sends a "prob request" frame contains broadcast destination address and the a prob timer. Mobile node waits for a "prob response" from the APs within the timer time. If mobile node does not receive any response during *MinChannelTime*, scans the next channel otherwise stop receiving responses at *MaxChannelTime* and chooses the current best AP by means of processing all prob responses. The carrier sense multiple access with collision avoidance(CSMA/CA)is used to manage the collision.

Since passive scanning increases the delay, the later method are more used by 802.11 due to its efficiency and better speed. The active procedure includes following steps:
Re-authentication and **re-association**.For completing handoff process the new AP must authenticate and re-association the mobile station. To achieve this, mobile

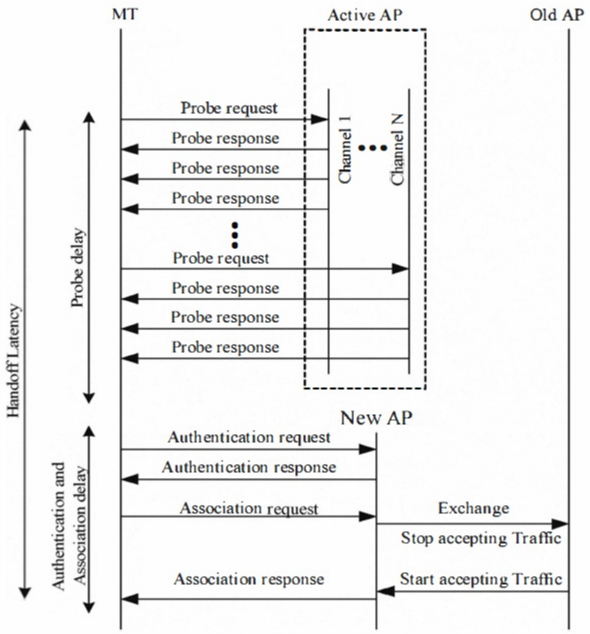

Figure 2.10: Handoff delay in IEEE 802.11 layer 2[70]

nodes credential is sent from old AP to new AP. the following steps accomplish re-authentication and re-association:
An authentication request is sent from mobile node to the new AP, then AP response to this request (either acceptance or rejection) after authentication has done, the re-association request is sent and a response is received. authentication is done according to priority list and also IAPP protocol [2](by transferring station information and credentials from old AP)[19]. the figure 2.10 indicates the different steps and handoff latency:

The handoff delay in layer 2: As the figure 2.10 shows, in layer 2 the handoff delay is the summation of probe delay and authentication/association delay:

– Probe delay: depends on scanning mode varies. in the passive mode, prob delay equals to beacon interval time multiplication by the number of available channels, while in active scanning, prob delay is device-dependent. if there are N available channel then prob delay is bounded to:

[2] Inter-Access Point Protoco is an extension to IEEE 802.11 and enables access point interoperability in multivendor systems, also provides roaming of 802.11 Stations within IP subnet.

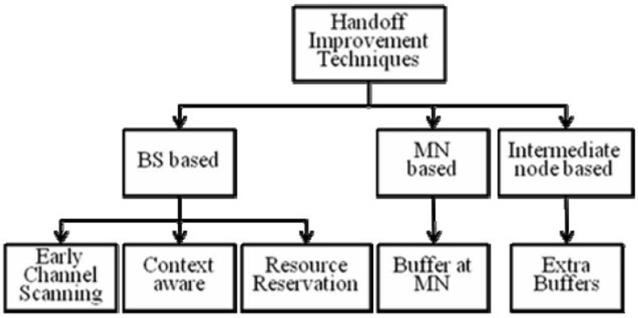

Figure 2.11: Handoff delay improvement [8]

$$N \times MinChannelTime \leq T_A \leq N \times MaxChannelTime$$

In order to reduce prob delay, the number of channel can be decreased or the values of *MinChannelTime* and *MaxChannelTime*, hence the channel waiting time depends on these values.

– Authentication/association delay: is a reason of re-authentication and re-association frame exchange. authentication is based on shared-key using Wired Equivalent Privacy (WEP) algorithm, this process includes four message exchanges between AP and MN: Challenge-Request message, Challenge-Response message,Response message and Approval message. in fact authentication delay incurred by the number of message exchange.in the same way, re-association delay occurs due to the number of re-association frames exchanges.

In addition to above mentioned intuitive methods for handoff delay and packet loos reduction, several other solutions are proposed, either through improving the AP or improvement at mobile node and also by using an intermediate node between AP and mobile nodes.most of them have focused on layer 2 and some other work on both layer 2 and layer 3. Adhvaryu and Raisinghani [8] classified the handoff improvement schemes in two general categories (Figure 2.11 shows the hierarchy)[8]

1. Handoff delay reduction: method is based on performing the initial phase of handoff in AP, or "by transferring context of the network to the other APs". [44]

2. Using buffer to store data packets:improvement is done through creating buffers at AP, mobile nodes or intermediate nodes.

More details about above mentioned methods and significant proposed solution are described in the next chapter.

The handoff process and delay in layer 3: Also known as network-layer handoff, and usually in large service area are used since more likely the network is divided into different subnetworks. MIP version 4 and 6 are widely used as handoff protocol in layer 3.
In the network-layer handoff, moreover than layer two delay, Mobile IP movement detection and Mobile IP registrations delay must be considered. In fact the duration of mobile IP handoff consists of the summation of the three following steps [22]:

– Link-layer hand-off

– MIP Movement Detection

– MIP Registration

To reduced the handoff delay in layer3 and decrease interruption during movement, ITEF has proposed many different mobility management protocols, such as Fast handover for MIPv6 protocol and hierarchical MIPv6. Since the last method is tested at this project, it will be presented with more details in the follow. besides describing Fast handover for MIPv6 protocol and Mobile IP that is the fundamental of two former protocols.

2.3.5 Mobile IP

In mobile IP (MIP) mobility process is improved by adding new entities such as home agent and foreign agent. Home agent is responsible for intercepting the packets and forward them. In this way, a mobile node has two addresses: one is a static address uses for identifying in the home network, and the second address is Care of Address (CoA) which is assigned to MN when it visits foreign network. CoA is using to route packets. MN sends the Binding Update Message to the home agent to informed it about the new CoA, home agent then updates its binding cache entries which contain the pair of the home address and CoA of MN. Corresponding node (CN)sends the data to MN's home(static) address, the home agent intercept and forwards the data to CoA. This is done through IP tunneling. Mobile IPv6 handoff procedure is illustrated in figure 2.3. As figure indicates there are three procedures: movement detection, configuration and registration.[67] Movement is detected through analyzing the Router Advertisements. MN uses auto-configuration feature to create CoA, in order to prevent duplication address, MN takes advantages of duplication address detection. After updating the binding cache and complete registration, MN will receive the packets.

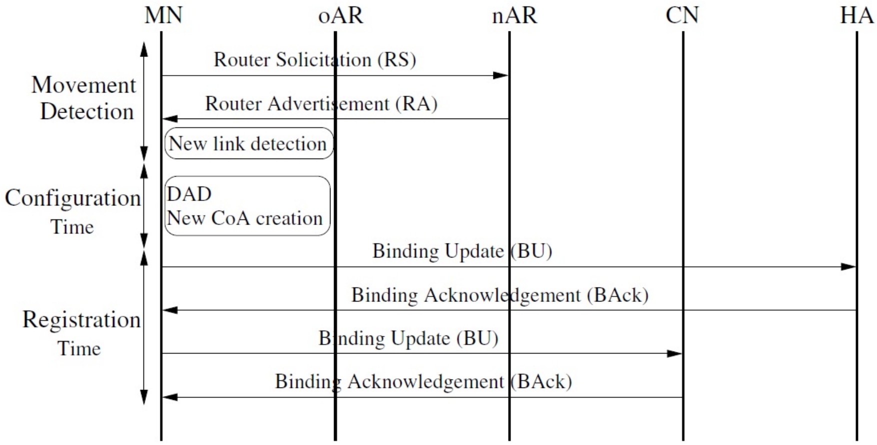

Figure 2.12: Handoff in Mobile IPv6 from[67]

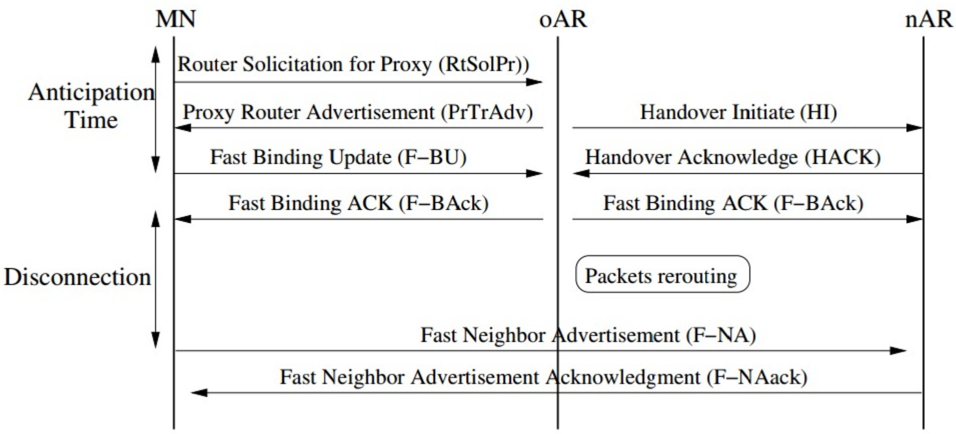

Figure 2.13: Fast Handovers for Mobile IPv6 from[67]

2.3.6 Fast Handovers for Mobile IPv6

Fast handover for mobile IPv6[36] proposed to overcome the service degradation during handoff. The protocol is based on two approached: tunnel-based handover that triggers the link layer and anticipated handover which relies on network layer. Anticipate handoff is considered as "make-before-break" approach. The figure illustrates this approach.

MN has informed about its new location when receives the Router Advertisement from new access router(NAR). Then MN informs previous access router(PAR) by sending a Router Solicitation for Proxy (RtSolPr). Next, the previous access router makes a new CoA. The nCoA and NAR address are sent to MN via a Proxy Router Advertisement. The PAR also sends handoff initial (HI) message to NAR; this message contains both old and new care of addresses. NAR checks the validation of proposed nCoA, if the validation is confirmed; nCoA is added to neighbor cache. NAR informs PAR about validation by sending a handover Acknowledge message (HACK), when PAR receives HACK, will forwards packets to nCoA. Upon the MN received PrRtAdv (which contained confirmation and nCoA),will send Fast Binding Update(FBU)to PAR. Then PAR sends Fast Binding Acknowledgment (F-BACK) to both MN and NAR. As soon as MN attaches to NAR, will send Fast Neighbor Advertisement(F-NA) to initiate the forwarding of packets.

FMIPv6 Handoff delay

As it described earlier and figure 2.15 shows, FMIPv6 reduces handoff latency through anticipating the handoff by using layer 2 trigger. According to [36] the handoff latency is defined as "the time duration from the MN receives the Fast Binding Acknowledgment(F-BAck) from the oAR with which it is currently associated until the MN receives theFast Neighbor Advertisement Acknowledgment(F-NAack) from the nAR" [67]. Based on literature [67], [56], [54], [58],[62] the total hand off is calculated by following equation:

$$T_{FMIPv6} = T_{(MN-AR)} + T_{(AR-MN)} + T_{dis} \tag{2.1}$$

Where: $T_{(MN-AR)}$ represents the duration of time for sending a signaling message from MN to access router.
$T_{(AR-MN)}$ is the time spent for sending the signaling message from an access router to an MN, and
T_{dis} shows the disconnection delay and can is expressed as the summation of the time for receiving L2-LD (disconnection of layer 2 link between MN and old access router) and L2-LU triggers (establishment of layer 2 links between MN and new access router):

$$T_{dis} = T_{(L2-LD)} + T_{(L2-LU)} \tag{2.2}$$

2.3.7 Hierarchical Mobile IPv6

Hierarchical Mobile IPv6 is a micro-mobility protocol that minimizes the signaling load to corresponding node and home agent. It also decreases handoff latency and packet loss. Micro-mobility protocol concerns in mobile node movement within a

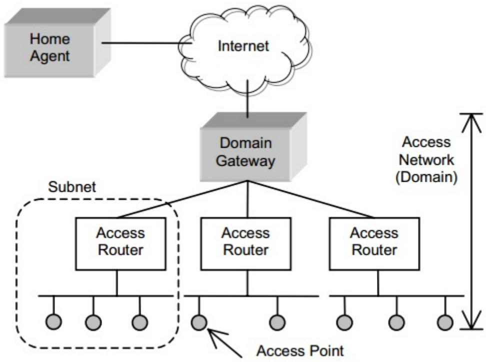

Figure 2.14: micromobility architecture.[28]

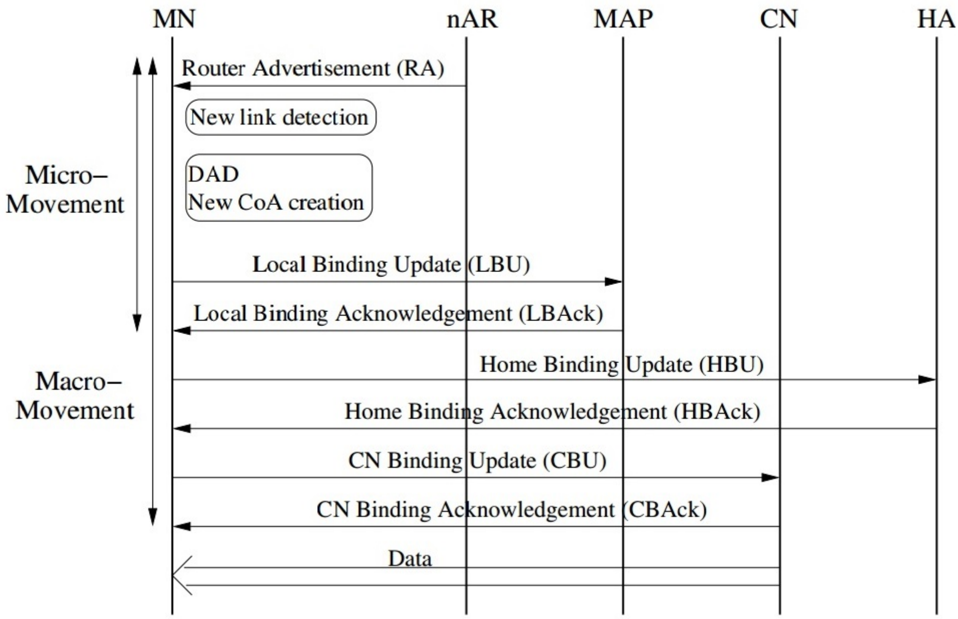

Figure 2.15: Handoff delay in Hierarchical Mobile IPv6from[67]

subnet [28]. Figure 2.4 depict a general structure of micro-mobility architecture. In this architecture a domain gateway is used to localize the handoff signaling, keep track of MN's location and keep the connectivity of domain and internet.

Xie et.al. in [67] defines the delay in HMIPv6 as "the time after an MN sends out the Local Binding Update(LBU) to a MAP until it receives the first data packet from the new subnet." According to [56], [67], [58], [62]HMIPv6 handoff delay, when the movement takes place inside MAP is calculated by:

$$T_{HMIPv6} = T_{(MAP-AR)} + T_{(AR-MN)} \tag{2.3}$$

In Where:
$T_{(MAP-AR)}$ indicates the delay in sending signaling message between MAP and access router, while $T_{(AR-MN)}$ represent the delay when a signaling message is sent from access router to the mobile node. If the movement take place between MAPs [3], the summation of signaling delay from MN to corresponding node (or home agent) and also from corresponding node (or home agent) is added to the equation 2.3:

$$T_{HMIPv6} = T_{(MAP-AR)} + T_{(AR-MN)} + T_{(MN-CN)} + T_{(CN-MN)} \tag{2.4}$$

2.4 Mobility models

Mobility model and traffic pattern are key parameters for wireless protocol simulation. Since they have the significant impact on the performance of protocols, it is necessary to study and analysis various mobility models. ([11], [14], [40], [32]).The purpose of using mobility models are *"to describe the movement pattern of mobile users, and how their location, velocity and acceleration change over time"*[11]. In this section, two mobility models are described: The Random Waypoint Model that is based on *"random directions and speeds"* and Gauss-Markov Mobility Model that is based on using *"tuning parameter to vary the degree of randomness in the mobility pattern"* [14]. These models are also used within the simulation scenarios.

2.4.1 Random Waypoint Mobility Model

The Random Waypoint model [13] is a simple and available model, therefore, is widely used in network simulator ns-2. In this model, an MN stays in its location for a set time that is called pause time. When the pause time expires, the MN can choose another destination randomly. MN's speed is between [*minspeed*, *maxspeed*] and follows a uniform distribution. Once MN arrived at destination, pauses again for the pause period. (figure2.16) This model can also be used without pause time[35].

The maximum velocity (V)and the duration of pause time(T) play an important role in node(user) mobility behavior and, as a result, network stability. A large V (high speed movement)and short Tcause the unsuitability of topology and make the network highly dynamic. While in contrast, small V and long T lead to relatively stable network.[11]

[3]Movement between MAP domains is known as macro-movements.

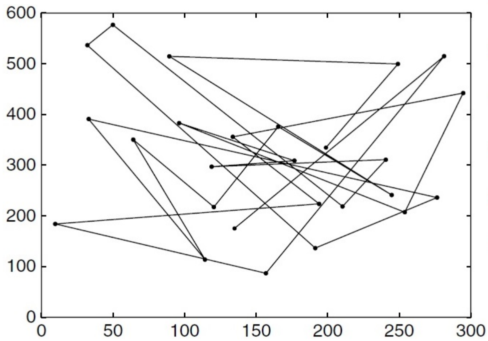

Figure 2.16: Traveling pattern of an MN using the Random Waypoint Model [14]

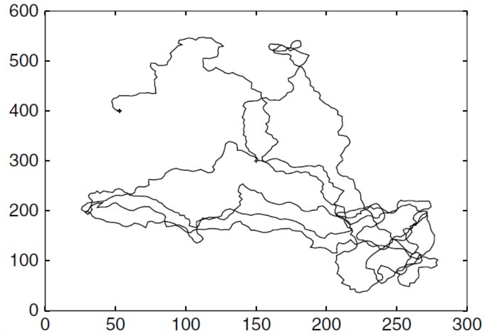

Figure 2.17: Traveling pattern of an MN using the Gauss-Markov Mobility Model [14]

2.4.2 Gauss-Markov Mobility Model

As mentioned earlier, Gauss-Markov Mobility Model [42] uses a tuning parameter to adapt different randomness [14]. In this model, each node has a current direction and current speed that are updated during time intervals. At each time the value of speed, location and direction of the n^{th} instance depend on the $(n-1)^{th}$ instance values. [14] shows the following equations(2.5, 2.6) in order to calculate speed and direction for nth instance s_n and d_n respectively):

$$s_n = \alpha s_{n-1} + (1-\alpha)\overline{s} + \sqrt{(1-\alpha^2)s_{x_{n-1}}} \tag{2.5}$$

$$d_n = \alpha d_{n-1} + (1-\alpha)\overline{d} + \sqrt{(1-\alpha^2)d_{x_{n-1}}} \tag{2.6}$$

In these equations α represents tuning parameter which is used in order to vary randomness, $0 \leq \alpha \leq 1$ which means if $\alpha = 0$ the random values are obtained and if $\alpha = 1$ the linear motion will obtain. s_n and $\overline{s}$ are mobile node speed at interval t and the mean value of speed respectively and d_n and $\overline{d}$ show the mobile node direction at interval t and the mean value of direction respectively. In addition, $s_{x_{n-1}}$ and $d_{x_{n-1}}$ denote to random variable of Gauss distribution. [44]

Chapter 3
Related works

In this chapter, review of relevant literature and previous works in this area are presented. Since the thesis includes three main broad area in both technology and academic researches, this chapter consists of sections for video codec over IEEE 802.11 (3.1), performance of fast handoff methods (3.2)and multicast streaming over the wireless network (3.3. As already mentioned, due to widespread researches, only few more relevant or more innovative ideas are presented in each part.

3.1 Video codecs over IEEE 802.11

Transporting network-friendly video codec is an important challenges in recent studies. However different codecs has been developed during recent years, but since wireless network has an error-prone environment, supporting different QoS in such distributed environments is a challenging issue. this challenge has more impact on real time services such as live video streaming.
IEEE 802.11e was developed to support the QoS in wireless network by adding access categories. the main focus of recent studies are on different cross-layer methods to map the videos with different priority to appropriate ACs.

3.1.1 Enhanced cross-layer architecture

Ksentini and Naeimi proposed a cross-layer architecture for H.264/AVC standard in order to support QoS features. [37] the scheme works based on the importance of each fragment. The architecture performs as top-down and cross-layer interaction. The first interaction allows H.264 NAL to send information regards QoS requirements to the network layer and second interaction provides the same QoS level at EDCA-based MAC layer.
In [21] the anthers suggested an algorithm to prioritize traffic of both H.264/SVC and AVC videos over ah hoc networks called Traffic Prioritization Algorithm (TPA).The

algorithm consists of two sequential stages, the first stage performs on NALUs [1] and the second stage packetizes the NALUs and then based on their priority level assigns them to the right AC, by using these two stages the most important part of stream is protected. According to [21] *"better visual quality can be obtained when the SVC codec is used in place of AVC for the same source coding bandwidth, in the same network conditions and using the same amount of network resources, especially in the case of high packet loss rate conditions. While the AVC codec offers in fact a very simple form of temporal scalability only, SVC provides advanced scalability options which enable more versatile traffic prioritization schemes such as the proposed one, which provides a better visual quality without consuming additional network resources."*

3.1.2 QoS evaluation of latest video codecs

In [69] Yoon et. al. studied the latest video codecs such as H.264/AVC , H.264/SVC and VP8 over wireless network. They also analyzed these methods performance by means of both network QoS metrics such as packet loss ratio and end to end delay and user QoE metrics such as peak signal to noise ratio (PSNR) and mean opinion score (MOS). The results are achieved through several different scenario simulation using the Open evaluation framework multimedia over networks (OEFMON) and based on different AC mapping schemes.
The result of their study indicates a robust multimedia transmission over IEEE 802.11 wireless network is accomplished by reducing the coded video data (including queue size of ACs and implementation of error recovery features). In addition, due to simulation results, VP8 achieved good quality with basic MAC techniques,in fact it provides better user-perceived quality by using distributed coordina-tion function (DCF) and enhanced distributed channel Access (EDCA).while H.264/AVC and H.264/SVC performance shows better user-perceived quality when using appropriate mapping scheme of encoded video data to ACs. [**?**]

3.2 Performance of fast handoff methods

There are numbers of innovative and notable schemes have been proposed in order to overcome handoff delay in IEEE 802.11 networks. Some of them are presented in follow:

3.2.1 Fast handoff schemes

An example is reducing prob delay in a method referred to SyncScan [50]. Shin et al. in [57] suggested a selective algorithm and caching mechanism to reduce the MAC layer latency, called channel mask scheme. In some research the authors focused on reducing re-authentication/reassociation delay such as the predictive scheme

[1] NAL Unit; Independant decode transport unit created by NAL

introduced by Pack and Choi, referred as frequency handoff region (FHR)[47].Another method aimed to reduce latency by decreasing authentication handoff proposed in [45]. It is eventually adopted as IEEE standard and is known as proactive neighbor cashing. The method uses a neighbor graph. This method however reduces the handoff delay but causes increasing the overhead signaling due to context exchanging between APs. Especially the problem with over heading has more impact on networks that consist of large number of APs.

To balance the trade-off between handoff delay and the high overhead signaling, selective neighbor cashing proposed in [48],this scheme add "neighbor weight" to the PNC scheme.

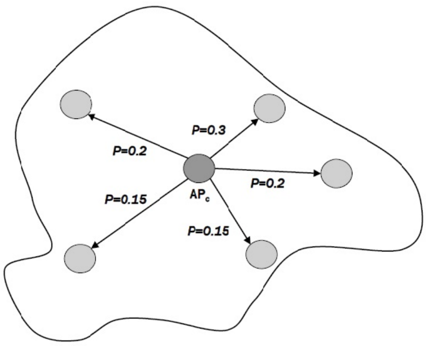

Figure 3.1: The PNC Operation [48]

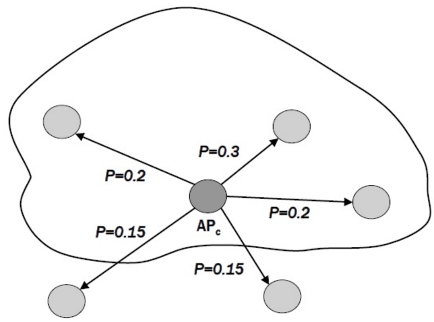

Figure 3.2: The SNC Operation [48]

The handoff probability for each neighbor AP. Based on the neighbor weight, the MH context is propagated only to the selected neighbor APs *"(i.e., neighbor APs with equal neighbor weights to or higher neighbor weights than a threshold value)."* [48] the figures 3.1 and 3.2 illustrate PNC and SNC schemes. The simulation results[48] indicate that"the SNC scheme showed a similar cache hit probability to the PNC scheme while reducing the signaling overhead significantly. The performances of the SNC and PNC schemes are expected to be highly dependent on the cache size and cache replacement policy".

There is also various techniques which have focused on APs architecture in order to minimize handoff latency, such as the scheme proposed by Wu, Tan and Zhang. Since handoff starts with channel scanning, a high amount of time is wasted during this stage. Authors in [24] suggested channel scanning takes place at AP. the idea is that the AP monitors available channel of neighboring APs, therefore, when the MN needs to handoff to another AP uses these already monitored channels. The drawback of this method is unnecessarily increasing in AP's load. in addition handoff

deficiency will be highly degraded by growing in the number of nodes which intend to simultaneously handoff.
The other suggested architecture is S-MIP suggested in [30] as a seamless handoff architecture for mobile IP. [30] In this method the next AP that MN intended to connect is chosen due to the MN's current location. While mobile node initiates the handoff process, the packet will be forwarded to both current AP and the AP which with high probability the node is going to attach to. The method although reduce the latency in transferring packets from old AP to new one, but since is dependent to MN's location, id location is not available the scheme does not work anymore. Also this method requires a large buffer at APs.

Buffer based solution is another proposed technique. In [20] authoress proposed buffering scheme in order to reduce packet loss and quality degradation. The buffer can locate at AP or any intermediate nodes. When handoff process completes, the packet in the buffer, are transferred to the MN. By using a buffer, packet delay will increase which in delay sensitive and real time traffic is not acceptable. In addition, the scheme works properly only when the limited number of MNs connect to the same intermediate node.

3.2.2 Handoff performance

There are some schemes proposed to handle mobility managements, the baseline of these protocols are mobile IP protocol. Many simulation researches have made on different aspect of these protocols, few of them are mentioned here.
In [61] Tsang et al. compared the performance of Mobile IP, mSCTP and Fast handover. The result shows the lowest throughput and longest handoff latency for mobile IP, while fast handover has better performance since, in fast handover, MN is allowed to do handoff at appropriate time after beginning of data transmission between routers. mSCTP that is used for multi homing avoids the problem with timing ambiguity but packet reordering is its drawback.
A performance study of fast handovers for mobile IPv6 is made in [60].NS2 simulator used to study the behavior of each method on up to 50 nodes under condition of different traffic (CBR, VoIP, Video and TCP). The simulation result shows *"Mobile IPv6 can eventually outperform FMIPv6 in packet losses terms in saturation conditions due to the higher number of packets discarded directly in the Neighbor Discovery entry queue that lower the load in the wireless channel.Random movements affect the experienced performance improvements but the difference in the perceived quality of service when using FMIPv6 is still clearly noticeable. In scenarios where the users produce a low rate with small packets, e.g., VoIP sources, the additional load in the wireless channel introduced by FMIPv6 can result in a worse performance than the baseline Mobile IPv6 one. By using pure Mobile IPv6 and enabling the option of establishing forwarding from the previous care-of-address, handoff latency and packet losses can be also improved without requiring and additional implementation*

effort and at a lower signaling load cost, however, without reaching the same level of improvement achieved by FMIPv6." [60]
Zhang and Pierre in [71]evaluate the performance of fast handover for hierarchical MIPv6 in cellular networks. They used fluid-flow and random walk mobility models.during their analytical study various parameters such as packet delivery cost, location update cost and total cost considered, they also investigated on the impacts of different factors like velocity, user density and mobility domain size. The results of their simulation show that F-HMIPv6 has more signaling cost for the location update while its packet delivery cost is same as HMIPv6. They concluded F-HMIPv6 is a better method for intra-domain roaming due to its high signaling overhead.

3.3 Multicast video streaming over the wireless network

As it described earlier through introduction chapter a wireless transmission faces with two main types of errors: either those occurs due to the nature of wireless environment, such as fading that depends on distance between source and destination, or those who are not depends on wireless wireless medium and channels such as collision. However the error-prone and bursty environment of wireless network can increase the collision.
There are several proposed methods to handle errors in wireless networks which some are aforesaid in previous section such as CSMA/CA or RTS/CTS are used by IEEE.802.11, but when it comes to multicast transmission the aforementioned methods are not sophisticated enough, for instance assume all n members of a multicast group send ACK to source to informed it about successful receiving, then a feedback implosion in source is expected. According to IEEE 80.11 standard the packets can be sent broadcast or multicast but since there is no feedback mechanism in this standard multicasting must take place at low and fixed bit rate. the lack of feedback mechanism such as ACK raises serious issues in reliability and efficiency. During last two decades, multicast over wireless networks was considered by many authors through different researches and papers.The prominent approaches are classified in following groups, however they neither cover all investigations nor the whole detailed.

3.3.1 Pseduo-broadcast approaches

Pseudo-broadcast is the main idea of many proposed methods to overcome multicast issues. In pseudo-random method, a node is selected out of multicast group, it acts as a unicast receiver hence AP must be modified to send RTS request to this node, when handshaking takes place through RTS/CTS and an ACK received by multicast source, then the data is sent to a broadcast address.
Park et. al. [49] used this idea to deliver IPTV to multi-room over IEEE 802.11 links, their proposed system includes a gateway which intends to send video streams

to set-top boxes where are located in different rooms, the gateway sends a unicast stream to designated node, the other nodes get the stream by listening promiscuously. according to their findings *"The pseudo-broadcast uses less wireless channel time than broadcast, but with retransmissions and rate adaptation as in unicast it has much higher delivery rate."* which leads to less bandwidth consumption and better video quality.
In 2009 Chandra et al [3] suggested a wi-fi multicast system referred as DirCast, it also works based based on pseudo-broadcast, The enhanced their model by adding more than one multicast group and one AP. DirCast has advantages of proactive adaptive FEC , destination control and association control to further reduction of loss rate.in order to have association control a DirCast server designed. The server is responsible for AP selecting when a new new member intended to join to a multicast group.This leads to air time reduction. Based on observed loss rates, the server chooses the target node for each multicast group. This node must be selected whenever a receivers observes more than 10 percent packet loss or a new member joins to the multicast group. Moreover redundant packets are sent proactively to reduce receivers' loss experiences.
Intra-based multicast service(LBMS) [16] is another research published in 2010, in this investigation authors suggested the use of leader node. A node is selected as leader and is responsible to send feedback to multicast data sender. In compare with pseudo-broadcast in which the multicast data use a unicast address (selected node address), the LBMS uses the multicast addresses as destinations. If the sender does not receive ACK from leader, increases contention windows in double. their simulation results illustrated *"Retransmissions, exponential back off, and rate adaptation in the LBMS help to achieve reliability for multicast, fairness between multicast and unicast, and wireless channel efficiency, respectively."*
In 5th international symposium on wireless computing a new mechanism presented for transmission of multimedia video contents to a large group of users over Wi-Fi networks referred to as WEVCast. [46] WEVCast which stands for Wireless Eavesdropping Video Casting, aimed to overcome few limitations of the multicast transmission without any changes to the 802.11 MAC. The WEVCast scheme consists of video server connected to AP. A target node (known as WEVCast TN) is selected, the video server provides video streams for WEVCast TN through a unicast link from AP, during this transmission all other client nodes (CNs) inside the AP's coverage area,receive the video contents by mean of eavesdropping transmitted data packets. WEVCast scheme dose not change the MAC, it uses MAC address spoofing on the CNs to change an assigned MAC address to another device. *"to make them indistinguishable from the TN on which an unicast connection has been set. In this way, the AP will not be aware of the presence of many CNs within its coverage area because everyone has the same MAC address and will be indistinguishable."* as well as IP address alias to forwards the video packets from MAC layer to higher layer,

"The IP address alias mechanism consists in the configuration of another IP address that is usable only by the specific terminal because it is logically implemented above layer 3; thus it is not reachable from other terminals within the network but it can be used within the same CN to forward video packets coming from MAC layer to the Network layer.". [46] The simulation results imply that by reducing the delay between packets, better PSNR is resulted.

3.3.2 Leader-based approches

One of the main issue in pseudo-broadcast approaches is to elect the right node as target node or designated node. In [38] Kuri and Kasera introduced leader-based protocol (LBP), a novel mechanism to solve reliability issues in wireless LANs.One member from multicast group is selected as leader in order to send feedback (CTS/ACK) to the sender. LBP use both ACK(Acknowledge) and NACK(Negative Acknowledge) as feedback, in fact LBP can be considered as an alternative for ARQ protocol in MAC layer. LBP approach is the basic idea of so many later investigations.some of them are pointed out here.
Choi et al. presented leader-based rate adaptive multicast for wireless LANs, called LM-ARF [18] which includes two main functionality: leader-based feedback and rate adaption. The leader is a receiver node responsible for transmitting ACK to the sender. In this approach if the other multicast participants don't receive multicast frame, can also request re-transmission by sending NAK.At the beginning of transmission, AP sends a CTS-to-self frame at the rate 1Mbps or 2Mbps,in order to *"guaranteeing the channel access and announcing the transmission of a multicast frame."* [18] the CTS-to-self frame also reserves the channel during transmission. [18] also uses a rate adaption mechanism in order to overcome inefficiency of multicast transmission. It is done through ARF mechanism. ARF consists of a timer and a tuple of current data rate and number of successful and failure consecutive transmission. Due to current status, the ARF decides to increase or decrease multicast data rate.
In 2010 the same authors came up with a framework for multicasting over WLANs called Leader-Based Multicast Service (LBMS)[17] This approach includes LBMS request/report action frames, leader election protocol, leader-based multicasting. The two first frames are management frames compatible with IEEE 802.11v . when a node joins to a multicast group, transmit an LBMS request to AP. this frame consists of multicast address, multicast ACK policy and retry limit. Then the AP sends a LBMS report frame to the leader node. In order to select leader node, leader election protocol is defined which includes different selection algorithms. The AP elect the leader by using arbitrary criterion, then informed the chosen node, by sending a LBMS report frame. and eventually the leader should send ACK to the AP after each successful reception, otherwise the AP will double the contention window in order to achieve fairness.
At the same year Miroll, Li and Herfet proposed wireless feedback cancellation for

leader-based MAC layer multicast protocols.[43]The main focus of their investigation is on ARQ based NACK LBP. Similar to other LBP approaches a receiver is selected as leader, before data transmission AP sends a sequence (SEQ) control frame to all receivers in order to inform them about data sequence number and to set their timer. After receiving data leader sends an ACK to AP to confirm successful data receiving. If any of non leader receivers has not received the data correctly, will send a NACK frame to AP, the NACK then collides with ACK frame issued by leader node and thus the positive feedback of ACK frame will be canceled and AP should repeat the data transaction. The capture effect [2] can influence the probability of successful cancellation. According to [43] the weakest receiver is selected as leader since the correct cancellation rate increases by decreasing oh ACK channel condition. the authors also believe *"a dynamic leader selection is required and may increase feedback cancellation probabilities dramatically."*[43]

3.3.3 Adaptive scheme

In order to error recovery in multicast content several approaches are used, such as FEC at forward error correction but since it lead to redundancy and increased channel traffic. According to researches a better outcome is expected by using transcoders. Chen et al. suggested an error resilient transcoding. [15] To achieve this transcoding, a two-pass intra-refresh transcoding is used which compresses video content at a media gateway. The transcoder is able to change the intra-refresh according to channel conditions such as packet loss rate and video content. Their proposed method includes *"a minmax loss-rate estimation scheme so as to determine an appropriate intra-refresh rate for all the clients."* [15]
In [41] the authors proposed and analyzed the performance of an error control scheme for multicast video transmitting over IEEE 802.11 WLAN, this scheme is based on interference eliminating to guarantee stream transmission, *"The proposed error control scheme makes the receiver node report errors in a best-effort manner via CP [3], and also make AP retransmit the requested packets through the over allocated slot that is unavoidably brought by QoS guarantee at CFP [4]".* [41] The error list is reported as long as receiver node receives the message, by taking note that each frame contains message size.
The problem with adaptive scheme over IEEE 802.11 WLANs is increasing overhead. Since these schemes require feedback from all receivers or a large number of them.

[2]The phenomenon indicates if two signals in the same frequency or near same frequency sent to a receiver, only the strongest one will be demodulate.

[3]Contention Period

[4]Contention Free Period

3.3.4 Cooperative computing

In order to increase coverage range,data rate,robustness and also reliability a new approach was proposed:"Cooperative Computing", the basic idea idea behind this approach is forming an ad hoc network, in the sense that if a receiver node losses a packet, it is allowed to recover it from another closed node. AP also can transmit multicast data to a "relay node" and then relay node send it out to the other adjacent nodes at higher bit rate.[68]
In [64]the reliable multicast based video conferencing studied. Wang et al. proposed a novel linked-layer protocol referred to as multicast with orthogonal CTS (MOCTS) protocol. In this approach spread spectrum modulation is done through orthogonal Walsh code is used to modulate CTS packets at receiving nodes. then the modulated CTSs send to source node and decodes there. so it enables the parallel transmission of CTSs from all receiver nodes.The simulation results illustrates by using MOCTS protocol the multicast CTS collision reduced as well as the total transmission effort that decreases more than 20 percent.
In [52] the anthers did a study on mobile IPTV deployment across wireless mesh network, to meet the reliability they employed multicast cooperative communication. In first step the relay nodes are selected, they classified relay nodes to two groups: in-group and out-group. then WMN considered as a typical mobile environment and by using a distributed relay-selection algorithm, the user mobility is provides. the simulation result indicates that *"approach can significantly improve the error-countering capability of WMN multicast with a small number of relays."* [52]
cooperative layered video multicast using randomized distributed space time codes (RDSTC) in addition to packet level FEC introduced in [10] to achieve error resilient video delivery . It consists of a multicast two-hop transmission scheme used in infrastructure-based networks.In these schemes randomized distributed space time codes (RDSTC) is performed in order to enable simultaneously data forwarding from different relays. *"This randomized cooperative transmission is further integrated with layered video coding and packet level forward error correction (FEC) to enable efficient and robust video multicast."* [10] The authors used three methods to find the system operating parameters, the proposed methods are based on channel's availability at source node, RDSTC with full channel information,limited channel information or with node count. The three schemes performed with rate adaptive direct transmission and conventional multicast (without rate adaption), the comparison of results comparison show that *"while rate-adaptive direct transmission provides better video quality than conventional multicast, all three proposed randomized cooperative schemes outperform both strategies significantly as long as the network has enough nodes. Furthermore, the performance gap between RDSTC with full channel information and RDSTC with limited channel information or node count is relatively small, indicating the robustness of the proposed cooperative multicast system using RDSTC."*

Chapter 4
Design

In this chapter, the system design is described. This design is based the knowledge and motivation from previous chapters. (Background and Overview). The chapter follows by describing network model in section 4.1. Section 4.2 belongs to a short introduction of QoS parameters that are test during simulation in NS2. According to Issariyakul and Hossain [31], there are 3 key steps for simulation in NS2:

- Step 1: Simulation Design
- Step 2: Configuring and Running Simulation
- Step 3: Post Simulation Processing

The first step is described in following sections, while two other steps will be discussed in the next chapters.

4.1 Network model

The figure 4.1 illustrates the simulated network through this project. The model is known as a wireless-cum-wired network. The network includes one fixed video server that is connected to three APs (AP1, AP2 and AP3) through reliable wired links. The transmission of video packets from the video server to APs is unicast, while the APs are located in a given fixed coordination and cover the whole simulation area (a 1000 * 1000 square meters). The focus of the simulation model is on the behavior inside the wireless environment. The network flows are either downlink or uplink. If the flow is transmit from a mobile node to AP is considered as uplink and in the opposite way (from AP to MN) is downlink.The most data flows are sent from AP to MNs (downlink). Since the transmission inside WLAN is multicast, there are no ACK from the receiver (MN) to the AP.
The WLAN area is divided to three domains, each of which is covered with either AP1, AP2 or AP3. The number of mobile nodes inside each domain is varied (Maximum

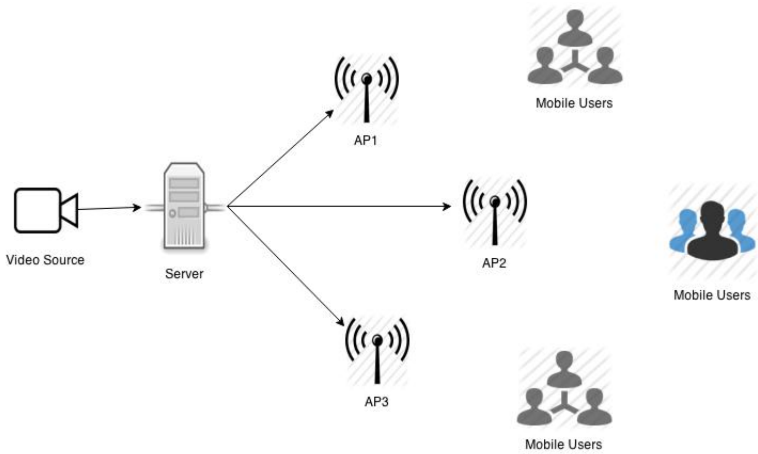

Figure 4.1: Network model

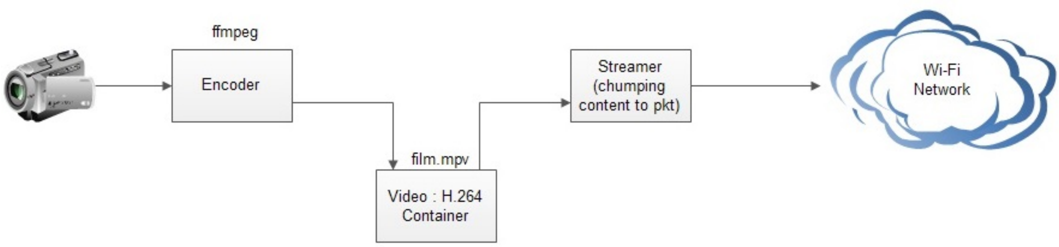

Figure 4.2: Encoding architecture

25 nodes). The nodes are moving around, and while a node moves away from its AP and get closer to the other AP's domain, the handoff will occur. In this project, the main focus is on HMIPv6 as handoff protocol when two different mobility model applied: The random waypoint and Gauss-Markov mobility models, which described in chapter 2.4.1 and 2.4.2 respectively.
The video server received and stored encoded streams, as it described during chapters 2 and 3. According to literature review [?], [?],[53], [34], [65],[23] and [55] the H.264 SVC is chosen as encoding format. Therefore the video streams includes IP header, UDP header and RTP header in addition to video payload.
Figure 4.2 demonstrates the encoding architecture. FFmpeg [1] is used to create video from the sequential images taken by camera and converts them to a mpv file which

[1] is a free tools produce libraries to handle multimedia.

contains h.264 svc video format and mp3 audio format. mpv files are sent to mpv container, and then streamer chumps the mpv content to real-time UDP packets. Eventually, the a packets are transmitted from the video server to APs, as figure 4.2 described.

4.2 QoS parameters

In this project handoff latency, throughput, jitter and loss rate during nodes scale up and hand off are studied. In teletraffic theory and network analysis these parameter are define as follow:

– Throughput: represents the rate of successful packet delivering it also has the same concept as digital bandwidth consumption.

– Jitter: In general jitter is the time or variation in delay between arriving packets and can be caused by different reasons such as network congestion, route changes or queuing. Jitter has a significant effect on real time protocols and applications. In[33] the video jitter is defined as *"Video jittering occurs when the synchronization signals on analog video tapes axe corrupted or when environmental electromagnetic interference randomly delays video signals. It can often result in random displacements (i.e. jitters) of horizontal lines in video image frames."*

– Packet loss rate: It shows the number of packets that failed to arrive at destination. In real time application, Packet loss can cause jitter.

– Handoff latency: when an MN moves from on AP domain to another. This parameter indicates to the time between receiving the last packet from old access router and the first packet from new access router. In delay sensitive application such as video, this parameter plays an important role in the sense of quality of experience.

4.3 Simulation scenarios

In order to study the efficiency of wireless multicast and the effect of increases in the number of nodes and their behavior on handoff process and network QoS, different scenarios are tested. The scenarios that are categorized in two group are presented in follow.

4.3.1 Wireless multicast or unicast?

In order to answer this question several test applied on the assumed scenario. Due to these scenarios, video packet in h.264 SVC format was sent to users in both unicast and multicast manner. The scenario tested while the number of mobile nodes (mobile users) was increased. Then their effect on handoff delay, throughput and jitter during handoff was stored. Moreover, the impact of transition method (multicast and unicast) on the delay, throughput and jitter when MNs are moving inside an AP domain is considered. The result and discussion on these scenarios are described in chapter 6.1.

4.3.2 User mass and behavior on QoS

The second category has focus on the impact of user mass and behavior on QoS during handoff as well as during movement inside AP coverage area.In these case scenarios the number of MNs also increased, moreover two different mobility model used, first Random waypoint and then Gauss-Markov.Both models were tested with HMIPv6 protocol. Chapter 6.2 includes The effect of both models on delay, Throughput and jitter.

Chapter 5
Implementation

Traditionally two approaches are used to model a system: analytical and simulation. Analytical method is preferred in relatively small and simple systems since the model can mathematically be traced by using tools such as queuing and probability theories while in a simulation approach, less abstraction and simplifying assumption is required. Since in a complex and large system, using mathematical formulation usually is not possible, simulation model is preferred to analytical.[31] In this section, the environment of simulation 5.1 and simulation tools 5.2 are described, the chapter ended with an overview on the some implemented features of the project.

5.1 Environment

The simulation of this project performed on a workstation ran Ubuntu 12.04 LTS 32-bit operating system with Intel Core™ i7 2.80GHz $\times$ 4 GHz processor and 32 Gigabytes memory. Version 2.32 of NS2 was running on the workstation. In the rest of this chapter, more details about simulation environment are represented.

5.2 Simulation tools

In order to perform the simulation, NS2 is used. Network animator (NAM) and AWK programming also used to animate the project and process the outputs, and Xgraph makes the plots.

5.2.1 Network simulator NS2

The Network Simulator version, 2.35(NS2)[31], is the tool chosen for simulation. It is a free tool for simulating discrete events. NS2 is now widely used in research and academic investigations. It is useful for simulating both wired and wireless network, and supports their functions and protocols, such as routing protocols, multicast protocols, TCP and UDP.
The basic architecture of NS2 is shown in figure 5.1. NS2 consists of two languages:

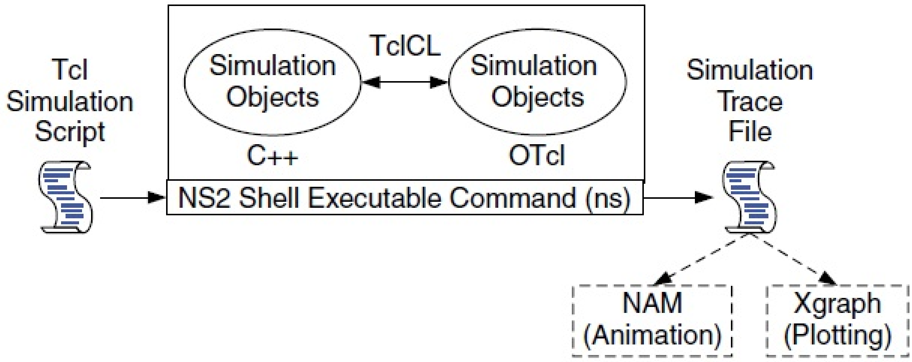

Figure 5.1: NS2 architecture [31]

C++, which characterize internal mechanism of simulation objects and Object-oriented Tool Command Language or Otcl, which used for simulation setup, object configuration and scheduling events. These two are connected together via TclCL. User provides the input of the simulation through TCL script. TCL is a simple text file that saved with .tcl postfix. By running the TCL file, NS2 executes the simulation and usually a trace file are created as output (out.tr).
As mentioned NS2 is a free simulation tool and can be downloaded from [4]. It runs on various platforms such as Linux, Mac or Windows (via CYGWIN)[1]. Its source codes can be found in the all-in-one suite, includes all required components, It requires 320 MB memory. For this work, NS2.32 installed on a Linux Ubuntu 12.04 LTS operating system, the installation and validation are done by following statements:

shell>./install
shell>./validate

the whole procedure with more detail is described in [4]. for executing an NS2 script the following command must be run in NS2 home directory:

»ns file-name.tcl

5.2.2 NAM: Network animator

The Network animator (NAM) [2] works based on Tcl/TK. It is used to show simulation trace file. In fact, NAM records simulation details and then uses them to animate the network. The tool has advantages of using different features that make it

[1] CYGWIN is a collection of different GNU and Open Source tools used on Windows in order to provide Unix-like environment and similar functionality

well suited for visualizing various network models. In [4] few such features are named: dragging and dropping nodes (positioning), shaping the nodes, labeling nodes at a specified instant, monitoring a queue, coloring a particular link, and animating colored packet flows. The following command runs the NAM trace file:

»nam filename.nam

The figures 5.2, 5.3, 5.4 and 5.5 show four samples of NAM for this project.

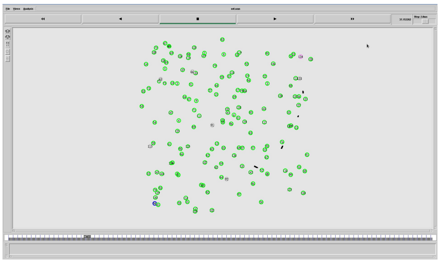

Figure 5.2: Case A

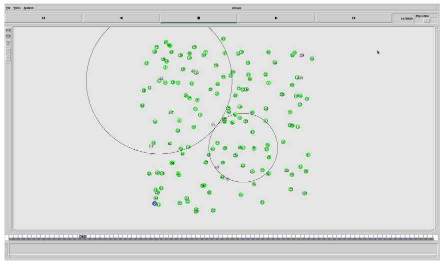

Figure 5.3: Case B

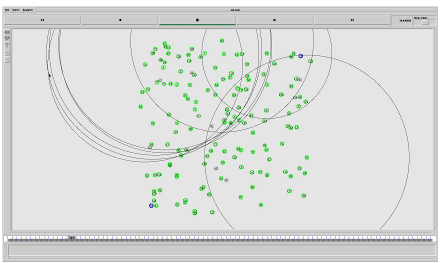

Figure 5.4: Case C

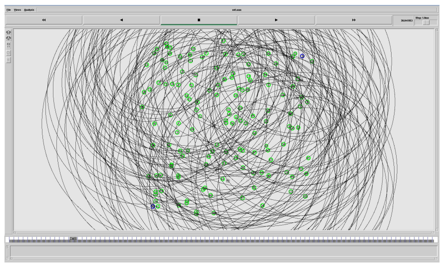

Figure 5.5: Case D

5.2.3 AWK language programming

In this project AWK used to interpret and process The NS2 trace files. AWK [9] proposed initially in 1974 by Alfred V, Weinberger and Kernighan. Its name is derived from the initials of AWK designers. AWK is designed for interpreting and processing text file. In [51] Robbins defines The main functionality of AWK: *"to search files for lines (or other units of text) that contain certain patterns. When a line matches one of the patterns, awk performs specified actions on that line. Awk keeps processing input lines in this way until it reaches the end of the input files."* It is a data-driven language, in the sense that an AWK user must describe the data and the interested action on them. This is explained in the same reference: *"When you run awk, you specify an awk program that tells awk what to do. The program consists* *of a series of rules.Each rule specifies one pattern to search for and one action to perform upon finding the pattern.Syntactically, a rule consists of a pattern followed by an action."* Appendix .1 shows the AWK programming that used in the multicast scenario.

5.2.4 Xgraph

Xgraph is a plotting program that is used for making graphs of simulation results (graphs are illustrated in chapter6.1 and 6.2). Xgraph is included in ns-allinone 2.35 package and can make plots of parameters such as throughput and delay, in addition deviation, automatically take by Ns2 for xgraph command. There are different way to use xgraph, in shell prompt the following command run it:

```
#xgraph filename.xg
```

also inside tcl script the following line does same plotting:

```
exec xgraph filename.xg −geometry 200∗200
```

In this work all xgraphs are define in file graph.sh, for example the following line is used to plot the jitter:

```
xgraph Jitter_AP −x "Nodes" −y "Jitter" −t "Nodes Vs Jitter
(Moving inside AP)" −lw 2 −tk −P −nb −ly 0,0.2
```

where in that:

- x, y and t is used for notation of x and y line and -t for the header name.
- lw width Specifies the width of the data lines in pixels. The default is one.
- tk (Ticks) : This option causes xgraph to draw tick marks rather than full grid lines.
- P (LargePixels) : Similar to -p but marks each pixel with a large dot.
- nb Do not draw buttons.
- ly <yl,yh> This option limits the range of the Y axis to the specified interval.

5.3 simulation

Following methods are used for the simulation of described scenarios. In addition, some changes in existing methods are made. Some part of simulation code that used during this work comes from the internet and ns-allinone tutorials.

The input video file is encoded based on H.264 SVC as described in chapter 4.1. The simulation code can be found in appendix .3

In appendix .2 MAC layer modification is shown. since handoff is done at MAC layer, for simulating the rate and have access to the node location and information (from Physical layer)the applied changes are necessary.

In order to test Random waypoint and Gauss-Markov, different scenarios are defined (Appendix .4 shows a sample scenario).
For run all scenarios at one, inside script.sh file a shell scripting is written which calls the tcl file with different scenario and store the results (Appendix .5).
Appendix .6 and .7 illustrate the tcl script for MultiCast Wired and Wireless Scenario for Random waypoint and Gauss-Markov models respectively.

Chapter 6
Evaluation and Discussion

The NS2 simulation for this work aims to study the parameters described in chapter 4.2 for the scenarios explained in chapter 4.3. All simulation is done for 200 seconds. The numbers of 5, 10, 15, 20 and 25 nodes are used in the simulation, and there are three access points which covered an area of 1000*1000 square meters.

6.1 Multicast VS. Unicast

In this section the results of the impact of increasing in number of users (or mobile nodes) during both multicast and unicast transmission on the QoS parameters such as delay, throughput and jitter are shown.

Delay

Figures 6.1 is the plot fro comparing the delay during handoff for unicast and multicast scenario. As the figure indicates, the top plot (Red) is used for the multicast situation while the below plot or green one belongs to unicast delay. According to this graph in terms of handoff latency, multicast stream has more delay.

In the figure 6.2 the MNs movement are assumed inside the AP domain, This case also unicast streaming has a better result. This behavior can be interpreted by difference in data rate in unicast and multicast transmission. Since unicast has higher data rate than multicast, it offers lower latency. This met our expectation for the small number of MNs. The simulation is done for up to 25 nodes, it is expected for the larger number of MN the result might change.

Throughput

The effect of increasing in number of nodes on throughput during multicast and unicast transmission are indicated in figures 6.3 and 6.4. While the first figure illustrates the changes during handoff, the figure 6.4 indicates on the change when movement takes place inside an AP domain.

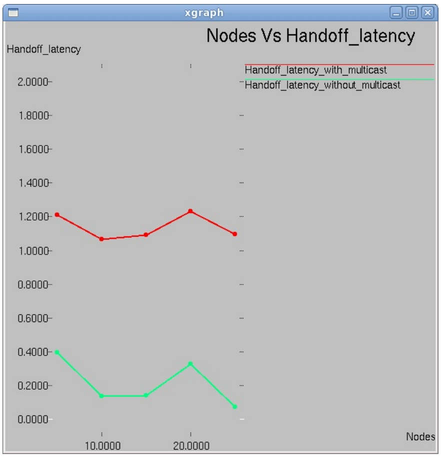

Figure 6.1: Delay during handoff

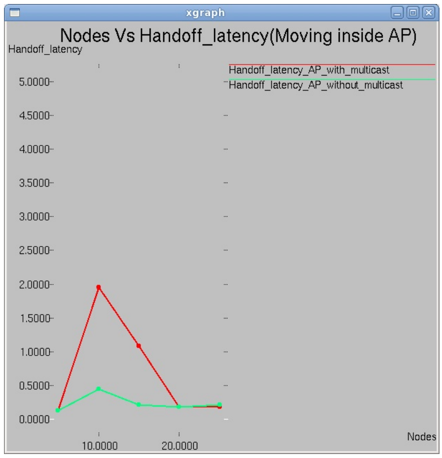

Figure 6.2: Delay inside one AP domain

Based on the simulation result it seems there are significantly higher throughput

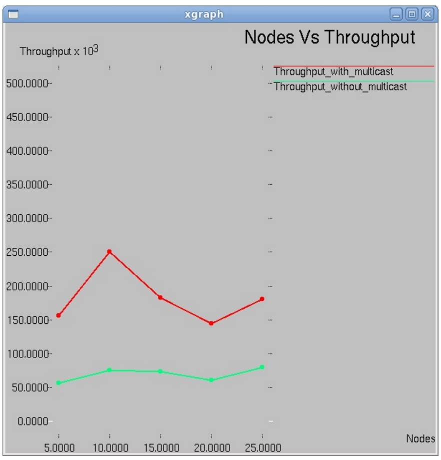

Figure 6.3: Throughput during handoff

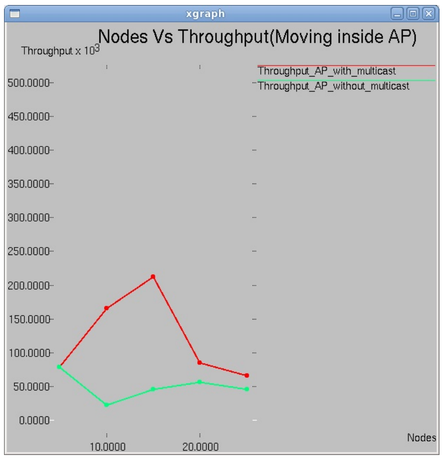

Figure 6.4: Throughput inside one AP domain

with multicast. And this difference seems rather stable even when the number of nodes scale up to 25, except the throughput when moving only inside an AP coverage area. The lack of retransmissions in multicast can be consider as the main reason. It is worthy to mention that the throughput is a QoS measure. It assumes the receiving application can make sense of the incoming data even though some packets are dropped and not retransmitted, and video player is indeed in that category, as they can interpolate and "repair" frames based on earlier received data and heuristics.

Jitter

Jitter is the last parameter tested in order to compare unicast and multicast behavior during MN scaled up, The result shows higher jitter for unicast transmission in both movement inside an AP area or during handoff. In these two scenario, the difference looks stable also. And that is a match with our expectation.

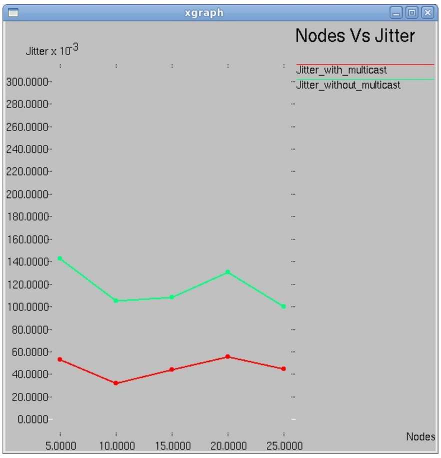

Figure 6.5: Jitter during handoff

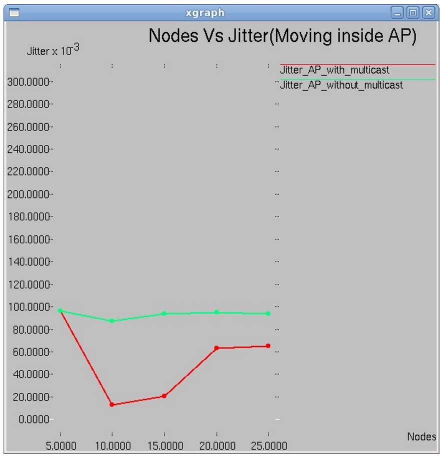

Figure 6.6: Jitter inside one AP domain

6.2 The effect of mobility models on handoff

In this section, the simulation results for two different mobility models are presented. Random waypoint and Gauss-Markov are the chosen models which theoretically are described in chapter 2.4, and a brief explanation of their simulation are described in chapter 5.3. In both scenarios, the hierarchical MIPv6 is applied as mobility management protocol.

Handoff Delay

Figure 6.7 shows the impact of two mobility models on handoff delay. The red plot shows the latency when the Random waypoint model is used, and the green one indicate the handoff latency for Gauss-Markov model.

Throughput

Throughput for Random waypoint and Gauss-Markov are shown in figures 6.8. This simulation is done with using HMIPv6 . The result shows the following semi symmetric plots:

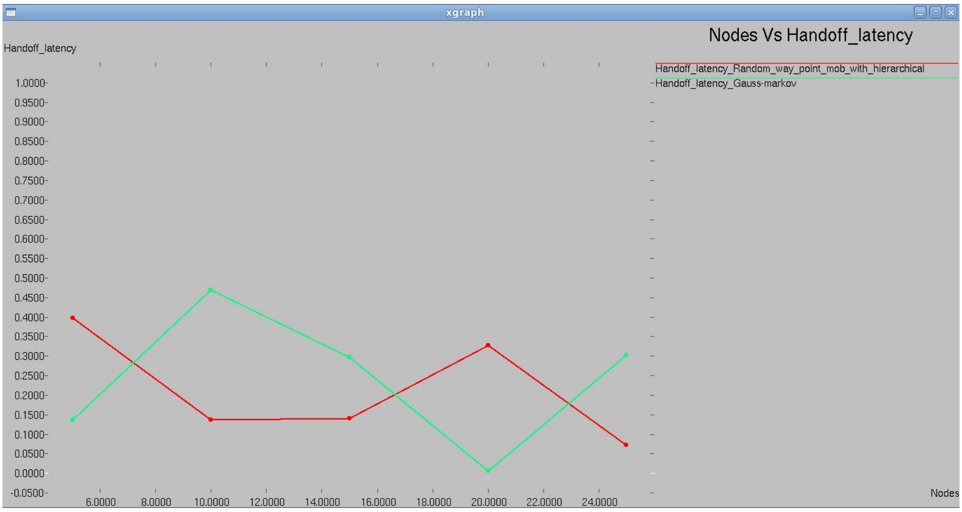

Figure 6.7: Handoff latency for Random waypoint vs. Gauss-markov

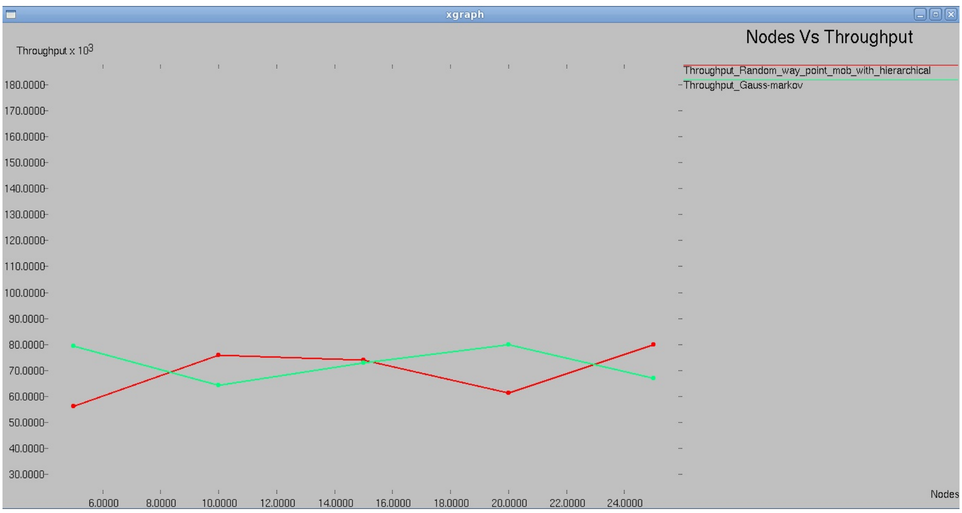

Figure 6.8: Throughput for Random waypoint vs. Gauss-markov

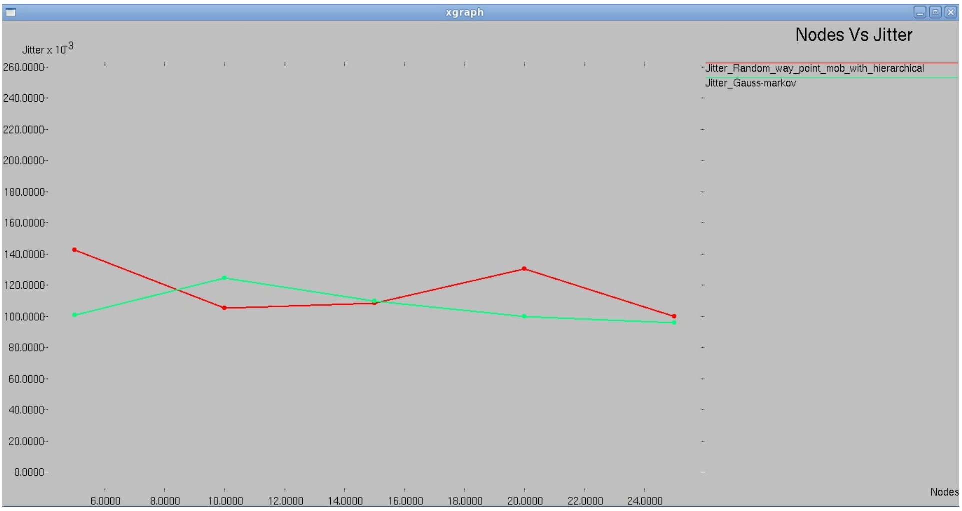

Figure 6.9: Jitter for Random waypoint vs. Gauss-markov

Jitter

The finally tested scenario has focus on jitter and the impact of two suggested mobility model and increasing in number of MNs on the jitter while HMIPv6 manage network mobility. The results show no particular behavior and change, though it seems both mobility models are not different in the assumed scale.

Chapter 7
Conclusion and future work

In today network, the capacity and bandwidth are important concerns that require strong backbones and appropriate infrastructure and these all means expense and cost for the provider and subsequently users. Network demand has migrated from wired network to wireless due to widespread of wirelessly enabled devices. On the other hand; a large fraction of current video traffic belongs to video and multimedia that are highly bandwidth consumption. Therefore, mobility and video streaming (live or on demand) face today's IP networks with a high challenge especially in large scale networks.
Different simulation on NS2 during this work shows that to provide satisfying wireless video services with low degradation and high performance, wireless multicast can be a solution, since according to results (chapter 6.1)less jitter and more throughput is achieved through multicast scenarios.

H.264 SVC is the encoding/decoding platform has been chosen for this project. H.264 SVC is the dominant codec that currently is used for video streaming over the wireless network, due to some improved feature; It can encode high quality video streams, provide different frame rate and quality based on terminal capacity and restriction. Compare with other codecs H.264 SVC has better speed and needs less storage space. It is suitable for video streaming in the networks with varying capacity. It also provides low latency streams that are a necessity in real time video transmission. When there are up to 20 percent network drop, H.264 SVC error resilience mechanism keeps video stream seamless and continues without obvious effect on video quality.

Random waypoint and Gauss-Markov mobility models simulated and tested for different number of users and the effect of these models on QoS parameters such as throughput, jitter and delay studied. As the results described in chapter 6.2, the performance of a network based on chosen mobility model, can not specifically determined since the graph do not follow particular behavior. The random waypoint mobility model a prominent model in different simulation studies, since it is flexible and models realistic mobility model, for example, the people movement pattern in an airport or university campus. Gauss-Markov model also creates practical model of movements. Figures 2.16 and 2.17 demonstrate these models. Moreover, Gauss-Markov avoids the edge effect by forcing mobile node away of the side area.

Hierarchical MIPv6 is The method simulated for handoff management, since it is a localized mobility management protocol and, therefore, significantly reduces the user mobility load as well as the handoff latency and packet loss. Moreover, it improves handoff performance and all these advantages were of interest of this project. The technology area involved in the thesis description was more mature and widespread than first expected. Also due to lack of adequate internet resources for simulating different technologies by NS2 (unlike first expectation), the simulation progress was more complicated, and need more time than expected schedule. Therefore, some parts remain as future works. Such as:

Study and simulate other handover methods such as Fast Handovers for Mobile IPv6 and Proxy mobile(PMIPv6).
In this work only QoS parameter have been tested and due to lack of time QoE features have been missed.study and test these features such as Signal-to-noise ratio (SNR), Peak signal-to-noise ratio (PSNR) and The Mean Opinion Score (MOS)can be considered in future works.
And finally the Reference Point Group Mobility (RPGM) Model is of interest for further study. RPGM is a generic mobility model that is appropriate for group mobility. Based on this model the movement of MN individually as well as movement of a group of MNs are modeled. This model can be tested for wireless multicast group.

Bibliography

[1] Ieee standard for local and metropolitan area networks: Overview and architecture. amendment 2: Registration of object identifiers. *IEEE Std 802-2001 (Revision of IEEE Std 802-1990)*, pages 0–1, 2002.

[2] Nam: Network animator@ONLINE, July 2002.

[3] Dircast: A practical and efficient wi-fi multicast system. In *Proceedings of the 2009 17th IEEE International Conference on Network Protocols. ICNP 2009*, ICNP '09, pages 161–, Washington, DC, USA, 2009. IEEE Computer Society.

[4] Downloading and installing ns-2 @ONLINE, June 2009.

[5] Ieee standard for information technology– local and metropolitan area networks– specific requirements– part 11: Wireless lan medium access control (mac)and physical layer (phy) specifications amendment 5: Enhancements for higher throughput. *IEEE Std 802.11n-2009 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, IEEE Std 802.11y-2008, and IEEE Std 802.11w-2009)*, pages 1–565, Oct 2009.

[6] The zettabyte era—trends and analysis, May 2013.

[7] 9tut. Wireless tutorial @ONLINE, August 2011.

[8] Kushal Adhvaryu and Vijay Raisinghani. A review of handoff optimization techniques in data networks. In Ketan Shah, V.R. Lakshmi Gorty, and Ajay Phirke, editors, *Technology Systems and Management*, volume 145 of *Communications in Computer and Information Science*, pages 195–200. Springer Berlin Heidelberg, 2011.

[9] Alfred V. Aho, Brian W. Kernighan, and Peter J. Weinberger. *The AWK Programming Language*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1987.

[10] O. Alay, Pei Liu, Yao Wang, E. Erkip, and S.S. Panwar. Cooperative layered video multicast using randomized distributed space time codes. *Multimedia, IEEE Transactions on*, 13(5):1127–1140, Oct 2011.

[11] Fan Bai and Ahmed Helmy. A survey of mobility models. *Wireless Adhoc Networks. University of Southern California, USA*, 206, 2004.

[12] J. Bankoski, P. Wilkins, and Yaowu Xu. Technical overview of vp8, an open source video codec for the web. In *Multimedia and Expo (ICME), 2011 IEEE International Conference on*, pages 1–6, July 2011.

[13] Josh Broch, David A. Maltz, David B. Johnson, Yih-Chun Hu, and Jorjeta Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. In *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, MobiCom '98, pages 85–97, New York, NY, USA, 1998. ACM.

[14] Tracy Camp, Jeff Boleng, and Vanessa Davies. A survey of mobility models for ad hoc network research. *Wireless communications and mobile computing*, 2(5):483–502, 2002.

[15] Chih-Ming Chen, Chia-Wen Lin, and Yung-Chang Chen. Adaptive error-resilience transcoding using prioritized intra-refresh for video multicast over wireless networks. *Signal Processing: Image Communication*, 22(3):277 – 297, 2007. Special issue on Mobile Video.

[16] Nakjung Choi, Yongho Seok, Taekyoung Kwon, and Yanghee Choi. Leader-based multicast service in ieee 802.11v networks. In *Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE*, pages 1–5, Jan 2010.

[17] Nakjung Choi, Yongho Seok, Taekyoung Kwon, and Yanghee Choi. Leader-based multicast service in ieee 802.11v networks. In *Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE*, pages 1–5, Jan 2010.

[18] Sungjoon Choi, Nakjung Choi, Yongho Seok, Taekyoung Kwon, and Yanghee Choi. Leader-based rate adaptive multicasting for wireless lans. In *Global Telecommunications Conference, 2007. GLOBECOM '07. IEEE*, pages 3656–3660, Nov 2007.

[19] R. Corvaja, A. Zanella, M. Dossi, A. Tontoli, and P. Zennaro. Experimental performance of the handover procedure in a wifi network. 2004.

[20] A. Dutta, E. van den Berg, D. Famolari, V. Fajardo, Y. Ohba, K. Taniuchi, T. Kodama, and H. Schulzrinne. Dynamic buffering control scheme for mobile handoff. In *Personal, Indoor and Mobile Radio Communications, 2006 IEEE 17th International Symposium on*, pages 1–11, Sept 2006.

[21] A. Fiandrotti, D. Gallucci, E. Masala, and E. Magli. Traffic prioritization of h.264/svc video over 802.11e ad hoc wireless networks. In *Computer Communications and Networks, 2008. ICCCN '08. Proceedings of 17th International Conference on*, pages 1–5, Aug 2008.

[22] NA Fikouras, AJ Könsgen, and C Görg. Accelerating mobile ip hand-offs through link-layer information, an experimental investigation with 802.11 b and internet audio. In *Proceedings of the International Multiconference on Measurement, Modeling, and Evaluation of Computer-Communication Systems (MMB), Aachen, Germany*. Citeseer, 2001.

[23] D.H. Finstad, H.K. Stensland, H. Espeland, and P. Halvorsen. Improved multi-rate video encoding. In *Multimedia (ISM), 2011 IEEE International Symposium on*, pages 293–300, Dec 2011.

[24] R. Flickenger. *Wireless Networking in the Developing World*. Limehouse Book Sprint Team, 2006.

[25] gc. Introduction to wireless lan and ieee 802.11 @ONLINE, February 2013.

[26] N. Golrezaei, A.F. Molisch, A.G. Dimakis, and G. Caire. Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution. *Communications Magazine, IEEE*, 51(4):142–149, April 2013.

[27] Cisco group. Tutorial of wireless standards: Ieee 802.11a, 802.11b, 802.11g, 802.11n and new 802.11ac@ONLINE, August 2013.

[28] Albert Hasson and Neco Ventura. Inter-subnet mobile ip handoff in 802.11 wireless lans.

[29] Alan R. Hevner, Salvatore T. March, Jinsoo Park, and Sudha Ram. Design science in information systems research. *MIS Q.*, 28(1):75–105, March 2004.

[30] R. Hsieh, Zhe Guang Zhou, and A. Seneviratne. S-mip: a seamless handoff architecture for mobile ip. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 3, pages 1774–1784 vol.3, March 2003.

[31] Teerawat Issariyakul and Ekram Hossain. *Introduction to Network Simulator NS2*. Springer Publishing Company, Incorporated, 1st edition, 2010.

[32] Amit Jardosh, Elizabeth M. Belding-Royer, Kevin C. Almeroth, and Subhash Suri. Towards realistic mobility models for mobile ad hoc networks. In *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking*, MobiCom '03, pages 217–229, New York, NY, USA, 2003. ACM.

[33] Sung Ha Kang and Jianhong Shen. Video dejittering by bake and shake. *Image and vision computing*, 24(2):143–152, 2006.

[34] Brent Kelly. Vidyo-google announcement of vp9 svc for webrtc: Why it's important, September 2013.

[35] Young-Bae Ko and Nitin H Vaidya. Location-aided routing (lar) in mobile ad hoc networks. *Wireless Networks*, 6(4):307–321, 2000.

[36] R. Koodli. Mobile ipv6 fast handovers, June 2008.

[37] A. Ksentini, M. Naimi, and A. Gueroui. Toward an improvement of h.264 video transmission over ieee 802.11e through a cross-layer architecture. *Communications Magazine, IEEE*, 44(1):107–114, Jan 2006.

[38] J. Kuri and S.K. Kasera. Reliable multicast in multi-access wireless lans. In *INFOCOM 99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 760–767 vol.2, Mar 1999.

[39] T. Lammle. *CCNA: Cisco Certified Network Associate Study Guide: Exam 640-802*. Cisco certification series. John Wiley & Sons, 2007.

[40] J.-Y. Le Boudec and M. Vojnovic. Perfect simulation and stationarity of a class of mobility models. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 4, pages 2743–2754 vol. 4, March 2005.

[41] Junghoon Lee, Mikyung Kang, Gyungleen Park, Hanil Kim, Choelmin Kim, and Seongbaeg Kim. An error control scheme for multicast video streaming on the last hop wireless lans. In VassilN. Alexandrov, GeertDick Albada, PeterM.A. Sloot, and Jack Dongarra, editors, *Computational Science – ICCS 2006*, volume 3994 of *Lecture Notes in Computer Science*, pages 962–969. Springer Berlin Heidelberg, 2006.

[42] B. Liang and Z.J. Haas. Predictive distance-based mobility management for pcs networks. In *INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, pages 1377–1384 vol.3, Mar 1999.

[43] J. Miroll, Zhao Li, and T. Herfet. Wireless feedback cancellation for leader-based mac layer multicast protocols : Measurement and simulation results on the feasibility of leader-based mac protocols using feedback cancellation on the 802.11aa wireless multicast network. In *Consumer Electronics (ISCE), 2010 IEEE 14th International Symposium on*, pages 1–6, June 2010.

[44] A. Mishra, M. Shin, and W.A. Arbaush. Context caching using neighbor graphs for fast handoffs in a wireless network. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 1, pages –361, March 2004.

[45] A. Mishra, M. Shin, and W.A. Arbaush. Context caching using neighbor graphs for fast handoffs in a wireless network. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 1, pages –361, March 2004.

[46] P. Pace, G. Iannitelli, and G. Aloi. Eavesdropping wireless video packets to improve standard multicast transmission in wi-fi networks. In *Wireless Pervasive Computing (ISWPC), 2010 5th IEEE International Symposium on*, pages 471–476, May 2010.

[47] S. Pack and Y. Choi. Fast handoff scheme based on mobility prediction in public wireless lan systems. *Communications, IEE Proceedings-*, 151(5):489–495, Oct 2004.

[48] Sangheon Pack, Hakyung Jung, Taekyoung Kwon, and Yanghee Choi. A selective neighbor caching scheme for fast handoff in ieee 802.11 wireless networks. In *Communications, 2005. ICC 2005. 2005 IEEE International Conference on*, volume 5, pages 3599–3603 Vol. 5, May 2005.

[49] Yongtae Park, Chiho Jo, Sangki Yun, and Hyogon Kim. Multi-room iptv delivery through pseudo-broadcast over ieee 802.11 links. In *Vehicular Technology Conference (VTC 2010-Spring), 2010 IEEE 71st*, pages 1–5, May 2010.

[50] I. Ramani and S. Savage. Syncscan: practical fast handoff for 802.11 infrastructure networks. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 1, pages 675–684 vol. 1, March 2005.

[51] Arnold Robbins. *Effective AWK Programming: Text Processing and Pattern Matching.* " O'Reilly Media, Inc.", 2001.

[52] Bo Rong and A. Hafid. Cooperative multicast for mobile iptv over wireless mesh networks: The relay-selection study. *Vehicular Technology, IEEE Transactions on*, 59(5):2207–2218, Jun 2010.

[53] T. Schierl, K. Gruuneberg, and T. Wiegand. Scalable video coding over rtp and mpeg2 transport stream in broadcast and iptv channels. *Wireless Communications, IEEE*, 16(5):64,71, October 2009.

[54] Thomas C Schmidt and Matthias Wählisch. Predictive versus reactive—analysis of handover performance and its implications on ipv6 and multicast mobility. *Telecommunication Systems*, 30(1-3):123–142, 2005.

[55] H. Schwarz, D. Marpe, and T. Wiegand. Overview of the scalable video coding extension of the h.264 avc standard. *Circuits and Systems for Video Technology, IEEE Transactions on*, 17(9):1103–1120, Sept 2007.

[56] Ritu Sehgal, Sandeep Makhija, and Panipat Haryana Israna. A performance comparison of fmipv6 and hmipv6, optimization techniques of mipv6.

[57] Sangho Shin, Andrea G. Forte, Anshuman Singh Rawat, and Henning Schulzrinne. Reducing mac layer handoff latency in ieee 802.11 wireless lans. In *Proceedings of the Second International Workshop on Mobility Management &Amp; Wireless Access Protocols*, MobiWac '04, pages 19–26, New York, NY, USA, 2004. ACM.

[58] Michal Skořepa and Richard Klügl. Analytical method for l3 handover latency evaluation. In *Proceedings of the European conference of systems, and European conference of circuits technology and devices, and European conference of communications, and European conference on Computer science*, pages 342–347. World Scientific and Engineering Academy and Society (WSEAS), 2010.

[59] Lars Strand. Linux optimized link state routing protocol (olsr) ipv6 howto@ONLINE, August 2003.

[60] Marc Torrent-Moreno, Xavier Pérez-Costa, and Sebastia Sallent-Ribes. A performance study of fast handovers for mobile ipv6. In *Local Computer Networks, 2003. LCN'03. Proceedings. 28th Annual IEEE International Conference on*, pages 89–98. IEEE, 2003.

[61] K.C.K. Tsang, Cho-Li Wang, and F. C M Lau. Handoff performance comparison of mobile ip, fast handoff and msctp in mobile wireless networks. In *Parallel Architectures, Algorithms, and Networks, 2008. I-SPAN 2008. International Symposium on*, pages 45–52, May 2008.

[62] Vasos Vassiliou and Zinon Zinonos. An analysis of the handover latency components in mobile ipv6. *Journal of internet engineering*, 3(1), 2010.

[63] Ganesh Venkatesan. Ieee 802 tutorial:video over 802.11, March 2007.

[64] Ju Wang, Hongsik Choi, E.A. Hughes, and Yong Tang. Efficient wireless multicast protocol with orthogonal cts modulation supporting video conferencing. In *Multimedia and Expo, 2006 IEEE International Conference on*, pages 1217–1220, July 2006.

[65] Stephan Wenger, Ye-kui Wang, and MiskaM. Hannuksela. Rtp payload format for h.264/svc scalable video coding. *Journal of Zhejiang University SCIENCE A*, 7(5):657–667, 2006.

[66] T. Wiegand, G.J. Sullivan, G. Bjontegaard, and A. Luthra. Overview of the h.264/avc video coding standard. *Circuits and Systems for Video Technology, IEEE Transactions on*, 13(7):560–576, July 2003.

[67] Jiang Xie, I. Howitt, and I. Shibeika. Ieee 802.11-based mobile ip fast handoff latency analysis. In *Communications, 2007. ICC '07. IEEE International Conference on*, pages 6055–6060, June 2007.

[68] Kang Jin Yoon and Young Yong Kim. Relayed multicast scheme in ieee 802.11 wireless lan systems. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pages 1–2, April 2006.

[69] Yohaan Yoon, Myungchul Kim, Sooyong Lee, B. Lee, Soon-Joo Hyun, and Kyunghee Lee. Performance analysis of h.264/avc, h.264/svc, and vp8 over ieee 802.11 wireless networks. In *Computers and Communications (ISCC), 2012 IEEE Symposium on*, pages 000151–000156, July 2012.

[70] M. Zaidi, J. Bhar, R. Ouni, and R. Tourki. Reducing wi-fi handover delay using a new positioning process. In *Communications, Computing and Control Applications (CCCA), 2011 International Conference on*, pages 1–6, March 2011.

[71] Li Jun Zhang and Samuel Pierre. Evaluating the performance of fast handover for hierarchical mipv6 in cellular networks. *Journal of Networks*, 3(6):36–43, 2008.

Appendix

.1 AWK for multicast scenario

An example of AWK programing using in this project. Awk files are written to get the result from multicast trace file.

```
BEGIN {
send = 0;
recv = 0;
bytes = 0;
st = 0;
ft = 0;
rtr = 0;
delay = 0;
last_pkt_recv=0;
jitter=0;
j_count=0;
}
{
 if( $1 == "+" && $3 == $9 )
 { if(send == 0)
                 st = $2;
        ft = $2;
        st_time[$12] = $2;
        send++;
 }
 if ( $19 == "AGT" && $1 == "r")
 {
if(recv == 0)
{
        last_pkt_recv = $3;
}
else
```

```
{    jitter+=$3 - last_pkt_recv;
         j_count++;
         last_pkt_recv    =         $3
}
         recv++;
         bytes+=$37
         ft_time[$41] = $3;
         delay += ft_time[$41]-st_time[$41]
 }
}
END {
         send=send*3;
         print "No_of_pkts_send: \t\t" send
         print "No_of_pkts_recv: \t\t" recv
         print "Pkt_delivery_ratio: \t\t" recv/send*100
         print "Delay: \t\t\t\t" delay/recv
         print "Throughput: \t\t\t" bytes*8/(ft-st)
         print "Jitter: \t\t\t" jitter/j_count
         print "Dropping_Ratio: \t\t" (send-recv)/send*100
}
```

.2 MAC modification

In order to simulate interested scenarios, MAC layer need some modification, here are the modified code:

```
void Mac802_11::intra_region_handover(nsaddr_t new_bs,nsaddr_t old_bs)
{
handover_req_to_old_bs(old_bs);
initate_handover_to_new_bs(new_bs);
}

void Mac802_11::inter_region_handover(nsaddr_t new_bs,nsaddr_t old_bs)
{
sendHandover(new_bs,old_bs);
}

void Mac802_11::handover_req_to_old_bs(nsaddr_t nid)
{
Packet *p = Packet::alloc();
hdr_cmn* ch = HDR_CMN(p);
hdr_ip * ih = HDR_IP(p);
```

```
struct rts_frame *rf = (struct rts_frame*)p->access(hdr_mac::offset_);

ch->uid() = 0;
ch->ptype() = PT_MAC;
ch->size() = phymib_.getRTSlen();
ch->iface() = -2;
ch->error() = 0;

bzero(rf, MAC_HDR_LEN);

rf->rf_fc.fc_protocol_version = MAC_ProtocolVersion;
rf->rf_fc.fc_type = MAC_Type_Control;
rf->rf_fc.fc_subtype = MAC_Subtype_RA;
ch->X = ((MobileNode*)netif_->node())->X();
ch->Y = ((MobileNode*)netif_->node())->Y();
ch->ra_flag = 2;
ch->to_dst = nid;
ih->saddr() = netif_->node()->nodeid();

ch->txtime() = txtime(ch->size(), basicRate_);
D_RM_ += ch->txtime();
reg_count++;

cout<<"MS "<<ih->saddr()<<" send Handover Req"<<endl;
ch->address_ = netif_->node()->address();
downtarget_->recv(p, this);
}

void Mac802_11::initate_handover_to_new_bs(nsaddr_t nid)
{
Packet *p = Packet::alloc();
hdr_cmn* ch = HDR_CMN(p);
hdr_ip * ih = HDR_IP(p);
struct rts_frame *rf = (struct rts_frame*)p->access(hdr_mac::offset_);

ch->uid() = 0;
ch->ptype() = PT_MAC;
ch->size() = phymib_.getRTSlen();
ch->iface() = -2;
ch->error() = 0;
```

```
bzero(rf, MAC_HDR_LEN);

rf->rf_fc.fc_protocol_version = MAC_ProtocolVersion;
rf->rf_fc.fc_type = MAC_Type_Control;
rf->rf_fc.fc_subtype = MAC_Subtype_RA;
ch->X = ((MobileNode*)netif_->node())->X();
ch->Y = ((MobileNode*)netif_->node())->Y();
ch->ra_flag = 3;
ch->to_dst = nid;
ih->saddr() = netif_->node()->nodeid();

ch->txtime() = txtime(ch->size(), basicRate_);
D_RM_ += ch->txtime();
reg_count++;

cout<<"MS "<<ih->saddr()<<" Initate New Handover Req"<<endl;
ch->address_ = netif_->node()->address();
downtarget_->recv(p, this);
}

void Mac802_11::recvHandOver(Packet *p)
{
hdr_ip * ih = HDR_IP(p);
if(members_.check(ih->saddr()) == -1)
{
members_.remove(ih->saddr());
netif_->node()->mem_list.remove(ih->saddr());
}
cout<<"Recv handover "<<endl;
}

void Mac802_11::recvNewHandOverReq(Packet *p)
{
hdr_ip * ih = HDR_IP(p);
members_.add(ih->saddr());
cout<<"Recv new handover Join "<<endl;
}

void Mac802_11::sendHandover(nsaddr_t nid,nsaddr_t old)
{
Packet *p = Packet::alloc();
```

```
hdr_cmn* ch = HDR_CMN(p);
hdr_ip * ih = HDR_IP(p);
struct rts_frame *rf = (struct rts_frame*)p->access(hdr_mac::offset_);

ch->uid() = 0;
ch->ptype() = PT_MAC;
ch->size() = phymib_.getRTSlen();
ch->iface() = -2;
ch->error() = 0;

bzero(rf, MAC_HDR_LEN);

rf->rf_fc.fc_protocol_version = MAC_ProtocolVersion;
rf->rf_fc.fc_type = MAC_Type_Control;
rf->rf_fc.fc_subtype = MAC_Subtype_RA;
ch->X = ((MobileNode*)netif_->node())->X();
ch->Y = ((MobileNode*)netif_->node())->Y();
ch->ra_flag = 4;
ch->to_dst = nid;
ih->saddr() = netif_->node()->nodeid();

ch->txtime() = txtime(ch->size(), basicRate_);
D_RM_ += ch->txtime();
reg_count++;
cout<<"MS "<<ih->saddr()<<" Send Handover"<<endl;
ch->address_ = netif_->node()->address();
downtarget_->recv(p, this);
}

void Mac802_11::recvHandover(Packet *p)
{
hdr_ip * ih = HDR_IP(p);
members_.add(ih->saddr());
netif_->node()->mem_list.add(ih->saddr());
cout<<"InterHandover"<<endl;
}

double Mac802_11::Transition_prob()
{
return Random::uniform(0,1);
```

.3 H.265 SVC simulation

The codes below shows the simulation of H.264 SVC video codec:

```
if (running_)
        {
        Packet *p;
        p = agent_->allocpkt();

        HDR_CMN(p)->frame_id    =    frame_id;

        FILE    *fp;
        fp = fopen(file_name,"r+");

        int i_=0;

        H_264 * stream = new H_264();

        while(!feof(fp))
        {
            fread(&stream->stream_data, pkt_size, 1, fp);
            if(feof(fp))
            {
                running_ = 0;
            }
            if(i_ == frame_id)
            {
                goto out;
            }
            i_++;
        }

        if(feof(fp))
        {
            cout<<"Wrong end"<<endl;
            running_ = 0;
            return;
        }
        out:

        fclose(fp);
```

```
if(i_ == frame_id)
{
    stream->stream_data[pkt_size] = '\0';
    p->mpeg_data    =    stream;
    frame_id++;

      if(file_create_ == 0)
     {
        int     total_frames = 0;

        {
            FILE *fp1 = fopen(file_name,"r");

            char     _bit[pkt_size];

             while(!feof(fp1))
            {
                fread(_bit, pkt_size, 1, fp1);
                total_frames++;
            }
            fclose(fp1);
        }

        cout<<"total_frames "<<total_frames<<endl;

        if(modelling == 1)
        {
        ifstream in_stream(file_name);
        FILE *meta = fopen("meta","w");

        for(int i=0;i<total_frames;i++)
        {
            unsigned char a[4];
            a[0] = in_stream.get();
            a[1] = in_stream.get();
            a[2] = in_stream.get();
            a[3] = in_stream.get();

            int     ptype = 0;
            int     size = pkt_size;
```

```
int      total_macro_block =     sizeof(in_stream);

if (a[3]==GOP_START_CODE)
{
    ptype = 4;
}
else if(a[3]==SEQ_START_CODE)
{
    ptype = 5;
}
else if(a[3]==PICTURE_START_CODE)
{
    a[2]=in_stream.get ();
    a[3]=in_stream.get ();
    a[3]=a[3]& 0x38;
    a[3]=a[3]>>3;

    if(a[3] == 0x01)     //     I FRAME
    {
        ptype = 1;
    }

    if(a[3] == 0x02)     //     P FRAME
    {
        ptype = 2;
    }

    if(a[3] == 0x03)     //     B FRAME
    {
        ptype = 3;
    }
}

if(ptype == 1) //  I slice
{
    int j;

    for (j = 0; j < LUMA_WIDTH/16; j++)
        macroblock(i, j, meta);

    int cost_of_mode = get_cost_of_mode(i,j);
```

```
                    int     _cost = 0;

                    if(cost_of_mode == _4_4)
                    {
                        _cost = get_cost(cost_of_mode,total_frames-i
                    }
                    else if(cost_of_mode == _16_16)
                    {
                        _cost = get_cost(cost_of_mode,total_frames-i
                    }

                    int     code[1000][1000];

                    if(_cost % (16*16))     //     is 16 * 16 chosen
                    {
                        intra_16_16_coding(i,j,code);
                        goto store_mode;
                    }
                    else
                        goto store_mode;
                }
                else if(ptype == 2)
                {
                    int i_,j;

                    for (i_ = 0; i_ < LUMA_HEIGHT/16 ; i_++)
//     copy search area
                        for (j = 0; j < LUMA_WIDTH/16; j++)
                            macroblock(i_, j, meta);

                    int mot_search = motion_search(i);
                    luma_residual_coding(mot_search,meta);
                    goto store_mode;
                }

                store_mode:

                fwrite(sps, 1, sizeof(sps), meta);     //
store_mode
                fwrite(pps, 1, sizeof(pps), meta);     //
store_reference
```

```
        chroma_residual_coding(i,meta);
        FILE *fp = fopen("slice","a");

        int     init = 0;

        if(i < total_macro_block)
        {
            _trec t;            //    encode and write
            t.trec_time = (htonl(4444));
            t.trec_len = (htonl(size));
            t.trec_ptype = (htonl(ptype));
            fwrite((char *)&t, sizeof(trec), 1, meta);
            init = 1;
        }
        else
        {
            fwrite(slice_header, 1, sizeof(slice_header), fp)
            int i,j;

            for (i = 0; i < LUMA_HEIGHT/16 ; i++)
                for (j = 0; j < LUMA_WIDTH/16; j++)
                    deblock_frame(i, j, fp);

            fputc(0x80, fp);
            fclose(fp);
        }

        if(init == 1)
        {
            fclose(fp);
        }
    }

    fclose(meta);
    }

    cout<<"Sending frame "<<HDR_CMN(p)->frame_id<<endl;
     file_create_    =    1;
     FILE    *fp1;
    fp1 = fopen("stream_file_send","w");
    fwrite(&p->mpeg_data->stream_data,pkt_size,1,fp1);
```

```
                fclose(fp1);
                agent_->Attach_Video(p);
                double t = next();
                timer_.resched(t);
            }
            else
            {
                if(HDR_CMN(p)->frame_id > playframe && mapp_start == 0)
                {
                    mapp_start = 1;
                    Tcl& tcl = Tcl::instance();
                    sprintf (tcl.buffer(), "exec mplayer ./stream_fil
                    tcl.eval();
                }

                FILE *fp1;
                fp1 = fopen("stream_file_send","a");
                if(fp1)
                {
                    cout<<"Sending frame "<<HDR_CMN(p)->frame_id<<endl;
                    fwrite(&p->mpeg_data->stream_data,pkt_size,1,fp1);
                    fclose(fp1);
                    agent_->Attach_Video(p);
                    double t = next();
                    timer_.resched(t);
                }
            }
        }
        else
        {
            running_ = 0;
        }
```

h264.cc

```
if(ih->daddr() == index)
{
    if(written_ == NULL)
        written_ = new recv_buffer();


    if(written_->check(HDR_CMN(p)->frame_id) != -1)
```

```
        goto down;

    if(recv_buffer_ == NULL)
        recv_buffer_ = new recv_buffer();

   cout<<"Recv frame: "<<HDR_CMN(p)->frame_id<<endl;

    if(recv_buffer_->count == 0 && (HDR_CMN(p)->frame_id != next_id) ==
    {
        cout<<"Just hold to write "<<HDR_CMN(p)->frame_id<<" "<<next_id<
       recv_buffer_->add(HDR_CMN(p)->frame_id,p->mpeg_data);
       req_pkt(next_id,ih->saddr());
    }
    else if(recv_buffer_->count != 0)
    {
        if(HDR_CMN(p)->frame_id == next_id)
        {
           recv_buffer_->add(HDR_CMN(p)->frame_id,p->mpeg_data);
           cout<<"buffer ";
           for(int i=0;i<recv_buffer_->count;i++)
           {
               cout<<recv_buffer_->pid[i]<<" ";
           }
           cout<<endl;

           if(buffer_is_in_order() == 1)
           {
               cout<<"Just write from buffer"<<endl;
               for(int i=0;i<recv_buffer_->count;i++)
               {
                   FILE    *fp;
                   fp = fopen("./stream_file_recv","a");

                   written_->add(recv_buffer_->pid[i],recv_buffer_->data
                   fwrite(recv_buffer_->data[i],1,pkt_size,fp);
                   next_id = recv_buffer_->pid[i] + 1;
                   fclose(fp);
                }
                recv_buffer_->count = 0;
           }
           else
```

```
        {
            cout<<"Buffer is not in order to write"<<endl;
            int unsort = get_unsorted_point();
            for(int i=0;i<unsort;i++)
            {
                 FILE    *fp;
                fp = fopen("./stream_file_recv","a");
                written_->add(recv_buffer_->pid[i],recv_buffer_->dat
                fwrite(recv_buffer_->data[i],1,pkt_size,fp);
                next_id = recv_buffer_->pid[i] + 1;
                fclose(fp);
             }

             int     array[unsort+1];
             for(int i=0;i<unsort;i++)
             {
                 array[i] = recv_buffer_->pid[i];
             }

             for(int i=0;i<unsort;i++)
             {
                 recv_buffer_->remove(array[i]);
             }
         }
    }
     else
     {
         cout<<"Just wait to write "<<HDR_CMN(p)->frame_id<<" "<<nex
        recv_buffer_->add(HDR_CMN(p)->frame_id,p->mpeg_data);
        req_pkt(next_id,ih->saddr());
    }
}
 else
 {
     if(ch->frame_id > playframe && mapp_start == 0)
    {
        mapp_start    =    1;
        Tcl& tcl = Tcl::instance();
        sprintf(tcl.buffer(), "exec mplayer ./stream_file_recv &");
        tcl.eval();
     }
```

```
        next_id = HDR_CMN(p)->frame_id + 1;
         FILE     *fp;
        fp = fopen("./stream_file_recv","a");
        written_->add(HDR_CMN(p)->frame_id,p->mpeg_data);
        fwrite(&p->mpeg_data->stream_data,1,pkt_size,fp);
        fclose(fp);
    }

    down:

    Tcl& tcl=Tcl::instance();
   double now = Scheduler::instance().clock();
   now += 0.00000001;
   tcl.evalf("[Simulator instance] at %.9f {[Simulator instance] puts-na
   dmux_->recv(p,0);
   return;
}
```

mipv6.cc

.4 A scenario file

The figure .1 is a sample of scenario files. It used for Random Waypoint model while there are 5 mobile nodes.

The figure .2 is the same scenario for Gauss-Markov model.

.5 script.sh file

Here the content of inside script.sh file is shown. It is used to run all scenarios at one.

```
rm -rf result
for i in 5 10 15 20 25
{
        echo "Running $i"
        ns wired_wireless.tcl $i &> o
        gawk -f wired_wireless.awk out.tr >> result
}

grep "Delay:" result | gawk 'BEGIN{val=5}{print val " " $2;val+=5}' > gra
grep "Throughput:" result | gawk 'BEGIN{val=5}{print val " " $2;val+=5}'
grep "Jitter:" result | gawk 'BEGIN{val=5}{print val " " $2;val+=5}' > gr
```

```
#
# nodes: 5, pause: 1.00, max speed: 50.00, max x: 750.00, max y: 750.00
#
$node_(0) set X_ 631.734789151427
$node_(0) set Y_ 670.139498400738
$node_(0) set Z_ 0.000000000000
$node_(1) set X_ 223.271695242848
$node_(1) set Y_ 210.499086017676
$node_(1) set Z_ 0.000000000000
$node_(2) set X_ 748.201501585057
$node_(2) set Y_ 258.267183955660
$node_(2) set Z_ 0.000000000000
$node_(3) set X_ 508.280482718800
$node_(3) set Y_ 170.980495823068
$node_(3) set Z_ 0.000000000000
$node_(4) set X_ 608.025832560233
$node_(4) set Y_ 712.692723432576
$node_(4) set Z_ 0.000000000000
$ns_ at 1.000000000000 "$node_(0) setdest 684.525929654807 281.076822363301 21.24670442876
$ns_ at 1.000000000000 "$node_(1) setdest 449.197593071288 624.092587167690 22.39053679100
$ns_ at 1.000000000000 "$node_(2) setdest 193.713939933223 196.503323589613 37.67949230674
$ns_ at 1.000000000000 "$node_(3) setdest 437.077932039325 349.719297313507 35.33490503897
$ns_ at 1.000000000000 "$node_(4) setdest 659.964108786670 485.041660812838 49.63502172010
$ns_ at 5.704354340481 "$node_(4) setdest 659.964108786670 485.041660812838 0.00000000000
$ns_ at 6.445011567822 "$node_(3) setdest 437.077932039325 349.719297313507 0.00000000000
$ns_ at 6.704354340481 "$node_(4) setdest 630.131977276749 457.515718744432 31.69405737248
$ns_ at 7.445011567822 "$node_(3) setdest 695.593505325599 126.848115791662 13.90105341361
$ns_ at 7.985069276940 "$node_(4) setdest 630.131977276749 457.515718744432 0.00000000000
$ns_ at 8.985069276940 "$node_(4) setdest 157.871510853822 256.472233879898 47.60671930904
$ns_ at 15.806910412961 "$node_(2) setdest 193.713939933223 196.503323589613 0.0000000000
$ns_ at 16.806910412961 "$node_(2) setdest 214.819260740747 728.473248081849 19.1381919502
$ns_ at 19.479473113320 "$node_(0) setdest 684.525929654807 281.076822363301 0.0000000000
$ns_ at 19.766578380053 "$node_(4) setdest 157.871510853822 256.472233879898 0.0000000000
$ns_ at 20.479473113320 "$node_(0) setdest 136.016790066262 653.474038914452 31.3175697145
$ns_ at 20.766578380053 "$node_(4) setdest 582.161312827271 665.611983672085 9.92562041257
$ns_ at 22.048047857891 "$node_(1) setdest 449.197593071288 624.092587167690 0.0000000000
$ns_ at 23.048047857891 "$node_(1) setdest 705.462183450846 682.785815563856 24.0053239728
$ns_ at 31.998812308935 "$node_(3) setdest 695.593505325599 126.848115791662 0.0000000000
$ns_ at 32.998812308935 "$node_(3) setdest 517.189501336640 738.221781223577 18.1937168548
$ns_ at 33.999787084353 "$node_(1) setdest 705.462183450846 682.785815563856 0.0000000000
$ns_ at 34.999787084353 "$node_(1) setdest 142.147166611769 74.752622819167 42.2348230809
$ns_ at 41.649047122940 "$node_(0) setdest 136.016790066262 653.474038914452 0.0000000000
$ns_ at 42.649047122940 "$node_(0) setdest 147.931522461529 199.068171415985 32.8702663856
$ns_ at 44.625025438569 "$node_(2) setdest 214.819260740747 728.473248081849 0.0000000000
$ns_ at 45.625025438569 "$node_(2) setdest 272.343276626757 557.467660855798 21.2332135337
$ns_ at 54.122162023344 "$node_(2) setdest 272.343276626757 557.467660855798 0.0000000000
$ns_ at 54.625100157941 "$node_(1) setdest 142.147166611769 74.752622819167 0.00000000000
$ns_ at 55.122162023344 "$node_(2) setdest 142.801840604000 97.136941960660 39.40840555516
$ns_ at 55.625100157941 "$node_(1) setdest 464.277288776281 744.812584114498 44.9858436260
$ns_ at 56.478020707422 "$node_(0) setdest 147.931522461529 199.068171415985 0.0000000000
$ns_ at 57.478020707422 "$node_(0) setdest 606.451007861600 103.441348128440 39.3558077132
$ns_ at 67.256897459687 "$node_(2) setdest 142.801840604000 97.136941960660 0.00000000000
$ns_ at 68.003855558159 "$node_(3) setdest 517.189501336640 738.221781223577 0.0000000000
```

Figure .1: A sample of scenario file for Random waypoint model

```
#
# nodes: 5, pause: 1.00, max speed: 50.00, max x: 750.00, max y: 750.00
#
$node_(0) set X_ 357.382343519835
$node_(0) set Y_ 558.894756250109
$node_(0) set Z_ 0.000000000000
$node_(1) set X_ 595.919216738827
$node_(1) set Y_ 522.621980334020
$node_(1) set Z_ 0.000000000000
$node_(2) set X_ 192.979200950282
$node_(2) set Y_ 248.508720353668
$node_(2) set Z_ 0.000000000000
$node_(3) set X_ 125.281498024369
$node_(3) set Y_ 278.060251401396
$node_(3) set Z_ 0.000000000000
$node_(4) set X_ 14.666569663828
$node_(4) set Y_ 251.382541909713
$node_(4) set Z_ 0.000000000000
$god_ set-dist 0 1 1
$god_ set-dist 0 2 16777215
$god_ set-dist 0 3 16777215
$god_ set-dist 0 4 16777215
$god_ set-dist 1 2 16777215
$god_ set-dist 1 3 16777215
$god_ set-dist 1 4 16777215
$god_ set-dist 2 3 1
$god_ set-dist 2 4 1
$god_ set-dist 3 4 1
$ns_ at 1.000000000000 "$node_(0) setdest 694.025424566886 207.523436382151 39.832225613087"
$ns_ at 1.000000000000 "$node_(1) setdest 725.048971027179 698.902573814088 21.716388633244"
$ns_ at 1.000000000000 "$node_(2) setdest 653.241890569756 728.774074426621 25.298973490062"
$ns_ at 1.000000000000 "$node_(3) setdest 555.448320585594 154.286984899475 46.037518460258"
$ns_ at 1.000000000000 "$node_(4) setdest 729.526066859584 353.913803902471 11.685116270209"
$ns_ at 4.359220999725 "$god_ set-dist 0 2 1"
$ns_ at 4.359220999725 "$god_ set-dist 0 3 2"
$ns_ at 4.359220999725 "$god_ set-dist 0 4 2"
$ns_ at 4.359220999725 "$god_ set-dist 1 2 2"
$ns_ at 4.359220999725 "$god_ set-dist 1 3 3"
$ns_ at 4.359220999725 "$god_ set-dist 1 4 3"
$ns_ at 5.195883854694 "$god_ set-dist 3 4 2"
$ns_ at 5.781303134455 "$god_ set-dist 0 1 16777215"
$ns_ at 5.781303134455 "$god_ set-dist 1 2 16777215"
$ns_ at 5.781303134455 "$god_ set-dist 1 3 16777215"
$ns_ at 5.781303134455 "$god_ set-dist 1 4 16777215"
$ns_ at 5.988879063649 "$god_ set-dist 0 3 1"
$ns_ at 8.201986182592 "$god_ set-dist 0 4 16777215"
$ns_ at 8.201986182592 "$god_ set-dist 2 4 16777215"
$ns_ at 8.201986182592 "$god_ set-dist 3 4 16777215"
$ns_ at 8.684103024854 "$god_ set-dist 2 3 2"
$ns_ at 8.835140879488 "$god_ set-dist 0 2 16777215"
$ns_ at 8.835140879488 "$god_ set-dist 2 3 16777215"
$ns_ at 10.722930918520 "$node_(3) setdest 555.448320585594 154.286984899475 0.000000000000"
$ns_ at 11.062272775469 "$node_(1) setdest 725.048971027179 698.902573814089 0.000000000000"
```

Figure .2: A sample of scenario file for Gauss-Markov model

```
rm -rf result
for i in 5 10 15 20 25
{
        echo "Running $i"
        ns wired_wireless_gauss_markov.tcl $i &> o
        gawk -f wired_wireless.awk out.tr >> result
}

grep "Delay:" result | gawk 'BEGIN{val=5}{print val " " $2;val+=5}' > gr
grep "Throughput:" result | gawk 'BEGIN{val=5}{print val " " $2;val+=5}
grep "Jitter:" result | gawk 'BEGIN{val=5}{print val " " $2;val+=5}' > g

rm -rf result
for i in 5 10 15 20 25
{
        echo "Running $i"
        ns wired_wireless_1.tcl $i &> o
        gawk -f wired_wireless.awk out.tr >> result
}

grep "Delay:" result | gawk 'BEGIN{val=5}{print val " " $2;val+=5}' > gr
grep "Throughput:" result | gawk 'BEGIN{val=5}{print val " " $2;val+=5}
grep "Jitter:" result | gawk 'BEGIN{val=5}{print val " " $2;val+=5}' > g
```

.6 tcl script for Random waypoint model

This script is used for multicast Wired and Wireless Scenario using Random waypoint:

```
set val(x)              1000    ;       # X topography
set val(y)              1000    ;       # Y topography
set val(stop)           200     ;       # Simulation Time
set val(routing)        MIPV6   ;       # Routing protocol
set val(energymodel)    EnergyModel;
set val(initialenergy)  100     ;# Initial energy in Joules
set val(nn)             5
set wired_nodes         1
set val(bs)             3

set ns_ [new Simulator]

if { $argc == 1 } {
```

```
set     val(nn)         [lindex $argv 0]
}

set     playframe       5000

Application/Video       set     modelling       0
Agent/MIPV6             set     playframe       $playframe
Mac/802_11              set     dataRate_       11Mb

Application/Video       set     playframe       $playframe
Application/Video       set     pkt_size        512
Application/Video       set     frame_id        0
Application/Video       set     interval_       0.1

Agent/MIPV6     set     multicast       0

$ns_ color 0 brown
$ns_ node-config -addressType hierarchical

AddrParams set domain_num_ [expr $val(bs) + 1]
lappend cluster_num 1 1 1 1
AddrParams set cluster_num_ $cluster_num
lappend eilastlevel 6 6 6 6
AddrParams set nodes_num_ $eilastlevel

set topo [new Topography]
$topo load_flatgrid $val(x) $val(y)

set t [open out.tr w]
$ns_ trace-all $t
$ns_ use-newtrace

set nt [open out.nam w]
$ns_ namtrace-all-wireless $nt $val(x) $val(y)

set god_ [create-god [expr $val(nn) + $val(bs)]]

for {set i 0} {$i < $wired_nodes} { incr i} {
        set W($i) [$ns_ node 0.0.$i]
}
```

```
$ns_ node-config -adhocRouting $val(routing)\
                 -llType LL \
                 -macType Mac/802_11 \
                 -ifqType Queue/DropTail/PriQueue \
                 -ifqLen 50 \
                 -antType Antenna/DirAntenna \
                 -propType Propagation/TwoRayGround \
                 -phyType Phy/WirelessPhy \
                 -channelType Channel/WirelessChannel \
                 -topoInstance $topo \
                 -energyModel $val(energymodel) \
                 -rxPower 0.6 \
                 -txPower 0.9 \
                 -initialEnergy $val(initialenergy) \
                 -agentTrace ON \
                 -routerTrace ON \
                 -wiredRouting  ON \
                 -macTrace OFF

for { set i 0 } { $i < $val(bs) } { incr i } {
        set BS($i) [$ns_ node [expr $i + 1].0.0]
        $BS($i) label "AP"
        set     mac($i) [$BS($i) getMac 0]
        set     ragent_bs($i)   [$BS($i) set ragent_]
#       $ns_ at 0.0 "$mac($i)   bs_exec"
}

$ragent_bs(0)   set     W_AP    1

for { set i 1 } { $i < $val(bs) } { incr i } {
        $ns_ at 0.0 "$mac($i) bs_exec"
}

$BS(0)  set     X_      250
$BS(0)  set     Y_      250

$BS(1)  set     X_      500
$BS(1)  set     Y_      250

$BS(2)  set     X_      250
$BS(2)  set     Y_      500
```

```
$ns_ node-config -wiredRouting OFF \
                 -mobileIP OFF

for {set i 0} {$i < $val(nn)} {incr i} {
        set node_($i) [$ns_ node 1.1.$i]
        $node_($i) base-station [AddrParams addr2id [$BS(0) node-addr]]
        $node_($i) label [$node_($i) node-addr]
        $node_($i) random-motion 0
        set     ragent_($i)     [$node_($i) set ragent_]
        $ragent_($i)    set     mobilenode      1
}

#source bs_access

$ns_ duplex-link $W(0) $BS(0) 10Mb 1ms DropTail
$ns_ duplex-link $BS(0) $BS(1) 10Mb 1ms DropTail
$ns_ duplex-link $BS(0) $BS(2) 10Mb 1ms DropTail

set d 270
for {set i 1 }  {$i < $wired_nodes } { incr i } {
        $ns_ duplex-link $W($i) $W(0) 10Mb 10ms DropTail
        $ns_ duplex-link-op $W($i) $W(0) orient [expr $d]deg
        set d [expr $d+40]
}

source scenario_$val(nn)

$W(0)   set     X_      100
$W(0)   set     Y_      100

$W(0)   label   video_server

proc running { } {
        global ns_ W wired_nodes BS
        for {set i 0 }  {$i < $wired_nodes } { incr i } {
            $ns_ puts-nam-traceall "n -t 0.000001 -s $i -x [$W($i) set X_
        }

        for { set j 0 } { $j < 3 } { incr j } {
```

```
        $ns_ puts-nam-traceall "n -t 0.000001 -s [expr $i + $j] -x [$BS(
        }
}

$ns_ at 0.000001 "running"

proc create_udp { id src dst st stp } {
global ns_ node_ W udp_ stream_
set udp_($id) [new Agent/UDP]
$ns_ attach-agent $W(0) $udp_($id)
$udp_($id) set packetSize_ 1000

set null_($id) [new Agent/Null]
$ns_ attach-agent $node_($dst) $null_($id)
$null_($id) set packetSize_ 1000

set stream_($id)         [new Application/Video]
$stream_($id) attach-agent $udp_($id)
$stream_($id) load_file "/root/Desktop/output_video"
$stream_($id) set pkt_size       512
$stream_($id) set interval_      0.1
$ns_ connect $udp_($id) $null_($id)

$ns_ at $st  "$stream_($id) start"
$ns_ at $stp "$stream_($id) stop"
}

for {set i 0 }  {$i < $val(nn) } { incr i } {
$ns_ initial_node_pos $node_($i) 30
set null_($i) [new Agent/Null]
$ns_ attach-agent $node_($i) $null_($i)
}

create_udp 0 1 4 15 195
#create_udp 1 1 2 15 195
#create_udp 2 1 1 15 195


$ns_ at $val(stop).0000001 "$node_(0) reset";
$ns_ at $val(stop).000001 "$ns_ halt; exit 0"
$ns_ run
```

.7 tcl script for Gauss-Markov model

This script is used for Wired and Wireless Scenario for Gauss-Markov model:

```
set val(x)                1000    ;      # X topography
set val(y)                1000    ;      # Y topography
set val(stop)             200     ;      # Simulation Time
set val(routing)          MIPV6   ;      # Routing protocol
set val(energymodel)      EnergyModel;
set val(initialenergy)    100     ;# Initial energy in Joules
set val(nn)               5
set wired_nodes           1
set val(bs)               3

set ns_ [new Simulator]

if { $argc == 1 } {
set     val(nn)           [lindex $argv 0]
}

set     playframe         5000

Application/Video         set     modelling       0
Agent/MIPV6               set     playframe       $playframe
Mac/802_11                set     dataRate_       11Mb

Application/Video         set     playframe       $playframe
Application/Video         set     pkt_size        512
Application/Video         set     frame_id        0
Application/Video         set     interval_       0.1

Agent/MIPV6     set     multicast       0

$ns_ color 0 brown
$ns_ node-config -addressType hierarchical

AddrParams set domain_num_ [expr $val(bs) + 1]
lappend cluster_num 1 1 1 1
AddrParams set cluster_num_ $cluster_num
lappend eilastlevel 6 6 6 6
AddrParams set nodes_num_ $eilastlevel
```

```
set topo [new Topography]
$topo load_flatgrid $val(x) $val(y)

set t [open out.tr w]
$ns_ trace-all $t
$ns_ use-newtrace

set nt [open out.nam w]
$ns_ namtrace-all-wireless $nt $val(x) $val(y)

set god_ [create-god [expr $val(nn) + $val(bs)]]

for {set i 0} {$i < $wired_nodes} { incr i} {
        set W($i) [$ns_ node 0.0.$i]
}

$ns_ node-config -adhocRouting $val(routing)\
                 -llType LL \
                 -macType Mac/802_11 \
                 -ifqType Queue/DropTail/PriQueue \
                 -ifqLen 50 \
                 -antType Antenna/DirAntenna \
                 -propType Propagation/TwoRayGround \
                 -phyType Phy/WirelessPhy \
                 -channelType Channel/WirelessChannel \
                 -topoInstance $topo \
                 -energyModel $val(energymodel) \
                 -rxPower 0.6 \
                 -txPower 0.9 \
                 -initialEnergy $val(initialenergy) \
                 -agentTrace ON \
                 -routerTrace ON \
                 -wiredRouting  ON \
                 -macTrace OFF

for { set i 0 } { $i < $val(bs) } { incr i } {
        set BS($i) [$ns_ node [expr $i + 1].0.0]
        $BS($i) label "AP"
        set     mac($i) [$BS($i) getMac 0]
        set     ragent_bs($i)   [$BS($i) set ragent_]
#       $ns_ at 0.0 "$mac($i)   bs_exec"
```

```
}

$ragent_bs(0)    set      W_AP    1

for { set i 1 } { $i < $val(bs) } { incr i } {
        $ns_ at 0.0 "$mac($i) bs_exec"
}

$BS(0)  set     X_      250
$BS(0)  set     Y_      250

$BS(1)  set     X_      500
$BS(1)  set     Y_      250

$BS(2)  set     X_      250
$BS(2)  set     Y_      500


$ns_ node-config -wiredRouting OFF \
                 -mobileIP OFF

for {set i 0} {$i < $val(nn)} {incr i} {
        set node_($i) [$ns_ node 1.1.$i]
        $node_($i) base-station [AddrParams addr2id [$BS(0) node-addr]]
        $node_($i) label [$node_($i) node-addr]
        $node_($i) random-motion 0
        set     ragent_($i)     [$node_($i) set ragent_]
        $ragent_($i)    set     mobilenode      1
}

#source bs_access

$ns_ duplex-link $W(0) $BS(0) 10Mb 1ms DropTail
$ns_ duplex-link $BS(0) $BS(1) 10Mb 1ms DropTail
$ns_ duplex-link $BS(0) $BS(2) 10Mb 1ms DropTail

set d 270
for {set i 1 }  {$i < $wired_nodes } { incr i } {
        $ns_ duplex-link $W($i) $W(0) 10Mb 10ms DropTail
        $ns_ duplex-link-op $W($i) $W(0) orient [expr $d]deg
        set d [expr $d+40]
```

```
}

source scen−$val(nn)

$W(0)    set      X_       100
$W(0)    set      Y_       100

$W(0)    label    video_server

proc running { } {
        global ns_ W wired_nodes BS
        for {set i 0 }  {$i < $wired_nodes } { incr i } {
            $ns_ puts−nam−traceall "n −t 0.000001 −s $i −x [$W($i) set X
        }

        for { set j 0 } { $j < 3 } { incr j } {
        $ns_ puts−nam−traceall "n −t 0.000001 −s [expr $i + $j] −x [$BS(
        }
}

$ns_ at 0.000001 "running"

proc create_udp { id src dst st stp } {
global ns_ node_ W udp_ stream_
set udp_($id) [new Agent/UDP]
$ns_ attach−agent $W(0) $udp_($id)
$udp_($id) set packetSize_ 1000

set null_($id) [new Agent/Null]
$ns_ attach−agent $node_($dst) $null_($id)
$null_($id) set packetSize_ 1000

set stream_($id)        [new Application/Video]
$stream_($id) attach−agent $udp_($id)
$stream_($id) load_file "/root/Desktop/output_video"
$stream_($id) set pkt_size       512
$stream_($id) set interval_      0.1
$ns_ connect $udp_($id) $null_($id)

$ns_ at $st  "$stream_($id) start"
$ns_ at $stp "$stream_($id) stop"
```

```
}

for {set i 0 }  {$i < $val(nn) } { incr i } {
$ns_ initial_node_pos $node_($i) 30
set null_($i) [new Agent/Null]
$ns_ attach-agent $node_($i) $null_($i)
}

create_udp 0 1 4 15 195
#create_udp 1 1 2 15 195
#create_udp 2 1 1 15 195


$ns_ at $val(stop).0000001 "$node_(0) reset";
$ns_ at $val(stop).000001 "$ns_ halt; exit 0"
$ns_ run
```