PERFORMANCE EVALUATION OF MIXED SCTP AND TCP TRAFFIC OVER LAST HOP WIFI

by

Qamar Naith

Bachelor's Degree in Computer Science, Umm Al-Qura University, 2009

A thesis

presented to Ryerson University

in partial fulfillment of the

requirements for the degree of

Master of Science

in the Program of

Computer Science

Toronto, Ontario, Canada, 2014

AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A THESIS

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my dissertation may be made electronically available to the public.

PERFORMANCE EVALUATION OF MIXED SCTP AND TCP TRAFFIC OVER LAST HOP WIFI

Master of Science 2014

Qamar Naith

Computer Science

Ryerson University

Abstract

The use of the internet has increased significantly with the continued increase in wireless communication devices. Recently, there is a large number of research contribution focused on Stream Control Transmission Protocol (SCTP). Multi-homing is an important feature of SCTP which improves the communication performance by usage of multiple paths during association establishment, and it can bring significant improvements of throughput.

In this thesis we evaluate the performance of SCTP and TCP traffic in the WLANs and we investigate the SCTP multi-homing to improve the communication performance in WLANs. We conducted some experiments to evaluate the performance of SCTP multi-homed host under various channel bit rates and mobility speeds. The results indicate that when the intensity of background traffic increases the SCTP multi-homed host with higher channel bit rate has better performance. In addition, the SCTP multi-homed host with using lower mobility speed has higher performance (throughput, delay and packet loss).

Acknowledgements

During our life time some of us are profoundly blessed with a transcendent professor who enkindles a student to search within and chance upon their brilliance. Dr. Jelena Misic is the epitome of an extraordinary professor that some students happen upon once in their lifetime if lucky. I would like to take a moment to thank this wonderful person with whom I happened to be blessed with as a supervisor. Dr. Misic enabled me to learn how to become more self-reliant and discover my own self-efficacy.

Dr. Jelena Misic benevolently agreed to take me under her wing; supervise my work and help me work towards accomplishing my thesis. Her knowledge has helped me combine what I learned in my undergraduate degree and current Masters work to provide me with a brilliant foundation to pursue my dreams of obtaining my PhD. Working with the brilliant and ever patient Dr. Misic I have honed in my intellectual development and my educational experience has been beyond my wildest expectations.

Dr. Misic, thank you for being there for me, and being a beautiful human being. You will forever be part of my successes in life and my contributions to the world at large.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Transport layer provides the communication service between devices connected with each other via the Internet. With the evolution of modern Telecommunication Wireless Networks we have been able to develop the permanent growth communication services over IP Networks. This development depends on the presence of some protocols in the transport layer that help the network to transfer huge amounts of data between the two end hosts. In general the protocols in the transport layers play a significant role by offering end to end data transport services to applications in the host. Transport layer services include: connection oriented data transport, ordered delivery, reliability, as well as congestion control [1].

Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) are the most common protocols that are used at the OSI transport layer. Both TCP and UDP are insufficient for some application requirements. As compared to these protocols, Stream Control Transport Protocol (SCTP) is a newly developed protocol for the transport layer. It was first and foremost created for the transportation of signaling messages over cellular networks, but later it appeared like a more generalized protocol of the transport layer. SCTP was developed for the transportation of telecommunication signaling over the IP layer. SCTP works like TCP with the extra characteristics essential to transport signaling data. It provides a reliable transmission, acknowledging when information is reordered, and retransmitting damaged information as indispensable [2]. SCTP provides better communication between two end devices by supporting multi-homing. Multi-homing is considered to be one of the key features of SCTP that provide a high performance level for mobile host in wireless environments . There are some reasons for

using a aforementioned feature immediately during the data transfer; if the primary path (IP address) used has failed during the association and data transmission phase, or if the SCTP multi-homed moves across the wireless network. In this case, SCTP multi-homing will implement the handover scheme by using an alternative path (IP address) from the addresses that are assigned for both the sender and receiver hosts in the beginning of the association. This feature gives the SCTP Protocol high network level of fault tolerance [2, 3].

1.1 Transport Layer Overview

The transport layer (4th level in OSI model) was designed to guarantee data delivery in the order that they were sent. It does not matter what kind of data is been transmitted, how and where because it provides the mechanism of transmission. There are many transport protocols ranging from the protocols that provide only basic transport functions (e.g., data transmission functions without acknowledgment) to the protocols that guarantee delivery to the destination of data packets in a correct sequence and ensure the accuracy of the received data [4].

1.2 User Datagram Protocol (UDP)

UDP is considered connectionless protocol that can be used in transport layer. It was defined in RFC 768.This protocol is sometimes referred to unreliable protocol delivery, which means the UDP provides datagram delivery and does not guarantee its implementation. Datagram is a data unit that is passed through the network independently of other data units without establishing a logical connection and acknowledgment in UDP. Datagrams, by themselves, do not contain the detection and correction of transmission errors. Methods of organization of reliability can be very different, unusually the same method is used to get an acknowledgment by sending an echo-response each time you receive a datagram packet. UDP is used for clients that send only short messages and can simply re-send the message if there is no response of confirmation it will not come fast enough. UDP protocol preserves message boundaries defined by the application process. For example, if the sending process produces three entries in the UDP- port, the process of the recipient will be required to make three readings. The size of each recorded message will be the same size as the corresponding read [3].

1.2.1 Restrictions / Limitations of UDP

- UDP unreliable protocol, which mean does not guarantee delivery of the datagram, a datagram may not be delivered, delivered twice, or delivered out of order[3].
- UDP lack the communication among two hosts and this will cause errors during transmission of datagram.
- UDP is limited to the control of the integrity of data in a single datagram, and does not exclude the possibility of losing the entire datagram, or datagram duplication, where as the TCP provides reliable continuous transmission of data [3].

1.3 Transmission Control Protocol (TCP)

TCP is considered one of the most popular transport protocols between two end host connections over the IP network today. TCP has been defined in RFC793 that was designed by the Internet Engineering Task Force (IETF). The main objective of this protocol is to make reliable communication between two hosts. TCP is byte-oriented data transmission, stream of byte is transfered in segments between end hosts. In each segment the number of bytes is decided based on the maximum segment size (MSS) of the connection. In TCP unique sequence number is assigned to each byte transmitted to reorder byte at the receiver [4]. The window size in TCP defines the number of bytes that may be sent before an acknowledgment from the receiver. TCP ensures that all PDUs (Protocol Data Unit) will be delivered successfully due to the strict ordered delivery for all data sent between the two hosts.

This delivery service with strict order has become one of the major limitations to some network applications such as VOIP applications.This is will cause a Head Of Line blocking (HOL), which means when the packet at the head of the queue line is waiting, the other packets can not be forwarded even if they are going to other destination[5]. The HOL blocking can be considered as a serious problem in TCP especially if the receiver window size of the hosts is very small. Further, in TCP if any packet is lost the other packets received by the receiver will be stored at the receiver buffer and will not be processed until retransmission of the lost packets. TCP treats every data like an unstructured series of bytes. Because

of these applications that handle individual messages should add to the byte stream border posts and track [4, 6]. In TCP connection when one host connects to another host it can only associate with one IP-address to each host. If the interface is assigned to that IP-address is disabled, TCP connection is broken and needs to be re-established .

Further, if the host attempts to start a TCP connection with another host, there are three steps that must follow to make the TCP connections successful. Basically, when the sender needs to connect to the receiver, the sender sends a SYN message to the receiver. In this case the receiver is allocated resources directly when it receives SYN then replays by SYN-ACK message to the sender to acknowledge sender message. Thus, sender responds to the receiver's message by sending ACK to the sender and setup connection. After all these three steps are achieved (" TCP three-way handshake"), the connection between sender and receiver is establishesd [3]. According to RFC 793, a TCP connection supports Half open/ Half close connection HOC/HCC). For instance, if the sender does not send an ACK to the receiver to complete the "three way handshake" connection and remove the socket without notifying the receiver then the receiver will wait at least one minutes in a half open state and then terminate the connection.

TCP is vulnerable to SYN flooding attacks such as DoS (Denial of service attacks). The SYN flooding attack means that attacker sends large number of TCP SYN requests to the receiver by using forged IP addresses and allocating resources to these forged requests without completing the third handshake step. This is will lead to amplified and increase resource allocated. In recent years, this is overcome by creating SYN cookies. The SYN cookies work as follows [6]:

- If the receiver receives a TCP SYN request, it does not know the source of this request; for instance, if this SYN request is coming from a legitimate hosts or is part of a SYN flood attack. Thus, the receiver will not create half open TCP connection for this SYN request.
- The receiver uses TCP cookie segment instead of sending SYN ACK segment immediately. This cookie includes an initial sequence number, source and destination IP address, and port numbers of SYN request as well as use secret number that only known by receiver.
- The receiver then sends a SYN ACK segment with the initial sequence number that has been

defined within cookie.

- if the sender it legitimate, then it will send an ACK segment to the receiver.
- Since, the receiver receives an ACK segment, it needs to verify that the ACK segment corresponds to same SYN request that has been send earlier.
- If the value in the Acknowledgment filed is equal to the sequence number and secret number in the SYN ACK segment. The receiver in this case will create fully open connection. Otherwise, the receiver will send special TCP reset segment with RST flag bit, which means sender should immediately stop using the TCP connection and stop sending any more packet because the receiver does not have a socket for the SYN request that has been sent earlier. Thus, the original SYN will be ignored and the server has not allocated any resources to this false SYN request.

1.4 TCP Mobility and Handover Management

Mobile IP (MIP) is an extension to standard Internet Protocol (IP) that was designed by IETF [7] to enable mobile node (MN) to roams between IP networks (move from coverage network area to another one based on the signal strength) while preserving a permanent IP address and connection [7]. Mobile IP is often found in wireless environments when the MN across multiple networks boundaries with various IP addresses. Further, TCP is often used MIP(over IPv4) approach in order to achieve the mobility across entire internet and resolved the mobility problem by creating two IP addresses to gain seamless and continuous internet connectivity. These IP addresses are Home Address (AH) and Care-of-Address (CoA). This approach will cause more delay to achieve the mobility function. Recently, TCP can also perform by IPv6 to improve the end-to-end communication performance between TCP hosts and to provide high efficiency of mobility in wireless environments. Table 1.2. shows that the use of MIPv6 for mobility and handover functions is better than the use of MIPv4.

1.4.1 Restrictions / Limitations of TCP

The TCP protocol provides the basic function for data transfer over the Internet for a reliable way. However, TCP imposes some restrictions on the transport of data:

- TCP requires a strict byte order delivery of data transmitted between multiple hosts. It means receiving the data transmitted in the same order as it was sent. This order can increase head of line (HOL) blocking in some cases[4, 5].
- TCP encourages the HOC, which means that sender waits for the acknowledgment from receiver in some cases; for example, if other of the two hosts malfunctions or something wrong occurs to the IP address (path) associated with the two hosts. In this cases the TCP needs to re-establish the connection between the hosts [3].
- TCP is prone to SYN flooding attacks, which lead to increase resource allocated.
- TCP does not encourage multi homing and multi streaming service which are crucial in high availability environments such as SS7 signaling transport [5].

1.5 Stream Control Transmission Protocol (SCTP)

SCTP is network transport layer protocol in the network TCP/IP, as described in RFC 2960 to extend and improve Transport Layer functionality. SCTP was designed in 2000 by the IETF Signaling Transport (SIGTRAN) working group to fulfill all the Signaling System (SS7) environment requirements that were recommended to be accomplished. In general, SCTP was initially developed to convey Public switched telephone network (PSTN) signaling messages over IP networks. Over the past decade, SCTP advanced into a universally useful transport protocol that incorporates advanced transportation choices [5]. This protocol is message-oriented that gives a full-duplex and reliable connection, known as an association. Its main characteristics are multi-streaming and multi-homing [5, 10, 11]. Stream Control Transmission Protocol (SCTP) provides transport signaling messages over an IP network between the two end points, with the redundancy of information delivery and increased reliability. SCTP architecture

stack is very compatible with the architecture of the Internet[12].

SCTP has been created as part of a project launched by the working group IETF Signaling Transport and dedicated to the development of a specialized transport protocol for decisions related to voice over IP-networks (VoIP-telephony). As with TCP, SCTP protocol provides applications with point to point service with guarantee delivery.

1.5.1 SCTP Packets Format

SCTP packet structure has two basic components: Common header and SCTP chunks. Common header can be considered the first main part of SCTP packet structures that consists of two main port addresses and IP addresses (source and destination IP address) used to identify the association, verification tag (a 32 bits random values is using in the beginning of association to validate the packets transmitted during association by using initiation tag that assign when the association start up. If any packet transmitted does not have this initiation tag, the packet will be dropped), checksum (Value that calculated by using CRC32 algorithms to guarantee data integrity that cross the IP network), and numbers of SCTP chunks, as illustrated in Figure1.1 [4].

SCTP chunks are the second main part of the SCTP packet structure that included user data. There are two types of SCTP chunks; control and data chunks. Both of them may exist within a single SCTP Packet. Control chunks contains information required to maintain and control the association, and it must be always ahead of data chunks. While data chunks is contained message (application data). According to [11], the length of individual data chunk is up to 655634 bytes or more; different numbers of data chunks are necessary in SCTP. Each data chunk has a different number of mandatory fields : Chunk Type (An 8 bits value used to recognize various types of chunks as shown in Table 1.1) , Chunk Flag (An 8 bits the default value is zero, it is related to the chunk types), Chunk Length (Can be considered a variable length approximately 16 bits, and it is required for each chunks), TSN (Transport sequence number for the association reliability from 32 bits), SSN (Stream sequence number for stream ordering 16 bit), SI (Each stream include multiple message, each message has a special stream identifier) PID (Protocol Identifier- the default value is always zero), and User Data (User Payload data), while each

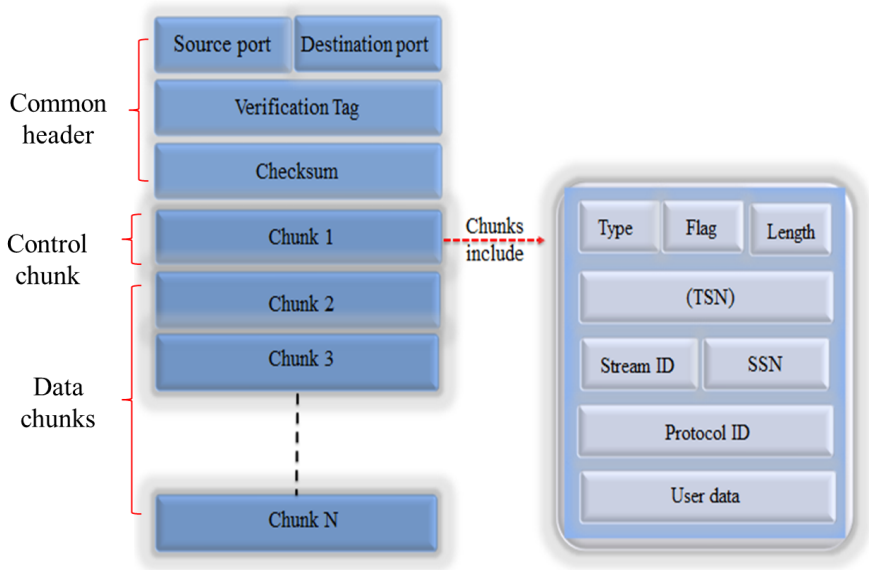

Figure 1.1: SCTP packet structure [4].

field has a set of parameters[13]. Table 1.1 describes all the chunk types which are used in the SCTP data chunk structure with particular numbers related to each type.

1.5.2 SCTP Features

SCTP has considered the most significant telecommunication protocol over IP networks. It was recently designed to prepare a reliable transport between two SCTP hosts over IP networks[2]. SCTP resembles TCP and it is able to eliminate most the TCP deficiencies by supporting a number of features inherited from TCP, which include reliability, order delivery of data, flow control, and full duplex data transfer. In contrast, it has also other features from UDP which are message oriented and preserves the boundaries of the message and other own significant function as well[4]:

- SCTP is connection-oriented, used Association expression rather than Connection;
- Support message fragmentation;
- Support Multi-streaming ;
- Stream message oriented instead of byte oriented;
- SCTP encourages using many data delivery modes;
- Eliminates head of line (HOL) blocking ;
- Support Congestion control and Avoidance;
- Support Multi-homing.
- Use Security Cookies Mechanism (SCM) instead of SYN flooding attacks;

Association Establishment

Initiation Process

SCTP uses the term a association instead a connection to establish a session. Each association has multiple streams while each host in SCTP has multiple IP addresses and port numbers. It is not allowed to establish more than one association at the same time in SCTP. There are two basic steps of association establishment: Initiation process that uses the "four way handshake, and shutdown process uses the three way handshake. SCTP uses "four- way handshake"(as shown in Figure 1.2), meanning that the sender sends an INIT chunk (with a cookie wait state). There are some important parameters included in the INIT chunk for setting up the initial state: IP addresses, TSN, initiation Tag, and

numbers of outbound and inbound streams. After the receiver receives INIT chunk and analyses all the data contained in, it will return an INIT-ACK (Initiation Acknowledgment) chunk to the sender by adding into the cookie state a secret key with derived MAC (Message Authentication Code). Further, INIT-ACK includes the same parameters that used in INIT chunk [4].

In general, INIT-ACK plays an important role in association establishment because the INIT-ACK chunk contains a cookie that saves all the significant information which is helpful for both hosts during communication. When the sender receives the INIT-ACK chunk with the cookie contained, the sender unpacks all the cookie information that has been sent directly in a new cookie chunk that is called COOKIE-ECHO chunk and resends it again with a COOKIE-ECHOED state to the receiver. Finally, the receiver unpacks all the cookie information that it receives from the sender and uses the MAC technique to check if it received the same information that has been sent in the beginning and if it has the same secret key. If the cookie verifies, and the result of MAC gives an OK, the receiver will send back a COOKIE-ACK chunk to the sender to acknowledge the complete setup. In this case the receive allocates resources and establishes the association. All of these cookies have been used in this procedure (cookies mechanism) are more secure than the other mechanisms that has been used in TCP which is SYN flooding attacks [4, 13].

Shutdown Process

Each connection-based protocol (TCP, and SCTP) needs an efficient technique to initiate and shut-down the association while the connectionless like UDP does not need this technique. SCTP has a powerful technique to terminate the association without losing any packet. Actually, the two hosts in SCTP do not support the HOC technique compared with TCP. In SCTP the two hosts should not accept any data from the upper layer and send it from one host to another when the termination is initiated [4, 13].

In SCTP, after exchanging the data between sender and receiver, if the sender decide to terminate the association, it will absolutely change its state from an established state to the shutdown pending state.This occurs by achieving three-way handshake to terminate the association (as illustrated in Figure 1.2). This means the sender will prohibit accepting any data from the upper layer until the receiver

responds with the SACK chunk for all the data chunks sent. The sender will send a SHUTDOWN chunk to the receiver. In this case, each of these two hosts must change their states to the shutdown state. The receiver will accept the SHUTDOWN chunk and send back all the DATA chunk that it had received back to the sender. Thus, sender sends the SACK and SHUTDOWN chunk to the receiver. The receiver will send a SHUTDOWN-ACK chunk to the sender. The sender will respond with a SUTDOWN-CMPL, which means the association is completely shut down[4].

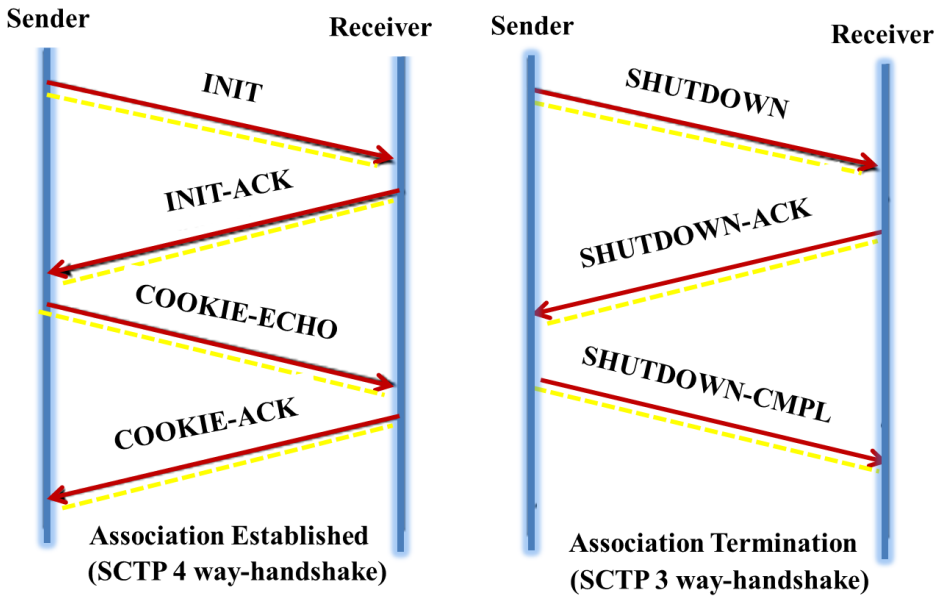

Figure 1.2: SCTP Association and Termination procedure[4].

Message Fragmentation

Message fragmentation is one of the most important features in the transport layer especially in the SCTP transport protocol. In fact, the transport layer accepts a large amount of data from the upper layer then sends it to the lower layer, via some routed path. SCTP has a new feature that can achieve a significant function, which is to fragment the message (user data) received from the upper layer into a number of smaller messages based on maximum transmit unit (MTU) size and then transmit it via

many routed paths within multiple independent chunks. Fore example, if the message that was accepted from the upper layer is larger than MTU size, the message will be segmented and transmitted within independent chunks. The segments of a message are multiplexed into one or more SCTP chunks. This methodology has been defined in RFC1191 to ensure the size of data transmitted, to avoid of data delay transmit, and to prevent data loss that could occur over the IP network[4].

Multi-streaming

Multi-streaming is one of the most important SCTP features.This feature allows a single association to have sequence of streams that are independent of each other. Each stream is assigned with unique stream number. Multi-streaming solves the problem of Head of Line (HOL) blocking by divide message into number of message streams to improve the flexibility to transfer data of different applications on different streams. Moreover, it is known that the SCTP sender receives message from the upper layer while the receiver sends the message to the upper layer. Therefore, if some segments (user message) of any streams are lost during the transmission phase, this does not affect the transmission of message to the application. The receiver will store these segments lost in stream buffers until retransmitted, while the other streams continue to send its segments to the upper layer [5].

Data Delivery Mode

SCTP is a reliable transport protocol that send and receive the message between two SCTP hosts correctly. Further, TCP is also considered as a reliable transport protocol; however, it has some delivery constraints which means that the destination host must deliver the data transmitted in which it has been sent by the source host then presented to the application layer. SCTP has several modes to deliver the message between the source and destination hosts without requesting any specific order (like TCP) or non-specific order (like UDP). That is because it has a new delivery order mode called partially order that can eliminate head of line (HOL) blocking , which means that the destination host will deliver the sequence streams of the message as it has been received[4].

This situation shows exactly the important functions of using TSN and SSN. Moreover, TCP uses TSN for each segment (data packet) transferred. This will often cause some data to be dropped and lost. On the other hand, SCTP use the same TSN for each stream (chunk) in addition to using SSN for each

segment within stream(user message) to retrieve the lost segments; any other streams can hold the same SSN for its separate segments[5]. The partial ordering of the SCTP provides the opportunity to carry out an orderly delivery of one or more related sequences of stream sent between the two hosts. With this SCTP can be particularly useful in applications that require reliable delivery and fast processing of multiple unconnected data streams.

Congestion control and Avoidance Mechanism

Congestion control is considered as one of the most important features that ensure the reliability of data transmitted in TCP and SCTP. In[2] TCP and SCTP use the same mechanism that has been defined in RFC2581. SCTP has used this technique for the each association, not for the individual stream as TCP. Further, this technique is used to provide highly reliable data transmission, to detects if the packet is lost or corrupt, and to ensure the reduction of the sending data transmission rate during network connection in SCTP. In SCTP, to perform the congestion controls each host should hold three variables to preserve the data transmission rate into the network [11]:

- Receiver advertised window size (rwnd)
- Congestion window size (cwnd)
- Slow start threshold size (ssthresh)

There are three basic component that could be integrated together to represent the main role of SCTP congestion control: slow start , congestion avoidance, and fast retransmit ; while TCP has a additional component named fast recovery.

Slow Start (SST)

It is recognized that when any two SCTP hosts connect to each other via the IP network, probably these hosts will sending huge amount of data from sender to receiver. The frequent transmission of these data respectively may lead to network congestion, failure data transmission, and poor network. Obviously, the slow start is the powerful algorithm to avoid the congestion and any issues that may lead to the poor communication network by achieving many functions: determine the space that has be available to transmit the maximum amount of data before starting the transfer operation, or determine

the space that has be available to retransmit any packet lost during the data transfer at the associations. The cwnd size in the slow start in SCTP is usually less than ssthresh, and it will be increased by the number of bytes acknowledged while in TCP increased by the number of new ACK received [2]. Moreover, the SST can occur when

$$cwnd \leq ssthresh \tag{1.1}$$

while it was in TCP either SST or CA when

$$cwnd = ssthresh \tag{1.2}$$

And in SCTP the initial cwnd size is less than or equal to twice the value of MTU

$$cwnd \leq 2^{*}MTU \tag{1.3}$$

and it was one in TCP according to the value of MTU that the recently adopted in [2, 11]. obviously, during the SST the cwnd size increases exponentially every RTT (Round Trip Time)[2]. RTT is the length of time it takes to send the packet plus the length of time it takes to receive the acknowledgment for this packet received and this also includes the propagation times between the two nodes on the network.

Congestion Avoidance(CA)

SCTP can use the congestion control and avoidance algorithm and increase the value of cwnd by approximately one MTU per RTT, when cwnd after SSA is greater than ssthresh.

$$cwnd \geq ssthresh \tag{1.4}$$

In [2, 5] ” cwnd can only be increased when the full cwnd is utilized” which means the two of hosts have been connects to each other and have used all the network resources available.

Fast Retransmit (FRT)

This algorithm has been created in order to control the retransmit single packet drop or that has been falling during the transmission process. FRT is helpful to guarantee the reliable retransmission, and lack of any negative effect leads to lower network efficiency or the data transfer process. According to [2], in SCTP any data chunks can be considered totally delivered depending on two conditions: when the cumulative TSN ACK index passes the TSN of data chunk, or when the data chunk have been acknowledged by SACK . Otherwise, the SCTP uses some methods to retransmit the packets that drop by using Gap ACK Blocks. This method is very useful when it transmit any packet over IP networks. For example, when sender sends 8 data chunks consecutively to receiver, there is the different TSN assigns for each data chunk sent. Receiver will respond by sending SACK (for all the data chunks or any data chunks received) to the sender.

In [2] during the transmission there is a retransmission timeout/timer (RTO) sets for each new data chunk that will be send. If this timer finishes before receiving SACK from the receiver for any data chunks that were send to, the cwnd is dropped to one and retransmits the data chunks again in SSA mode. If some of TSN is not acknowledged while other newer TSN is still acknowledged. In other word the if some Gap ACK Block are found in any SACK (i.e. if some data chunks that lost or dropped) for any reasons, the fast retransmission algorithm is involved to solve this problem by retransmits directly the TSN that was not acknowledged yet [11]. There are two main reasons to find a Gap ACK Block in any SACK :

1. If some TSN is missing in any SACK chunk during association when the data is transmitted, sender will wait to receive four consecutive messages to retransmit the data chunks which holds TSN that missed. The minimum rwnd size that is required to be in FRTM is 5 * MTU as it mentioned in [2]. In contrast, TCP in fast transmit occurs when sender is waiting to receive three consecutive messages to start performing the fast FRT instead of four. In this case the minimum rwnd size required is 4 * MTU as mentioned also in [2]. This can be shown this mathematically as indicated in [11].

$$cwnd = ssthresh \tag{1.5}$$

And this means that, sender in the slow-start phase will reduce the size of cwnd to be equal to

ssthresh.

2. if the retransmission timer (RTO) has expired for any data chunks not received by SACK yet . This situation, it can be shown mathematically as indicated in [11].

$$cwnd = 1^{*}MTU \tag{1.6}$$

Fast Recovery (FR)

The fast recovery is the fourth algorithm that is used in the TCP congestion control as it defined in [11]. SCTP basically does not need this algorithm due to it has a good technique that avoids flooding the network or the duplicate ACK, called Gap ACK Block[2]. This mechanism indicates the number of gap ACK block start and gap ACK block end between two nodes.

1.5.3 Multi-homing technology

The most significant features of SCTP, which distinguishes from other transport protocol and plays important role over IP network are multi-homing and multi-streaming capability. These two features help SCTP to overcome all the problems that may lead to poor throughput and delay. Multi-homing is one of the main SCTP features which allows two SCTP hosts to establish an association and transfer data over more than one path (IP addresses) interfaces. When the association is established and starts transmitting the data among two SCTP hosts, it is necessary for each one of the hosts (in one side) to have a list of all the IP addresses of another host (other side). In the beginning of association sender has to detect and advertise the address that will be used as a primary destination IP address that can send all the data chunks through this address by default; the remaining IP addresses will be used as needed. For example, sender sends data chunks to the receiver. The receiver must reply by acknowledgment chunk to the sender, using the same path that was used previously via client. In addition, there is another situation if sender sends multiple data chunks through various paths (using more than one IP address) simultaneously, receiver will reply acknowledgment through any of these paths used as incorporated in RFC 2960 [15].

There is a reason for using an alternative path (IP address) immediately during the data trans-

fer in SCTP, if the connection with primary path has failed during association, SCTP will switch to alternate path (backup destination address) that are assigned for both sender and receiver hosts in the beginning of the communication. If the primary path is operational again SCTP switches back to the primary path and use it again for data transmitting [13]. According to [1], there are many types of multi-homing in SCTP; Asymmetric multi-homing and Symmetric multi-homing. Asymmetric multi-homing means that one of the two hosts has equipped with multiple interfaces (multiple IP addresses) and the other with only one interfaces (one IP address) as shown in Figure 1.3. In addition, Symmetric multi-homing means both hosts have equipped with multiple interfaces (multiple IP addresses) as shown in Figure 1.4.

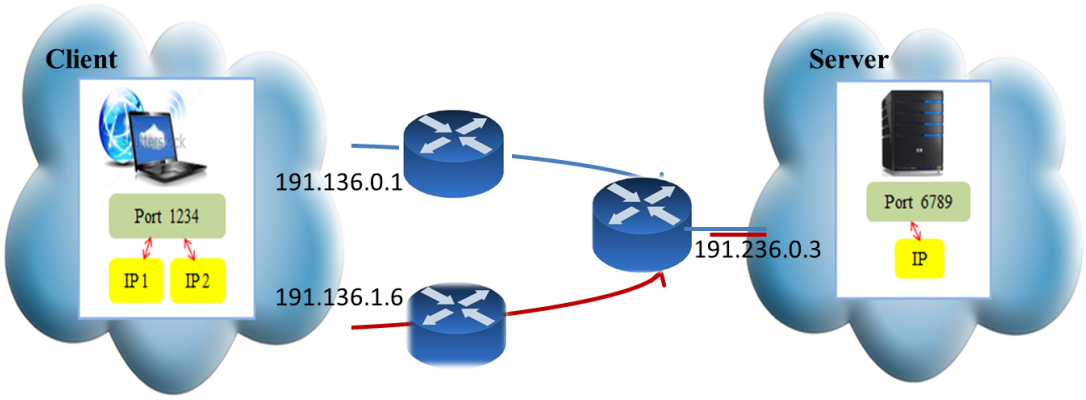

Figure 1.3: Asymmetric Multi-homing.

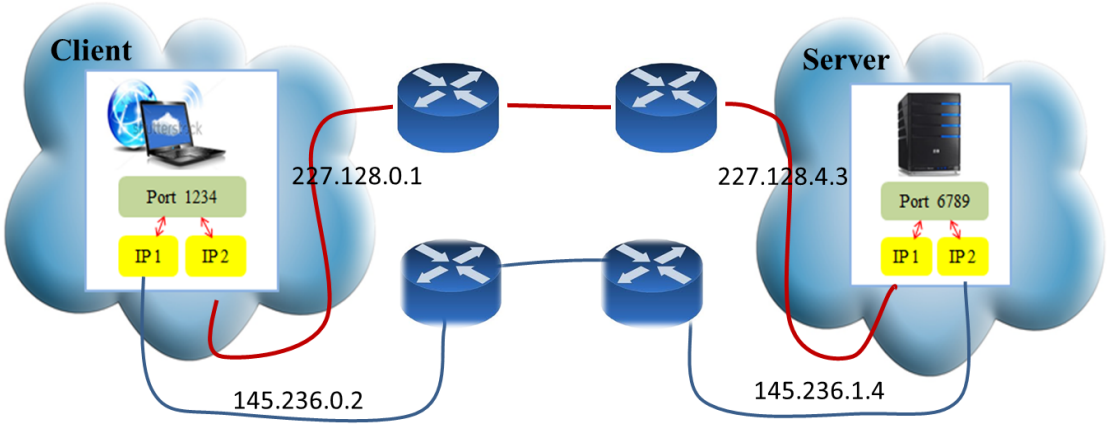

Figure 1.4: Symmetric Multi-homing.

In SCTP Multi-homing, when two hosts are connected to each other through the paths that has used, the SCTP uses essential chunks named HEARTBEAT chunks. HEARTBEAT chunks sent over all paths to observe all the data that was transferred over the paths and to determine the transmission time of send the HEARTBEAT chunk and reception of HEARTBEAT-ACK chunk [1]. Further, SCTP supports two essential extensions, which are related to multi-homing feature: Dynamic Address Reconfiguration(DAR) and Partial Reliable SCTP (PR-SCTP).

The DAR, which has been defined in RFC 5061 [16] to provide high degree of capacity; it gives flexibility dynamically to add, change, or drop any IP addresses that is not needed for any reason during the association. In order to reconfigure the IP addresses dynamically, we need to include special chunk within the SCTP packet such as Address Configuration Chunk (ASCONF), which used by the sender to inform the receiver that will add, change, or delete the IP address from the association and Address Configuration Acknowledgment (ASCONF-ACK), which is used by the receiver to inform the sender that was received the ASCONF chunk [17]. In addition, the PR-SCTP is the second SCTP Multi-homing extension , which has been defined in RFC 3758 [18] to provide a partially reliable delivery option and provides flexibility to set the validity of the packets. For example, if the validity of any packets is expired

at the receiver, the packets are dropped [12].

1.5.4 SCTP Mobility and Handover Management

SCTP Multi-homing with ADDIP extension (mSCTP) has become one of the mobility management schema at the transport layer that is used for the handoff management solution. The mSCTP is supporting the IP diversity, which achieves seamless handover for mobile nodes that are roaming between different networks, and aims to improve the wireless networks performance, such as low handover latency (delay) , packet loss, and high throughputs of mobile nodes during handoff [19]. In SCTP, handover can be divided into Horizontal and Vertical handover schemes. These two types of handover schema depends on the types of access technology used, and the number of network interfaces (NIC) participate during handoff [20].Table 1.3. describe the differences between Horizontal and Vertical handover schemes.

1.6 Why is SCTP Is Better Than TCP

SCTP is better than TCP for a number of reasons [2, 21, 22]:

- SCTP provides more flexibility in certain applications; the best example is the Voice over IP (VoIP) that requires data to be transmitted reliably.
- SCTP is focused on sending as a message because of messages oriented, whereas TCP is focused on sending as a byte because of bytes oriented.
- Byte-oriented structures of the TCP applications have to add their own mark records to save message boundaries, while SCTP use it's own packet structure (chunks).
- SCTP offers additional security features not present in TCP and UDP, which is using the cookie exchange mechanism. This mechanism is very useful for allocation of resources during connection and which reduces the likelihood of a denial-of-service.
- TCP provides a reliable and strictly ordered delivery of data while SCTP uses the principle of multiple streaming on the same connection, which provides a partial order delivery of a logical division of flows .

- SCTP offers some fault tolerance by using a multi-homing feature in order to provide network level redundancy.

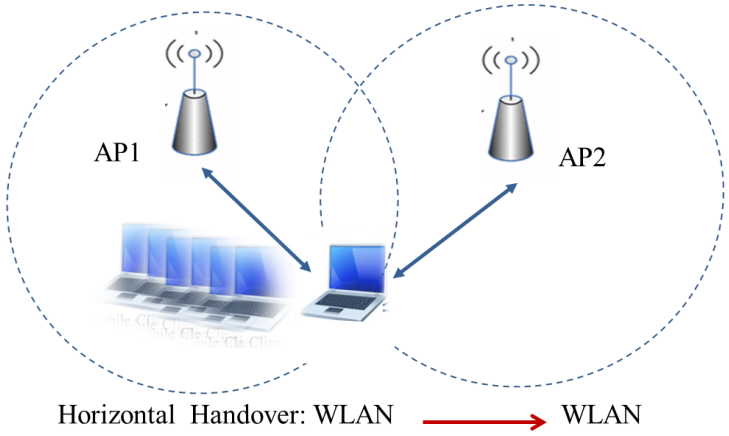

Figure 1.5: Horizontal handover scheme

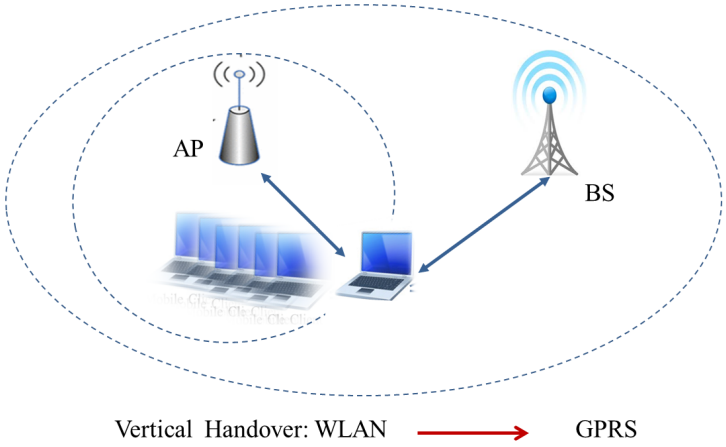

Figure 1.6: Vertical handover scheme

1.7 Thesis Problem

Nowadays, communication networks have become one of the modern technologies. It also widely used among different societies; especially, when we have communication via more than one device through the IP network and when we share some files and videos (through sending and receiving data). When

two devices connect with each other over an IP network there are multiple layers employed for each of the two devices; application layer, transport layer, network layer, data link ayer, and physical layer. TCP is the transport protocol most widely used; however, it is considered insufficient to support all the requirements of modern networks. Therefore, SCTP has been developed to satisfy all the requirements of modern networks that are unavailable with TCP [4].

The enhancement of the SCTP multi-homing featur increased rapidly to perform the mobility function more efficiently. According to recent research that the uses of SCTP Multi-homing between heterogeneous network improves the communication performance (handover latency, throughput, packet loss, and end to end delay) over IP networks without network failure. Implementing SCTP multi homing between homogeneous networks it would also be better to improve the performance of WLANs. However, there are still some aspects that require further significant research in order to analyze the efficiency and the ability of this protocol.

In this dissertation we considered the WLANs an example to evaluate significant effects of background traffic (TCP and SCTP traffic , which leads to a congested network and then causes collisions) with using various channel bit rates, and various mobility speeds on the performance of SCTP multi-homed host by doing some simulation experiments.The major questions that concern us and will be discussed in this thesis are:

- What are characteristics of the paths?
- How will this heavy traffic congestion effect the performance of SCTP Multi-homed host while using different bit rate and mobility speeds?
  - Is there a time delay over the handoff from the primary path to an alternative path? How much time is needed?
  - How will the heavy traffic congestion influence the end-to-end delay?
  - Is there any packet loss? and how did it occured?
  - How will the heavy traffic congestion influence the end-to-end throughput?
  - How will the heavy traffic congestion influence the HO delay time?

1.8 Thesis Approach

This approach includes some strategies that have been done in order to achieve our goal, as follows :

- We have run the simulation experiments 50 times by using the OMNeT++ network simulator tool, in order to present results of two simulation scenarios and then measure the performance metrics.
- We used the basic INET framework of the OMNeT++ environment and its API kernal library
- The methodology that has been used in this project in order to implement the failover/handover mechanism consists of two computers connected to each other. Each of them are configured with two interface cards; the mobile host has two wireless adapters (two NICs) while the destination has two Ethernet card. It also consist of some background traffic hosts (TCP or SCTP host) to study the effect of these hosts on the performance of SCTP multi-homed host under various simulation conditions (detailed information is presented in Chapter 3).
- Thomas Dreibholz Algorithm's with handover management is used to implement a wireless SCTP multi-homing model within OMNeT++ environment.
- We have improved the horizontal handover scheme in order to use it in the proposed SCTP Multi-homing simulation scenarios.

1.9 Thesis Contributions

This thesis includes three contributions that are significant in the area of wireless communication networks:

- Evaluation of the SCTP snd TCP traffic in the wireless LAN network
- Implementation of the wireless SCTP multi-homing model with a handover management in order to improve the communication performance of the congested WLAN.
- Evaluation of the impact of using various channel bit rates and mobility speeds on the performance of the SCTP multi-homed host in the congested network.

1.10 Thesis Organization

The remaining sections of this thesis is organized as follows:

- In Chapter 2, we describe the related work in this area of study.
- In Chapter 3, we present the simulation implementation of the failover (handover) mechanism of the SCTP multi-homing under various situations.
- In Chapter 4, we present our results and analyze the performance of the simulation scenarios that have been performed.
- In Chapter 5 we conclued of the thesis and present some of the future work.

Table 1.1: The differences between MIPv4 and MIPv6 [8, 9]

Mobile IPv4	Mobile IPv6
The need of use the FAg as essential element to assist the MN to connect to the FN	There is no need to use FAg, it is using a plain IPv6 router on the FN
In MIPv4, MN must assign two IP address (HA and CoA)	In MIPv6, MN must assign three IP addresses (HA, LLA, and CoA)
The CoA could be foreign agent care-of address (FAg-CoA) or collocated care-of address (CO-CoA).	All the CoA are collocated care-of addresses (CO-CoA).
In MIPv4, prefers to use (FAg-CoA) because of limitation of IPv4 address space	In MIPv6, it is possible to use (CO-CoA) all the times because of infinite address space in IPv6
The CoA may be acquire via Agent Discovery, DHCP, or manually	The CoA may be acquired via stateless IP adresses auto-configuration, DHCP, or manually
In registration phase MN should be register its CoA in the HAg only	in registration phase MN should register its CoA in both the HAg and corresponding node CN
It can route the packets to the MN via tunneling	It can route the packets to the MN via tunneling and source routing
In MIPv4, MN is unable to inform CN of CoA change	In MIPv6, MN can inform the CN of CoA change directly as part of the protocol
In IPv4 packets are tunneled by using an IP packet encapsulation	In MIPv6 packets are tunneled by using an IP routing header
Route optimization may be difficult sometimes	Route optimization in MIPv6 is easier than MIPv4
To implement route optimization, is required to do some changes to the stack protocol in the CN	To implement route optimization, there is no need to do some changes to the stack protocol in th CN
Because of the route optimization is related to the separate protocol specification in MIPv4 (support Triangle routing)	Because of the route optimization is integrated in MIPv6 (build optimal path between CN and MN to eliminate the problem of Triangle routing)
In MIPv4, the mobility management uses algorithms for purposes of regional registration; it called Mobile IP Regional Registration (MIP-RR)	In MIPv6 , the mobility management uses algorithms for purposes of regional registration; it called Hierarchical Mobile IPv6 (HMIPv6)

Table 1.2: Chunk type and related number to each chunk[14].

Numbers	Chunk Type
0	Payload Data
1	Initiation (INIT)
2	Initiation-Acknowledgement (INIT-ACK))
3	Selective Acknowledgement (SACK)
4	Heartbeat Request
5	Heartbeat Acknowledgement
6	Abort
7	Shutdown
8	Shutdown Acknowledgement
9	Operation Error
10	State Cookie (COOKIE-ECHO)
11	Cookie Acknowledgement (COOKIE-ACK)
12	Explicit Congestion Notification Echo (ECNE)
13	Congestion Window Reduced (CWR)
14	Shutdown Complete
15-254	Reserved by IETF

Table 1.3: Horizontal and Vertical handover schemes

Horizontal handover	Vertical handover
Allows the MN to handoff and change its point of connection across homogeneous networks (same types of networks) via same network interface	Allows MN to handoff and change its point of connection across a heterogeneous networks (different types of networks) via multiple network interfaces
Actually horizontal handover supports single network interface (one NIC) with single IP address at a time	Actually vertical handover supports multiple network
This type of handover supports single-homed mobile nodes (One network connection)	This type of handover supports multi-homed mobile nodes (multiple network connections).
The main characteristic of horizontal handover is to solve the problem of the change of IP address in order to maintain network connectivity.	The main characteristic of vertical handover is not only to solve the problem of the change of IP address, but it also to solve the problem of the change of network interfaces, or QoS characteristics in order to maintain network connectivity.
Horizontal handover mechanism is performed by hiding the change of the IP address (using Mobile IP), or dynamically updating the change of the IP address (using mSCTP) in order to solve the problems that could be occur while the change of the IP.	Vertical handover mechanism is performed by using SCTP Multi-homing feature with capability of dynamic IP address reconfiguration (mSCTP) in order to solve the problems that could be occur while the change of network interface.
Figure 1.5. shows how the MN handoff and establish another connection with new AP after disconnect the connection with the old AP.	Figure 1.6 shows how the MN handoff and establish another connection with new AP before disconnect the connection with the old AP.

Chapter 2

Related work

The purpose behind this chapter is to present some of the research that has been previously accomplished and that is related to our research study. The research gathered within this chapter will be beneficial in the subsequent chapters, and will be used as a foundation for this thesis study. In the last few years, the SCTP multi-homing has be considered one of the most significant methods for the purpose of achieving seamless mobility in wireless network environments. Several research and studies have been done to evaluate the performance of SCTP multi-homed host for mobility and handover mechanism in wired and wireless networks.

In [1], the authors present a performance comparison between TCP and SCTP protocol (in terms of throughput). They are focusing on analyzing the failover mechanisms of SCTP in multi-homed host.

In [23], the authors describe the performance analysis of SCTP multi-homing for wireless networks environment. They found that SCTP-multi-homing can provide better throughput and more robustness in wireless multi-access scenarios.

In [24], the authors investigate the handoff performance of three mobility protocols SIP, MIP, and SCTP within heterogeneous network such as: cellular networks, wired, and wireless local area. They discover that the SIP and SCTP have the lower handoff delay compared to MIP. In[25] the authors describe the performance analysis of SCTP and TCP in regards to Web traffic. This comparison clarifies that SCTP multi-homing can help to improve the throughput and decrease the latency. This is slightly different in our research study since we use FTP traffic instead of Web traffic.

In [26], the authors provide a comparison between TCP and SCTP in wireless networks. Thus, the results were indicated to that SCTP provides better performance than TCP due to its attractive features such as multi-streaming and multi-homing.This is true in our research study since the SCTP multi-homed host have better throughput performance than TCP hosts.

In [27], the authors provide experiments to analyze the performance of FTP over TCP and SCTP in congested network, especially when they transfer data with 4MB or 32MB. In the results, they found that the average throughput of FTP over SCTP multi-homing host is more than FTP over TCP host. This is true in our research study SCTP provide better performance than TCP in our congested network scenarios.

In [28], the authors present a comparison between the performance of TCP, SCTP, and SCTP-CMT. They found that due to multiple paths, the SCTP and SCTP-CMT have a better performance than TCP. This is true in our research study since the SCTP multi-homed host in congested WLAN have better throughput than TCP hosts.

In [29], the authors investigate the performance of SCTP multi-homing handover in a WLAN environment. The results show that increasing the RTO and RTT value for the SCTP multi-homed host allows more time before handover occurs in WLAN environments. This is true in our congested network scenario; the handover delay is increased due to RTO and RTT increase due to increase the intensity of background traffic.

In [30], the authors proposes an improvement handover scheme for mobile SCTP host based on IEEE 802.11b to be more suitable to a WLAN environment. The proposed scheme was performed based on measured the RTT delay time of each path and made a handover decision based on the measurements obtained. The results show that SCTP host can perform handover before path failure occurs, which led to fewer retransmissions were occur and increased the efficiency of the WLAN. This scheme is different from our handover scheme due to we use IEEE 802.11g standard. Also because our handover scheme was performed based on measured the QoS (signal strength) of access points and sensitivity threshold value of SCTP multi-homed host before a handover occur.

In [31], the authors present a comprehensive review of SCTP multi-homing. They focused in this research on handover management.

In [32], the authors analyze the SCTP multi-homing performance between UMTS - WLAN to measure the handover behavior in terms of throughput, transmission delay, and handover effective-

ness.The results show, accurate SCTP multi-homing parameter setup can significantly eliminate the data transmission interrupts and decrease the handover delay. Similar results we are obtained in our research study since we analyze the SCTP multi-homing performance between WLAN - WLAN.

In [33], the authors evaluate the SCTP performance over IEEE 802.11 WLANs to clarify the impact of using different SCTP receiver side window sizes and the different number of hops between source and destination the on throughput. The results demonstrate that throughput of SCTP degrades when the number of hops increases and increasing the window size does not help to increase the throughput in case of the hidden node problem and the exposed node problem.

In [34], the authors present a comparison between the performance of using SCTP and TCP for FTP file transfers (10 files with 200KB or 1MB each). The results indicate that SCTP is better suited for an FTP file transfer in wireless network due to SCTP significantly reduces file transfer time and more robust to losses. Similar results we are obtained in our research study since the TCP hosts take more time than SCTP to uploading FTP file.

In [35], the authors analyze the throughput performance of IEEE 802.11b WLAN with one access point via OPNET simulator tool. This analysis was based on some network parameters such as the data-rate, buffer-sizes. In the results, they found that with increase the data rate in the wireless network (1Mbps, 5.5 Mbps, and 11Mbps), the throughput will be increased and packets will be delivered more accurately. In addition, for a using small size of buffer, when the data-rate was increased, the throughput reduced due to packets drop due to buffer has no space to accommodate more packets. Similar results we are obtained in our research study from OMNeT++ simulator since we used date rate (24 Mbps, 36Mbps, and 54 Mbps) of IEEE 802.11 g standards.

More significantly, this dissertation demonstrates the impact of increasing the background traffic (TCP and SCTP traffic) on the performance of SCTP multi-homed host between two congested WLANs. In addition, we believe that, this is the first research that studies the impact of using various channel bit rates and various mobility speeds on the SCTP multi-homed host between congested WLANs.

Chapter 3

Methodology and Simulation Experiments

Introduction

In the previous chapters, we discussed the SCTP overview and SCTP Multi-homing feature. In addition, we presented the handover mechanism and mobility protocols of the two different layers: network layer (mobile IP) and transport layer (SCTP Multi-homing/ mSCTP). In this chapter, we will present the simulations implementation along with the performance analysis of the results. At first, we will introduce an overview of the OMNeT++ 4.4 network simulator tool and the related modules used in the simulation scenarios. Then, we will describe the simulation experiments, including the setup, scenarios, configurable parameters, and network topologies. These simulation scenarios clarify the performance of SCTP Multi-homing host with regards to the handover latency, packets loss, end-to-end delay, and average throughput when traveling across congsted WLANs.

3.1 OMNeT++ Network Simulator

OMNeT++ (Objective Modular Network Testbed in C++) is an open source network simulator tool that was created in 1997 on the Linux and Mac OS/X Platforms. Since then, it was expanded

to include the Windows Platform. OMNeT++ is a discrete event simulation framework (DES) that is designed to be used in various problem domains which include of the following [36, 37]:

- Modeling of communication networks such as: Wired and WLANs Networks, Mobile Ad-hoc Networks, Sensor Networks, Vehicular Networks, Cellular Networks, and Cloud Computing.
- Modeling of queuing systems
- Modeling distributed system
- Protocol Modeling such as TCP, UDP, SCTP, IPv4, IPv6, PPP, etc.

Since its creation, the OMNeT++ simulation tool has become one of the most popular simulation tools with Graphic User Interface (GUI) support. The main features of OMNeT++ is the component-based architecture in which the components (modules) are assembled to provide large components or models that allow users to build their own network simulations using a high-level language NED. The implementation of the modules is defined in the NED language that is programmed in C++. New modules can be derived from basic libraries or classes of other modules [38, 39].

There are two categories of modules in OMNeT++ simulator: simple modules and compound modules as shown in Figure 3.1. One simple module can for example be an Ethernet implementation or a SCTP implementation. An Ethernet implementation coupled with an SCTP implementation can form either a StandardHost or a WirelessHost which is individually characterized as compound modules. Each of these modules contains various numbers of simple modules such as Ethernet interfaces, the network layer (IPv4 or IPv6), the transport layer (TCP, UDP, or SCTP) or corresponding applications [40].

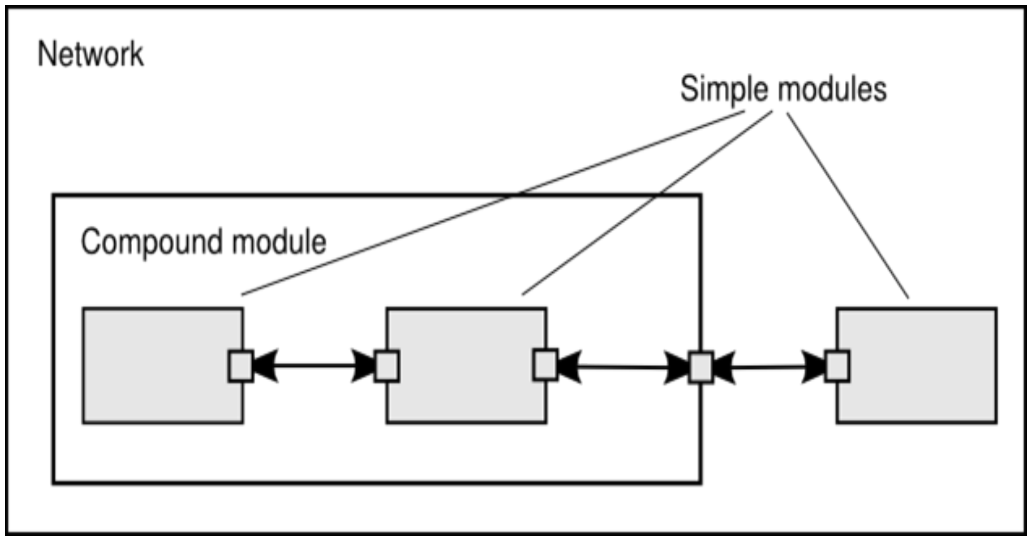

Figure 3.1: OMNeT++ model structure [source [36]].

3.1.1 OMNeT++ Environment Components

The OMNeT++ simulator incorporates components for all phases of the simulation research, including model design, simulation execution, and data analysis.These components are considered to be main tools of the OMNeT++ simulation environment, which enables the users to implement, simulate, and analyze their scenarios more efficiently and flexibly.The description of each OMNeT++ component is listed below [36].

- **Simulation kernel library (SKL):** Kernel libraries are the basic component of OMNeT++, which allow users to compile and debug the simulation algorithms.
- **NED Source editor:** NED is important component that helps users to create NED file with textual descriptions as illustrated in Figure 3.2. This file describes the simple and compound modules (nodes), the gates that connect these nodes as well as their general parameters.
- **Graphical NED editor:** The GNED is a graphical user editor that enables users to create NED file without textual descriptions through graphical interface as described in Figure 3.2.
- **GUI for simulation execution (Tkenv):** Tkenv is a GUI with simulation execution that is able to display the network graphics directly. The main purpose of Tkenv is to visualize the node

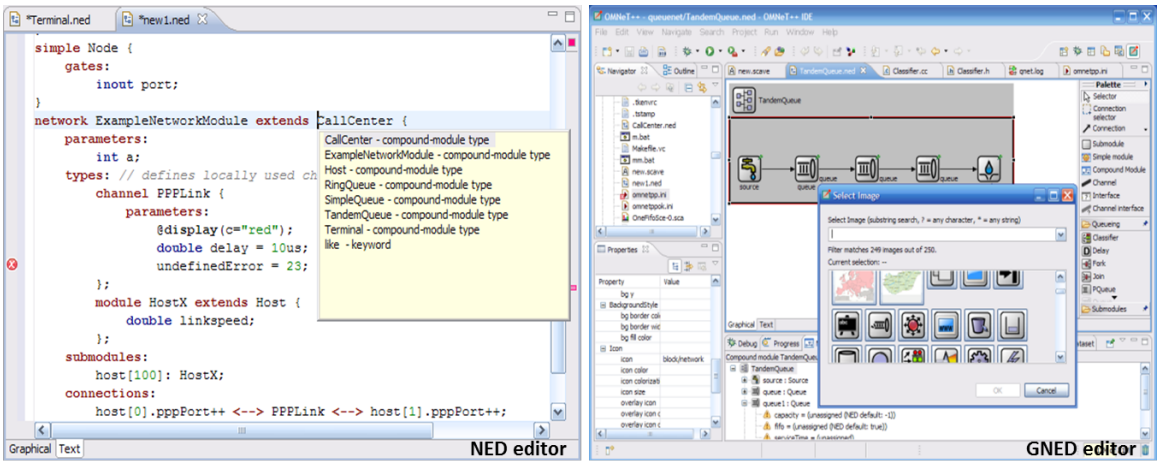

Figure 3.2: NED Source and GNED Graphical editor in OMNeT++ [source [41]]

and track the route of packets graphically. Furthermore, it provides users the ability to start/stop simulation execution, and possibly allow the user to understand and analyse the node state and events between nodes.

- **CLI for simulation execution (Cmdenv):** Cmdenv is a command-line user interface (CUI) with simulation execution in the OMNeT++. The Cmdenv achieves the same purpose of Tkenv which visualizes the node and tracks the route of packets as textual messages without any graphics. It also can be used easily to execute all simulation specifications that are described in the configuration file.

- **Analyzing and Visualizing simulation results within OMNeT++:** There are four result files that help users with the visualization and statistical analysis of simulations through IDE of OMNeT++: anf.file, logs.file, vec.file, and sca.file. The "anf file" is the most important results file; it includes two types of results: vector and scalar.

  The vector is a type of result that is inherited from vec.file which records the directed information like nodes state, channels state, messages exchange, and route of packets, etc. The second type is the scalar result that is inherited from sca.file which records the numeric information like time average, number of packets drop/loss, delay, and throughput.

The logs file is a second important results file in OMNet++ that can be considered as a sequence chart and event log results file. It illustrates the movement of nodes and how the messages exchange between the various nodes in the network during the simulation as shown in Figure 3.3.

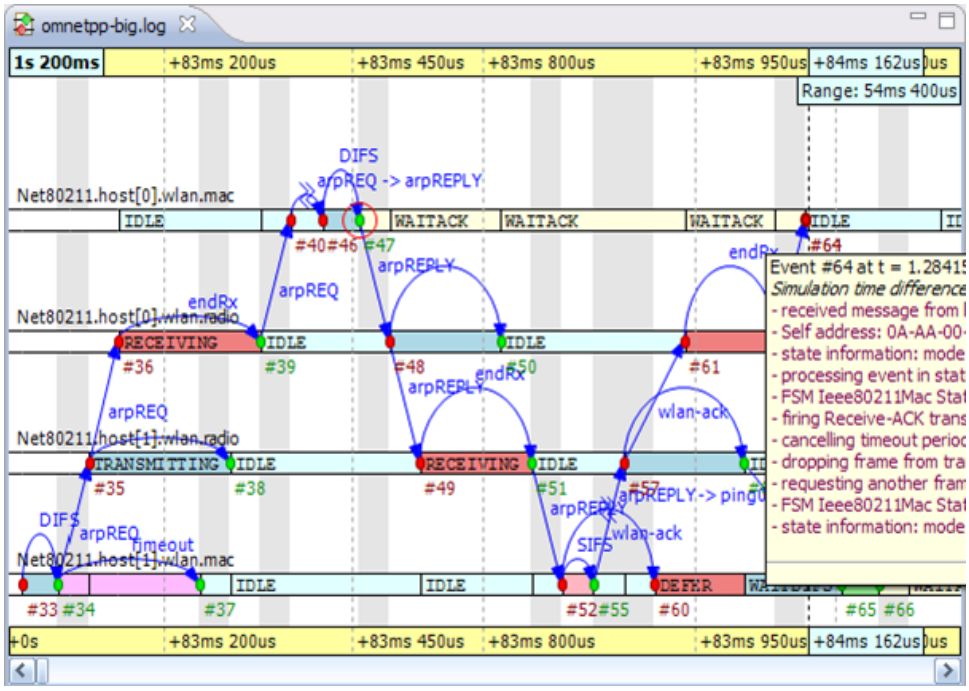

Figure 3.3: The Sequence Charts and events log output in the OMNeT++ [source [41]].

3.2 Simulation Requirements

3.2.1 INET Framework

The INET Framework is an open source framework that is also referred to as the Mobility Framework. This framework is commonly used for communication networks, which comprise of the mobility and handover functions, in order to achieve the functions in a transparent and efficient way. It has been developed to include some models that are useful to simulate wired and wireless network such as TCP, UDP, SCTP, ICMP, Ethernet, IPv4, IPv6, PPP, 802.11, 802.16, and some routing protocols [42, 38].

In 2013, INET Framework was developed rapidly, which led to the publishing of a new version of INET Framework. INET-2.2.0 Framework stable release is available as of August 23, 2013 and includes several bug fixes, important changes and additions to the SCTP Model, especially those associated with SCTP Multi-homing [37].

In our SCTP Multi-homing project, version 4.4 of OMNeT++ and the latest version of INET Framework (INET-2.2.0) were used because they provide some SCTP extensions that were included in our simulation in order to improve the performance of SCTP Multi-homing (see the details in Subsection 3.2.2). Version 2.3.0 of INET came out on March 27, 2014 while this project was in progress.

3.2.2 SCTP Models

SCTP is the newest transport layer protocol that is used specifically for network fault tolerance, as mentioned in Chapter 1. In 2008, the SCTP model was developed and included in the INET Framework for the purpose of supporting both IPv4 and IPv6 as network layers and multi-homed hosts. However, within the recent two years, several improvements and extensions have been released by the IETF and then incorporated into the SCTP model in order to improve the SCTP Multi-homing within the INET Framework [43]. In our SCTP Multi-homing project, we implemented the SCTP Multi-homing across the network and transport layers by using the Autoconfiguration IP Address Feature that was part of the INET-2.2.0 Framework and the OMNeT++ SCTP Module that was implemented by Thomas Dreibholz Algorithm's [43]. The next section delves further into the Autoconfiguration IP Address Feature.

Autoconfiguration IP Address Feature

Dynamic Address Reconfiguration was made to be a significant SCTP feature in this project so as to enhance the implementation of SCTP Multi-homing (mSCTP) and acquire more realistic results. As indicated in chapter 1, it can also be called ADD/DELETE- IP extension. As per [37], the API defined in the INET-2.2.0 enables the SCTP endpoints to reconfigure (i.e., add, change, or delete the IP addresses) the IP address information that is being used during the lifetime of an SCTP association dynamically.

3.3 Simulation Setup

This section explain the simulation setup of our experiments based on OMNeT++ kernel API. The SCTP Multi-homing handover in this research is conducted by simulating Mobile Client (MC) that perform a handover between two congested WLANs.

3.3.1 Kernel Library

INET is the main library that was utilized in all simulation scenarios in this research. The NED files of the modules (including **module.cc** and **module.h**) that were imported from INET library and incorporated within our NED file (**SCTP2NIC.ned** file) are listed as follows:

- **import** inet.networklayer.autorouting.ipv4.IPv4NetworkConfigurator;
- **import** import inet.world.radio.ChannelControl;
- **import** import inet.nodes.ethernet.Eth10M;
- **import** import ned.DatarateChannel;
- **import** import inet.nodes.inet.Router;
- **import** import inet.nodes.inet.StandardHost;
- **import** import inet.nodes.inet.WirelessHost;
- **import** import inet.nodes.wireless.AccessPoint;

3.3.2 Wireless LANs

The IEEE 802.11 standards supports using of multi-homing technology in order to select appropriate transmission channel before handover process according to differences in the channel quality and to achieve greater wireless communication efficiency [44]. The WLAN model that has been defined in OMNeT++ was used in this research to implements all simulation scenarios based on the specifications of IEEE 802.11g standard. All simulation experiments are configured with IPv4 configuration addresses and five OMNet++ simple modules which consist of two routers (R1, R2) that support various types of links and two access points APs (ap1, ap2) that support multiple wireless radios and ethernet ports.

The aforementioned APs have been configured in the **omnetpp.ini** file as following: ap1 [MAC address 10:00:00:00:00:00] works on channel 1 and ap2 [MAC address 20:00:00:00:00:00] works on channel 6. The power transmitted by the APs and MC are determined as 2 dBm in the configuration file. The transmission rate of the wireless LAN that was utilized is 54 Mbps.

The WLANs model in OMNeT++ defines two threshold values: Sensitivity threshold and Active Scanning threshold value. The sensitivity threshold value is referred to the minimum level of signal strength that the NIC card can detect. The sensitivity threshold value that we have used in the simulations is -90 dBm. Further, the active scanning threshold value is referred to the signal level of the MC when starts scanning for appropriate ap to perform a WLAN to WLAN handover. The MC will disconnect the connection with ap1 and connect with ap2 when the signal strength value is reaches the minimum threshold value (sensitivity threshold level). The active scanning threshold value that we have used in the simulations is -80 dBm.

3.3.3 Node Base

These experiments are configured with two OMNet++ compound modules. The aforementioned modules are referred to the standard host (SCTP server/Dst), which is located in the wired network and wireless hosts (SCTP client/MC), which is located in the wireless network. The Dst operates the SCTP application and works as a SCTP Multi-homed host, is equipped with two Ethernet interfaces [eth0: 172.17.164.1/24] and [eth(1): 172.17.165.1/24] to connect to MC. The MC is a SCTP Multi-homed host equipped with two NICs [wlan0: 10.1.1.1/24] and [wlan1: 10.2.1.1/24] that travels between two WLAN congested networks as per the configuration file **(omnetpp.ini file)**.

The MC interface [wlan0:10.1.1.1/24] is associated with ap1 over radio link (54 Mbps). This ap1 is in turn connected to the gateway R1 [eth0:10.1.1.2/24] within WLAN1 to reach the Dst network [172.17.164.0/24] and send data. Whereas the Dst interface[eth0:172.17.164.1/24] is connected to the gateway R1 [eth1:172.17.164.2/24] through Ethernet link (100 Mbps) for access to the wireless host network [wlan0:10.1.1.0 /24]. In contrast, the MC uses the second interface [wlan1: 10.2.1.1/24] as an alternative path to attach the ap2 which is in turn connected to the gateway R2 [eth(0): 10.2.1.2/24] in the WLAN2 over the same radio link (54 Mbps) in order to reach the Dsts network [172.17.164.0/24].Per-

haps, the Dst uses the second interface [eth(1): 172.17.165.1/24] which is directly connected to the gateway R2 [eth(1): 172.17.165.2/24] through the Ethernet link (100 Mbps) for access to the host network [10.2.1.0/24].

It should be noted that Channel Control (simple module) is an essential node in every wireless network topology in OMNeT++. It is responsible for controlling the determinants of Wi-Fi such as channels, frequency range, etc. All these simple and compound modules with their characteristics together form the network that is called: **sctpwifi2nic** within the simulation scenarios that were presented within dissertation as illustrated in Figure 3.4.

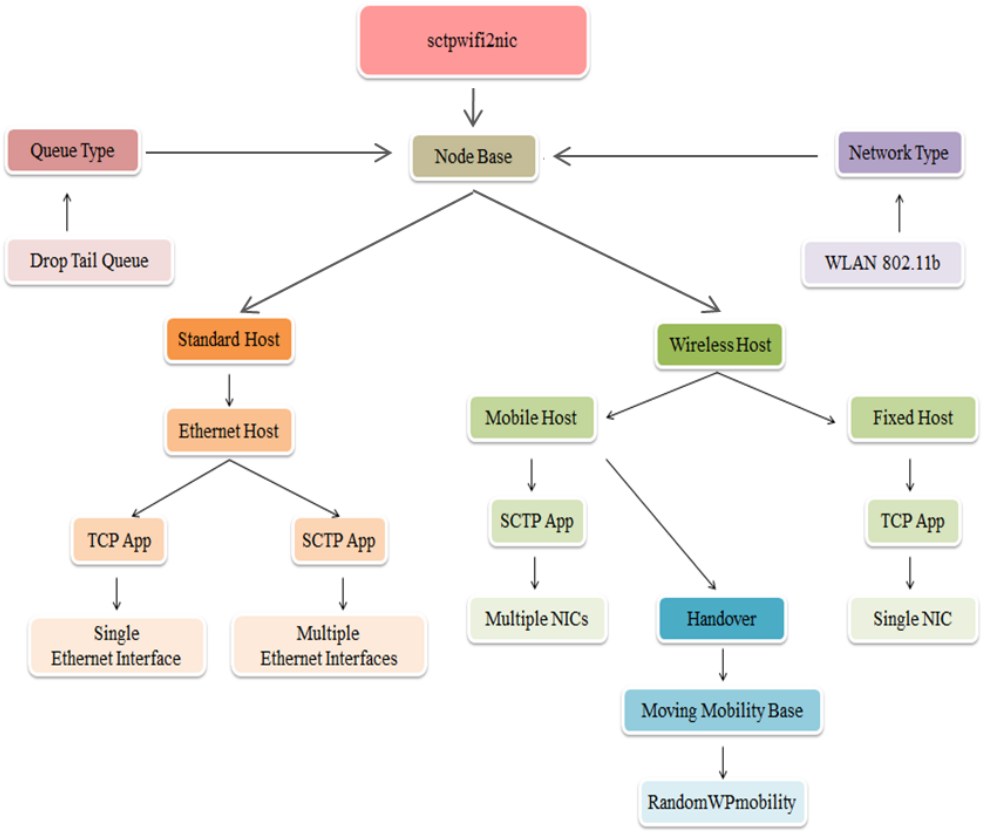

Figure 3.4: Illustrates the Architecture of proposed SCTP Multi-homing

3.3.4 Handover Algorithm

In general, the SCTP implements two types of handover schemes: Vertical and Horizontal handover scheme, as discussed previously in Chapter 1. In this research, we improve the uses of the horizontal handover scheme by combining some features of the vertical handover scheme (multi-homing/multiple network interfaces and connection at the same time). This combination gave us the ability to use mSCTP instead of use MIP for the mobility function.

The improvment of the horizontal handover scheme aims to utilize the SCTP Multi-homing feature efficiently, to reduce the time that NICs takes to scan for the neighboring APs when the MC roams across WLANs. This scheme enables MC to utilize its interfaces (wlan0 and wlan1) through established connection with ap2 before terminating its previous connection with old ap1 for a few moments; in order to avoid communication breakdown and the need to re-establish a new connection.

Actually, the handover operation may take place in various instants. In this project, the handover occur according to WLAN - WLAN handoff criteria, which is based on the signal strength and threshold value as following:

1. The MC is in WLAN1 network.

2. If $RSS \leq$ *Active Scanning Threshold* (Th_1), probe process of NIC starts scanning for appropriate ap.

3. RSS (Received Signal Strength) is measured.

4. If $RSS \geq$ *Sensitivity Threshold* (Th_2) , then the MC continues using first interface (wlan0) .

5. Repeating step2 each slot of time.

6. If $RSS <$ (Th_2), then the MC handoff from WLAN1 to WLAN2 and uses second interface (wlan1) after T seconds

3.4 Simulation Experiments

In order to evaluate the effect of increasing the background traffic intensity on the performance of MC (handover latency, average throughputs, packet loss, and average end-to-end delay) roams between WLAN congested networks without interrupting any ongoing data. Two different scenarios have been carried out using the OMNeT ++ network simulator. The main difference between these scenarios is the background traffic types (TCP or SCTP). In the first scenario, TCP background traffic has been implemented to clarify the effect of the background traffic intensity on the performance of MC under different channel bit rates in the network. In the second scenario, SCTP background traffic has been implemented to clarify the effect of the background traffic intensity on the performance of MC which moves among congested WLANs with different mobility speeds. The aforementioned simulation scenarios will be described in detail in Sections 3.4.1 and 3.4.2.

3.4.1 Simulation Scenario A

This scenario was used to demonstrate the impact of increasing background traffic generated by TCP hosts on the performance of MC roams between two WLAN congested networks under different channel bit rate (54, 36, and 24 Mbps). It was implemented by using two SCTP Multi-homed host. The SCTPClient (MC) was placed in the wireless network and SCTPServer (Dst) was placed in the wired network. It was also done by using a different number of the stationary TCP hosts (H_i... H_{n-1}, H_n) which were placed on the WLAN, a various distance away. Moreover, the Dst operates two different transport applications; TCP and SCTP application. It works as the TCP application and it is equipped with a single interface to connect with stationary TCP hosts through only one IP address each. Nevertheless, the Dst also works as a SCTP application and it is multi-homed host equipped with two Ethernet interfaces to connect to the MC.

The router hosts (R_1, R_2) used between hosts and Dst are a drop tail router with a queue size of 50 packets and static routing. The MC randomly moves between two WLAN congested networks, during uploading 100 MB file with mobility speed of 4.5 km/h. The heartbeat time used by MC to adjust the accessibility to the second path is 10 ms. Meanwhile, the other TCP hosts continue send its data to the Dst. Furthermore, each TCP host is also uploading 100MB file and generates an elastic

traffic (FTP) through the associated AP with 54 mbps to the Dst and causing channel congesttion for MC. Actually, when MC wants to participate in an SCTP session or handoff to another network it must contend with other TCP hosts to take its turn of communication through a shared wireless channel. This scenario was accomplished by running multiple times and increasing the number of TCP hosts starting from 5 up to 30 to study the effect of increasing hosts on the performance of MC (handover latency, throughputs, packet loss, and end to end delay) as indicated in my results analysis in Chapter 4.

Configurable parameters

In order to be able to test this simulation scenario under specific network conditions, it is necessary to specify some parameters needed in this experiment and add them in the **omnetpp.ini file**. The main parameters and their values that has been used in this scenario are the default value derived from the standards and several hypotheses that might be required for each scenario as shown in Table 3.1.

Network Topology

In the first scenario, the simulation based on TCP background traffic has been selected. This simulation was modeled, using the OMNeT++4.4 network simulator. All the hosts located in both WLANs are stationary TCP hosts except the MC is an SCTP multi-homed mobile host which travels between congested WLANs.Both TCP and SCTP hosts send packets through links R_1 and R_2 to the destination (Dst) host as displayed in Figure 3.5.

Table 3.1: The Configurable Parameters for test case of TCP background traffic

Parameters	**Specifications**
Simulation Time	210s
Simulation runs repetition	50 Times
Network Area Size	600m× 400m
No. of TCP stationary nodes	5, 10, 15, 20, 25, 30
No. of SCTP multi-homed nodes	1
No. of streams	2
SCTP size request	1456 Bytes without header
SCTP header size	96 Bytes
TCP header size	144 Bytes
MTU size	1500 bytes
heartbeat time	10 ms
RTO	3s
Initial SCTP arwnd	65536 byte = 64 Kbyte
Initial SCTP ssthresh	ssthresh = arwnd = 64 Kbyte
Initial SCTP cwnd	2*MTU
Max Round Time Trip (RTT)	100ms
Wireless operation mode	IEEE 802.11g
propagation delay	1 ms
Carrier Frequency	2.4 GHz
Data rate	Wi-Fi = 54, 63, and 24 Mbps Wired=100 Mbps
Bandwidth speed	100 Mbps
Transmitter power	2.0 mw
Queue type	Drop Tail Queue
Queue Limit	50 packets
Mobility model	Random way point
MC Mobility Speed	(4.5 km/h)
MC Traffic Source	FTP (uploading file of 100MB)
TCP hosts Traffic Source	FTP (uploading file of 100MB)

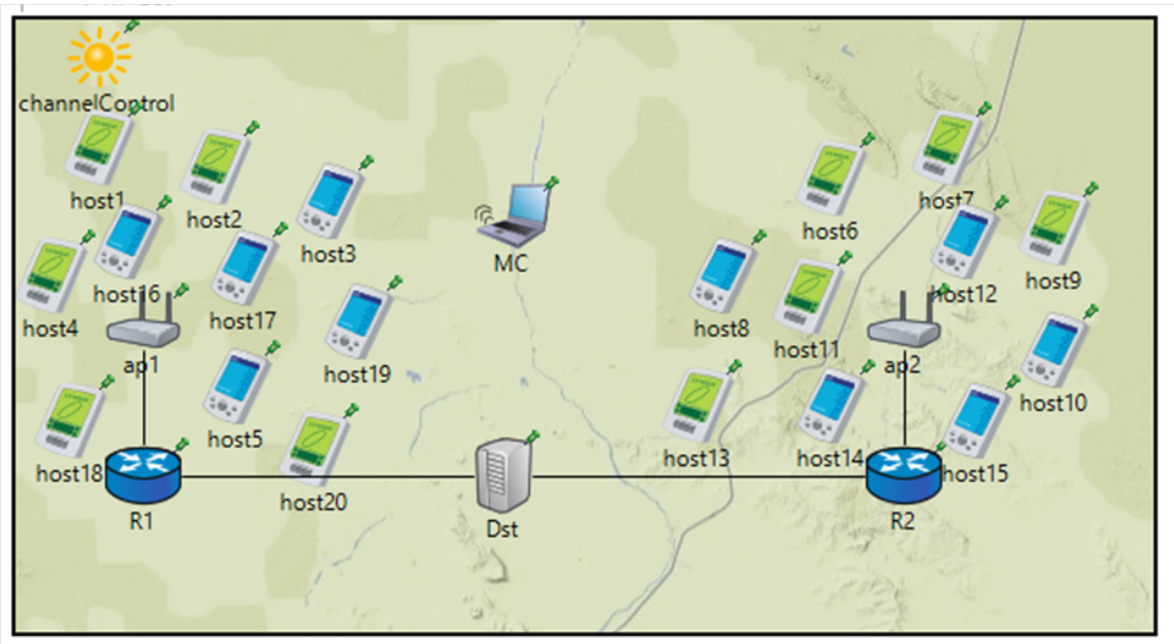

Figure 3.5: Illustrates simulation topology for studying the impact of background traffic (i.e. ten stationary TCP hosts in each WLAN) on the MC

3.4.2 Simulation Scenario B

This scenario was used to observe the impact of increasing background traffic generated by SCTP hosts on the performance of MC which travels between two congested WLANs with various mobility speeds. The traffic was made by using various numbers of SCTP single homed hosts as the stationary SCTP hosts (H_i... H_{n-1}, H_n). The Dst in this scenario operates only SCTP applications. It is equipped with two Ethernet interfaces to connect to the SCTP single-homed host (Asymmetric multi-homing mode) and SCTP multi-homed host (Symmetric multi-homing mode as described in Chapter 1).

The router hosts (R_1, R_2) used in this scenario are the same type used in the previous scenario. The routing in this scenario was achieved via static routing. The MC randomly moves between congested WLANs while uploading 100MB file to the Dst with different mobility speeds has been selected as follows: Random Walking Speed (4.5 km/h), Random Brisk Speed (6.5 km/h), and Random Cycling Speed (15.5 km/h). The MC heartbeat time needed to monitor the accessibility to the second path before switching to the second network is 10 ms.

All SCTP hosts in this scenario are uploading file of 100MB to the Dst to generate FTP traffic through the AP with 54 mbps. Similar to scenario A, when MC wants to participate in an SCTP session or handoff to another network, it should contend with other SCTP hosts to take its turn of communication through a shared wireless channel. Running the scenario multiple times and increasing the number of SCTP hosts from 5 up to 30 it will be possible to get significant results. These results will clarify the impact of the traffic on the performance of MC (handover latency, throughputs, packet loss, and end to end delay).

Configurable parameters

To test this simulation scenario under different condition from scenario A, it will use the same parameters defined in Table 3.1; in addition to changes on some of the earlier hypothesesas as summarized in Table 3.2.

Network Topology

In the second scenario, the simulation based on SCTP background traffic has been selected.This simulation was also modeled using the OMNeT++4.4 network simulator. The SCTP hosts were placed in the WLANs and within a yellow circle representing the SCTP single homed hosts, which work as the stationary hosts and generate background traffic. However, the MC represents an SCTP multi-homed mobile host, which moves from WLAN 1 toward WLAN 2 as illustrated in Figure 3.6. All SCTP hosts send packets through links R_1 and R_2 to the destination (Dst) host.

Table 3.2: The Configurable Parameters for test case of SCTP background traffic

Parameters	**Specifications**
Simulation Time	210s
Simulation runs repetition	50 Times
Network Area Size	600m× 400m
Type of SCTP node	single homed and multi-homed
No. of SCTP single homed nodes	5, 10, 15, 20, 25, 30
No. of SCTP multi-homed nodes	1
No. of streams	2
SCTP size request	1456 Bytes without header
heartbeat time	10 ms
RTO	3s
Initial SCTP arwnd	65536 byte = 64 Kbyte
Initial SCTP ssthresh	ssthresh = arwnd = 64 Kbyte
Initial SCTP cwnd	2* MTU
Max Round Time Trip (RTT)	100ms
Wireless operation mode	IEEE 802.11g
propagation delay	1 ms
Carrier Frequency	2.4 GHz
Data rate	Wi-Fi = 54 Mbps Wired=100 Mbps
Bandwidth speed	100 Mbps
Transmitter power	2.0 mw
Queue type	Drop Tail Queue
Queue Limit	50 packets
Mobility model	Random way point
MC Mobility Speed	4.5, 6.5, and 15.5 km/h
MC Traffic Source	FTP (uploading file of 100MB)
SCTP hosts Traffic Source	FTP (uploading file of 100MB)

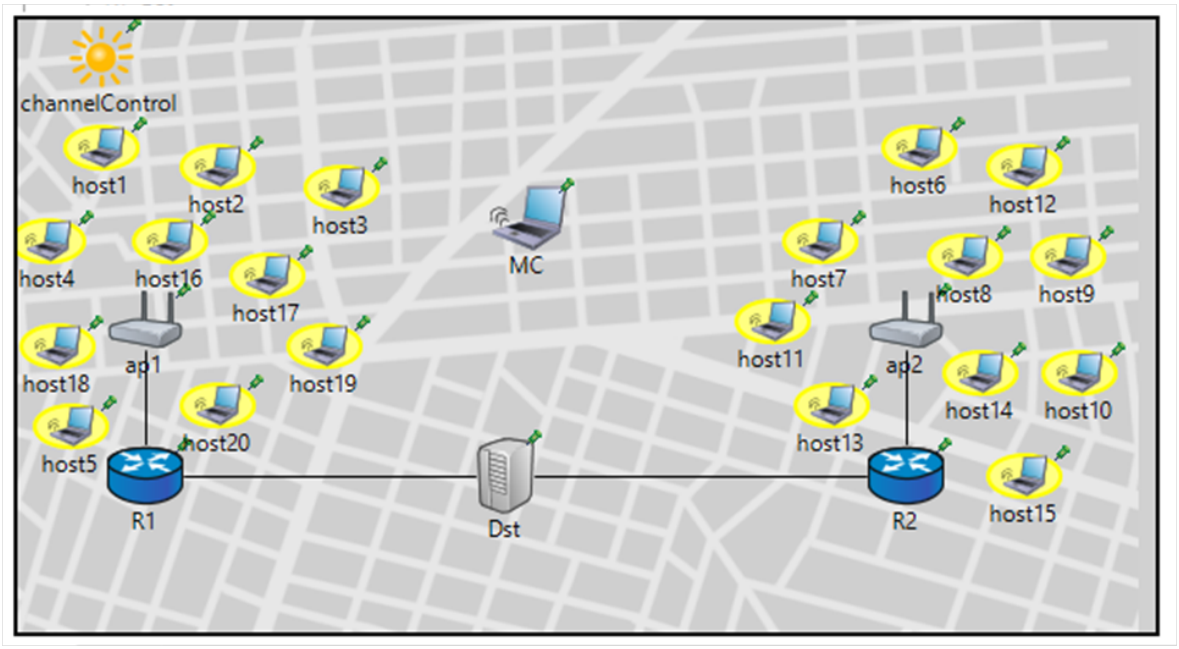

Figure 3.6: Illustrates simulation topology for studying the impact of background traffic (i.e. ten SCTP single homed hosts in each WLAN) on the MC.

3.5 Simulations Description

To measure and evaluate the MC performance in both simulation experiments, it is significant to understand a general view of how SCTP multi-homed hosts (MC) behaves under network conditions in both simulation scenarios. Therefore, when the MC is located in the WLAN1 coverage network, it starts discovering for neighboring access point in order to connect to WLAN1 network. Both APs (ap1, ap2) periodically broadcast beacon frames to all neighboring devices based on a specific parameter value *(**.ap*.wlan[*].mgmt.beaconInterval = 100ms)* that has been mentioned in the **omnetpp.ini file** configuration file. The MC listens carefully to the beacon frame of both APs and scans for channel 1 to connect to either ap1or ap2.

After the MC connecting to ap1 it will use its interface wlan0 as a primary path. In addition to that, the MC establishes an SCTP association with the Dst host at $\mathbf{T}_{Asso1}$, which means sending an INIT Chunk by the MC, and replying with an INIT-ACK Chunk by the Dst. While it continues this phase via sending a COOKIE Chunk by the MC, and replying with a COOKIE-ECHO Chunk by the

Dst until finishing the association establishment phase, as discussed in Chapter1. Indeed, after SCTP initiation at $\mathbf{T}_{Snt1}$, the MC uses the interface wlan0 to start sending the packets (data chunks) to the Dst by using the path that was determined previously as the primary path.

Since the MC starts sending its packets, the background traffic hosts (TCP or SCTP) are connecting with APs and start sending data to the Dst. After a while, MC starts moveing toward the WLAN2 coverage network while continuously using the interface wlan0 to sending the remaining data packet the Dst. Actually, when the MC arrives into the overlapping region, the MC receives the beacon frames from the ap2 and uses the interface wlan1 to scan for channel 6 to reach the ap2. In this case, the benefit of multi-homing technology appears.

The MC has become a multi-homed host (using multiple NICs/path). In this time, the MC maintains its old active connection with the ap1 and adds the new IP address that is obtained from the ap2 into the SCTP association. This is definitely achieved by sending an ASCONF Chunk to the Dst at $\mathbf{T}_{ADD}$ while Dst returns an ASCONF ACK Chunk to the MC. In this time, it should be notice that the background traffic hosts (TCP or SCTP) are still sending data packets to the Dst. Furthermore, the MC will continue sending its data packets through wlan0 while it is monitoring the reachability with the ap2 (if the connection is still active) by exchanging HEARBEAT chunks and HEARBEAT-ACK chunks between MC and Dst.

Subsequently, in order to determine the best path and to continue sending the data packets, the MC will check the Quality of Service (QoS) of both paths (access point signal strength and threshold values as described in section 3.3.4), in each slot of time under a condition. This condition is that the received signal strength value is less than the Sensitivity threshold values, as follows:

$$\mathbf{RS}_{strength} < sen\mathbf{T}_{hresholds} \tag{3.1}$$

Thus, when the MC detects that the signal strength value is less than the threshold value. Obviously, the MC informs the Dst that it will use the second path (wlan1) as a primary path. This will be achieved by sending an ASCONF SET Chunk at $\mathbf{T}_{SET}$ and then the Dst replay with an ASCONF

ACK Chunk to the MC. At this moment, the MC handoff at $\mathbf{T}_{HO}$ to the WLAN2 network and uses the second path that connects with the ap2 to continue sending data packets to the Dst. Thus, at $\mathbf{T}_{DEL}$ the MC will send an ASCONF DELET Chunk to the Dst to delete the old IP address from the SCTP association. After deletion step, the Dst replays with an ASCONF ACK Chunk to complete the process of deleting an old IP address. Figure 3.7 describes the three steps that the MC must be performe before handoff and start using the wlan1 to send its data packets through the ap2 to the Dst.

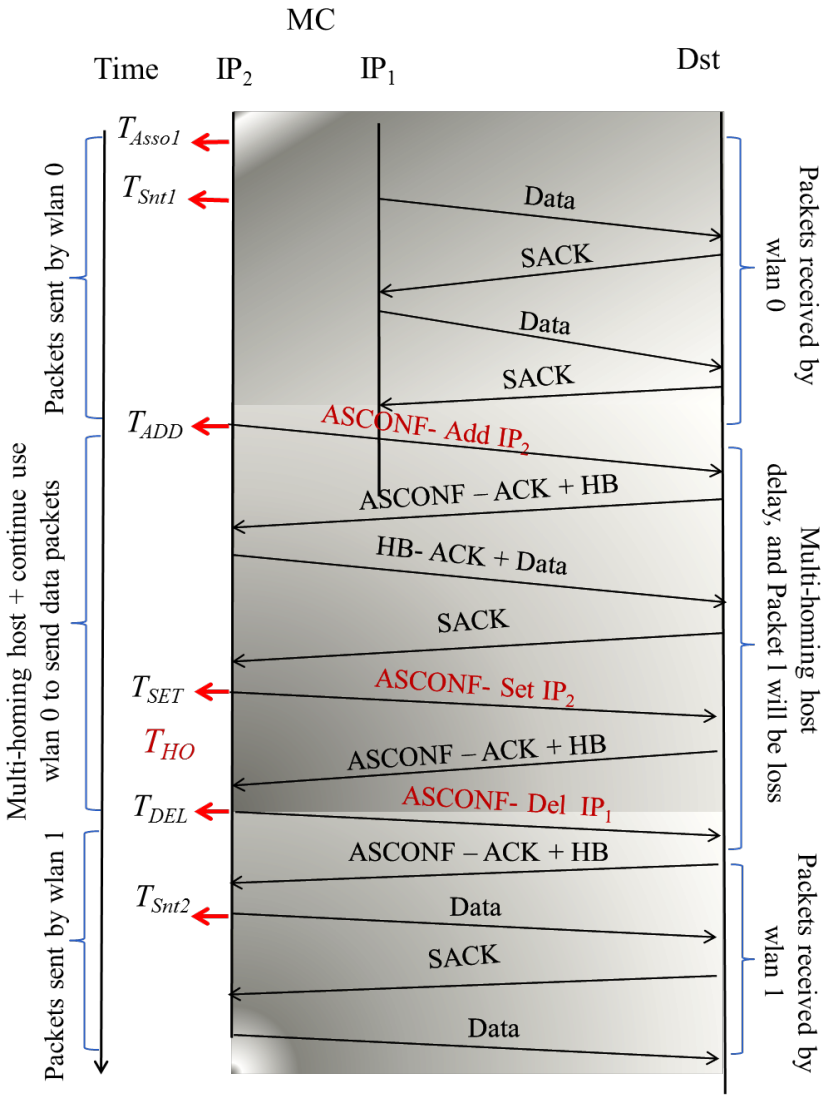

Figure 3.7: Timing chart of the SCTP multi-homed mobile host (MC)

Chapter 4

Simulation Results and Discussion

Introduction

The results presented in this chapter were obtained by running the simulation scenario multiple times and increasing the intensity of background traffic on the networks and get the mean and standard deviation value of each performance metric (as listed in section 4.2). As mentioned previously, both simulation scenarios were performed with network capacity of 54 Mbps by uploading 100 MB FTP file by each host with a transmission time of 200s within the simulation time 210s. The random waypoint mobility model is used for both simulation scenarios in a rectangular filed of 600m x 400m with the various number of hosts to gather reasonable results. The performance analysis of simulation results indicate that increasing the background traffic (number of TCP or SCTP hosts) causes more congestion in the network. Moreover, data packets can "collide" and this affects the MC performance (increases handover delay, decreases throughputs, increases packets loss rate, and increases end to end delay). How we obtained these results and what the differences are between these two simulation results? We will explain each of them in detail in the next sections with a simple comparison between them.

4.1 Performance Metrics

We performed simulation experiments using a varied set of the metrics for measuring the performance of SCTP Multi-homed host roams between WLAN congested networks.

- **Average End-to-end Throughput**

  End-to-end throughput is the rate of successful message (packet) delivery over a communication channel per unit (bits/sec).

- **Handover Delay**

  Handover delay is the time that taken by host to handoff and utilizes another access point

- **Average End-to-end Delay**

  End-to-end delay refers to the time taken for a packet to be transmitted across a network from source to destination.

- **Packet Loss**

  Packet loss occurs in the wireless link that is located between two nodes due to buffer overflow, which means the size of the buffer become less than the flow of packets into the buffer. On the other hand, link capacity [α] is less than rate the flow of packets [λ] .

$$\alpha < \lambda \tag{4.1}$$

4.2 Scenario A: Results and Performance Analysis

Average end-to-end throughput

In this section, we will evaluate the wireless throughput of the MC, which moves between congested WLANs. We will also demonstrate how the MC will be affected with the intensity of background traffic in the network under different channel congestion scenarios and channel bit rates of 802.11g: 54 Mbps, 36 Mbps, and 24 Mbps. Congestion is done by increasing the number of stationary TCP hosts that generate FTP traffic in our simulation. As illustrated in Figure 4.1, the X-axis shows the number

of background traffic hosts whereas the Y-axis shows the average end-to-end throughput of the MC.

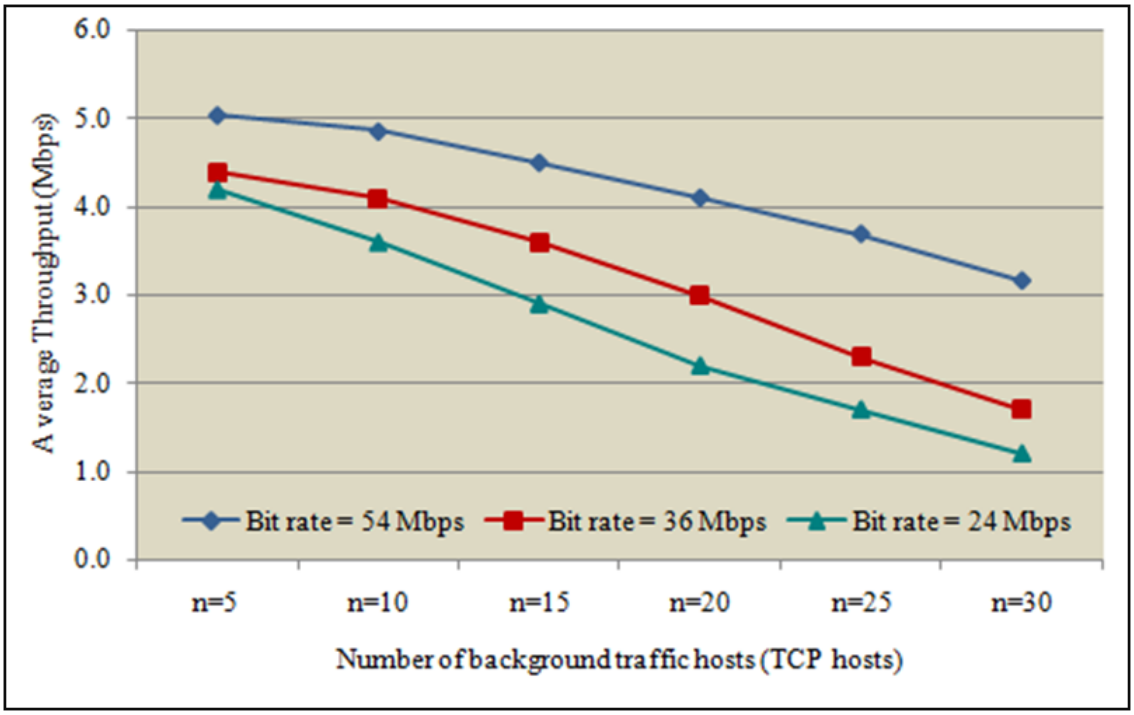

Figure 4.1: Average end-to-end throughput of the MC when the background traffic intensity increased in WLANs under various channel bit rates.

In Figure 4.1, it can be seen that, when the number of TCP hosts is increased from 5 up to 30 hosts in the network, the MC with a channel bit rate of 54 Mbps has a higher average end-to-end throughput than the MC compared with a channel bit rate of 36 Mbps and/or 24Mbps. Consequently, this causes more FTP traffic and then network collision occurs. In addition, when we use a channel bit rate less than 54Mbps in our network such as 36Mbps or 24 Mbps the collisions in the shared wireless channel of access point increases. Therefore, the MC will use the congestion avoidance mechanism "increase/multiplicative-decrease (AIMD) algorithm" to reduce the transmission rate (window size) and to reduce the congestion window (cwnd) by half after each loss. Further, we found that, in case of 36 Mbps and 24 Mbps, the use of AIMD algorithm increased, and transmission rate and cwnd decreased frequently. As a result, MC became unable to successfully transmit more data chunks as when we use a channel bit rate of 54 Mbps.

In Figure 4.1, when there are 5 TCP hosts in the network the average throughput under different channel bit rates were as follows: with a bit rate of 54 Mbps the throughput of MC was 5.067 Mbps, with a bit rate of 36 Mbps the throughput was 4.421 Mbps, and with a bit rate of 24 Mbps the throughput was 4.237 Mbps. Moreover, when we increased the number of TCP hosts to 30 hosts, the average throughput decreased based on the bit rates 54 Mbps, 36 Mbps, and 24 Mbps in order as follows: 3.168 Mbps, 1.758 Mbps, and 1.228 Mbps.

In conclusion, when the channel bit rate is increased, the average throughput of MC also increases; and more data packets were successfully delivered with less requirements for retransmission. The mean and standard deviation (Stdev) value of the average throughput of the MC under various channel bit rates was described in Tables 4.1, 4.2, and 4.3 respectively.

Table 4.1: The mean and standard deviation value of the average end-to-end throughput of the MC with a channel bit rate of 54 Mbps.

No. of hosts	**Mean**	**Stdev**
5 hosts	5.067 Mbps	93 Kbps
10 hosts	4.868 Mbps	86 Kbps
15 hosts	4.518 Mbps	78 Kbps
20 hosts	4.116 Mbps	73 Kbps
25 hosts	3.705 Mbps	53 Kbps
30 hosts	3.168 Mbps	35 Kbps

Table 4.2: The mean and standard deviation value of the average end-to-end throughput of the MC with a channel bit rate of 36 Mbps

No. of hosts	**Mean**	**Stdev**
5 hosts	4.421 Mbps	88 Kbps
10 hosts	4.135Mbps	72 Kbps
15 hosts	3.627 Mbps	63 Kbps
20 hosts	3.045 Mbps	58 Kbps
25 hosts	2.337 Mbps	33 Kbps
30 hosts	1.758 Mbps	19 Kbps

Table 4.3: The mean and standard deviation value of the average end-to-end throughput of the MC with a channel bit rate 24 Mbps

No. of hosts	**Mean**	**Stdev**
5 hosts	4.237 Mbps	75 Kbps
10 hosts	3.613 Mbps	68 Kbps
15 hosts	2.914 Mbps	62 Kbps
20 hosts	2.266 Mbps	54 Kbps
25 hosts	1.370 Mbps	36 Kbps
30 hosts	1.228 Mbps	21 Kbps

Increasing the number of TCP hosts will decrease the average throughput of all the TCP background traffic hosts in the network. In Figure 4.2, It is clear that the average throughput of each TCP background traffic host is decreased under different channel bit rates when the background traffic hosts increased from 5 to 30 (more FTP traffic generated by TCP hosts).

For instance, when there are 5 TCP hosts in the network, the average throughput for each TCP host under different channel bit rates it were as follows: with a bit rate of 54 Mbps = 4.584 Mbps, with a bit rate of 36 Mbps = 4.015 Mbps, and with a bit rate of 24 Mbps = 3.611Mbps. While when the number of TCP hosts increased to 30 hosts the throughput was as following: with a bit rate of 54 Mbps = 1.725 Mbps, with a bit rate of 36 Mbps = 1.013 Mbps, and a bit rate of 24 Mbps = 0.671. Tables 4.4, 4.5, and 4.6 describe the mean and standard deviation value of the average end-to-end throughput for a single TCP background traffic host under different channel bit rates.

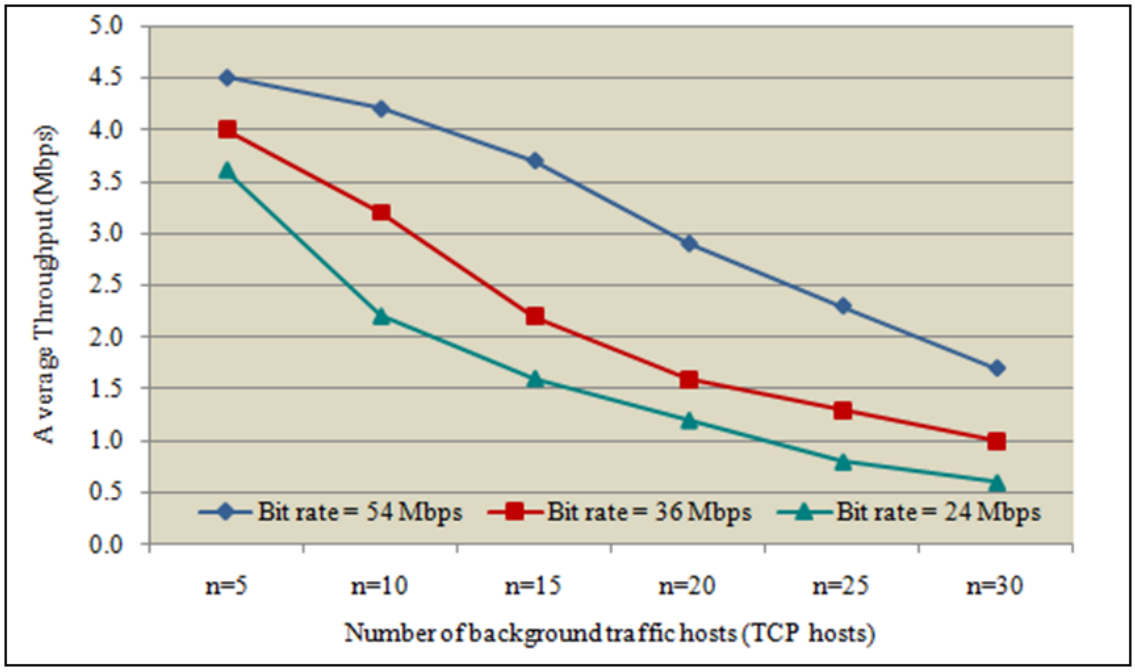

Figure 4.2: Average end-to-end throughput of a single TCP host when the background traffic intensity increased in WLANs under various channel bit rates.

Table 4.4: The mean and standard deviation value of the average end-to-end throughput of a single TCP host with a channel bit rate of 54 Mbps

No. of hosts	**Mean**	**Stdev**
5 hosts	4.584 Mbps	97 Kbps
10 hosts	4.213 Mbps	90 Kbps
15 hosts	3.734 Mbps	83 Kbps
20 hosts	2.913 Mbps	72 Kbps
25 hosts	2.312 Mbps	45 Kbps
30 hosts	1.725 Mbps	23 Kbps

Table 4.5: The mean and standard deviation value of the average end-to-end throughput of a single TCP host with a channel bit rate of 36 Mbps

No. of hosts	Mean	Stdev
5 hosts	4.015 Mbps	83 Kbps
10 hosts	3.245 Mbps	68 Kbps
15 hosts	2.273 Mbps	63 Kbps
20 hosts	1.642 Mbps	47 Kbps
25 hosts	1.355 Mbps	27 Kbps
30 hosts	1.013 Mbps	17 Kbps

Table 4.6: The mean and standard deviation value of the average end-to-end throughput of a single TCP host with a channel bit rate of 24 Mbps

No. of hosts	Mean	Stdev
5 hosts	3.611 Mbps	101 Kbps
10 hosts	2.255 Mbps	94 Kbps
15 hosts	1.602 Mbps	77 Kbps
20 hosts	1.274 Mbps	61 Kbps
25 hosts	0.835 Mbps	37 Kbps
30 hosts	0.671 Mbps	22 Kbps

Comparisons between the average throughput of the MC and a single TCP traffic host

In this section, we will describe a small comparison between the average end-to-end throughput of the MC and a single TCP hosts. In Figures 4.3, 4.4, and 4.5, it can be seen that, the MC achieves a higher throughput than the TCP throughput of background traffic under various channel bit rates due to the multi-homing and multi-streaming features.According to the multi-streaming feature of the MC, this feature avoids the HOL blocking in the MC buffer. In addition, in case of the multi-homing feature of the MC, when the MC moves to WLAN2, it will use a secondary path with an independent transmission rate and an initial cwnd value from the primary path.

Therefore, this helps the MC to have a new cwnd value equal to 2*MTU and a large window size; since it still decreased by half each time congestion was detected when using primary path and TCP hosts. In this situation, the MC will have a chance to transmit more data which is the reason to have higher throughput than the TCP hosts. Moreover, the packet structure and unordered data delivery

may have a chance to be other reasons for the MC to have higher throughput than the TCP hosts, which includes more data than the TCP packet. Consequently, all previous reasons prove that the MC has more robustness to achieve a better throughput than the TCP host.

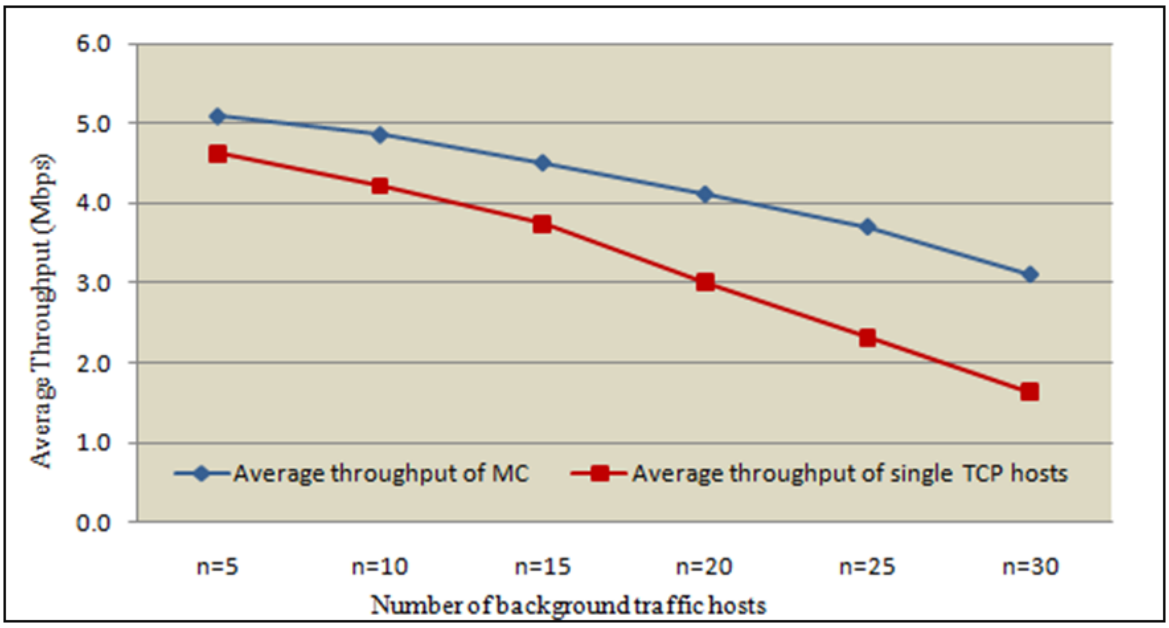

Figure 4.3: Average end-to-end throughput of the MC and a single TCP host when the background traffic intensity increased in WLANs with a channel bit rate of 54 Mbps.

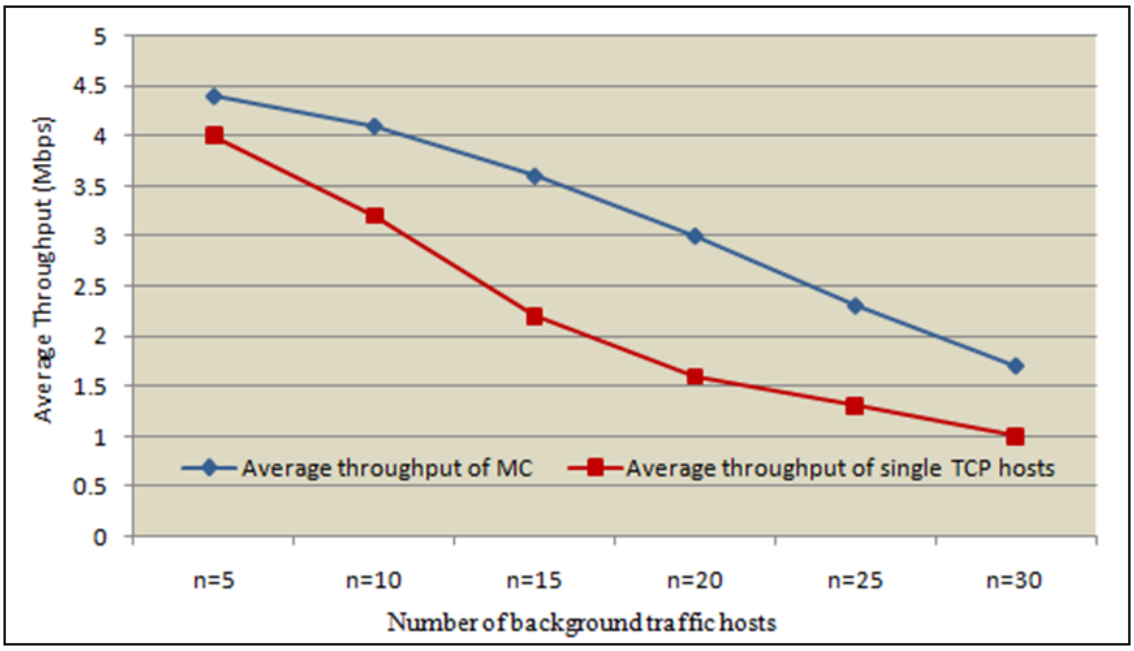

Figure 4.4: Average end-to-end throughput of the MC and a single TCP host when the background traffic intensity increased in WLANs with a channel bit rate of 36 Mbps.

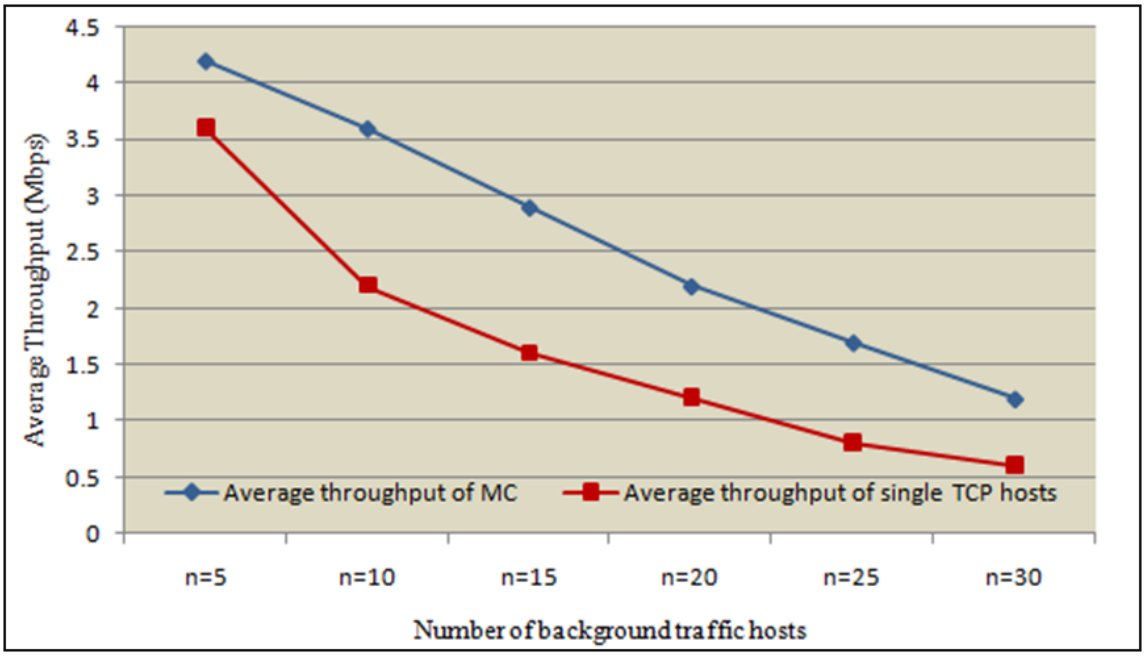

Figure 4.5: Average end-to-end throughput of the MC and a single TCP host when the background traffic intensity increased in WLANs with a channel bit rate of 24 Mbps.

Handover Delay and Average End-to-end Delay

In this section we measured the impact of increasing background traffic intensity (number of stationary TCP hosts) on end-to-end delay of the MC. We also evaluate the average end-to-end delay and handover delay for the MC which moves among congested networks under different channel bit rates. Handover delay is one of the most important parameters in every multi-homing method. In our experiment, handover delay is the time that is taken by the MC when deciding to use a second path and the actual time the MC starts to use it to start network communication with Dst. In Figure 4.6, the X-axis shows intensity of background traffic whereas the Y-axis shows the handover delay time.

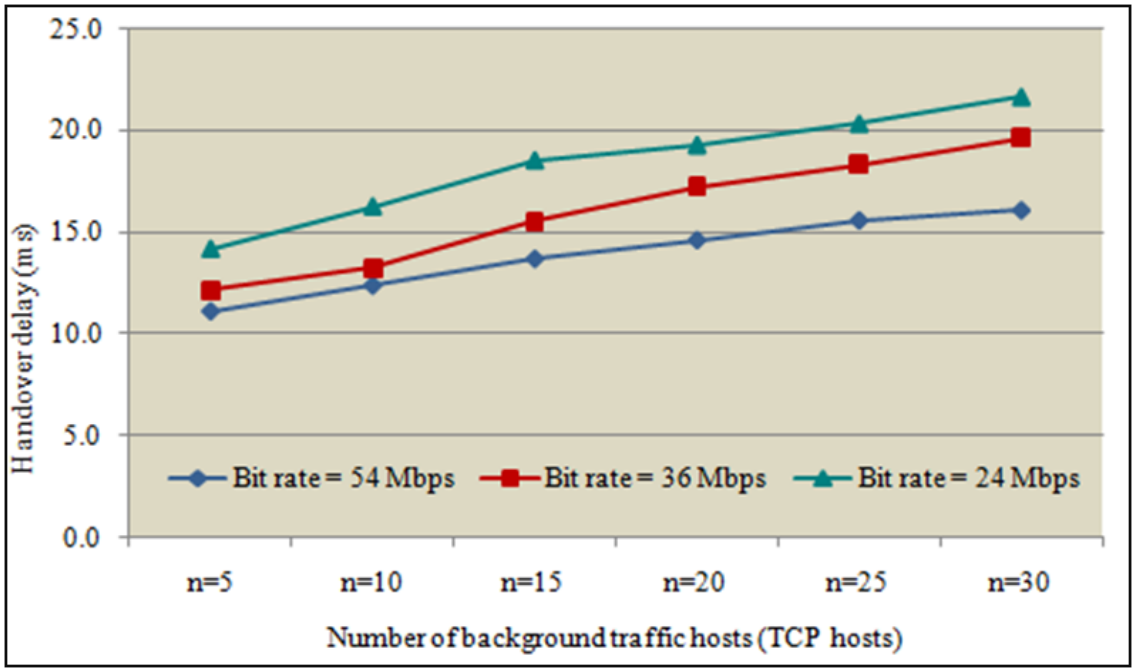

Figure 4.6: Handover delay of the MC when the background traffic intensity increased in WLANs under various channel bit rates.

As can be seen in Figure4.6, the handover delay can be affected by channel congestion, which mean that the handover delay between MC and Dst linearly increased when we increase the background traffic hosts under different channel bit rates. In SCTP multi-homing technology, the actual time that is required for SCTP host to monitor the accessibility of the second path is called HEARTBEAT delay

time (the delay of sending HEARTBEAT chunks and replying with HEARTBEAT ACK chunks). The HEARTBEAT delay time of the MC in our experiment set to10ms.

As shown in Figure 4.6, In case of 54 Mbps, when we increased the number of stationary TCP hosts in the networks, the HEARTBEAT delay time increases. This increased the handover delay. Moreover, based on the HEARTBEAT delay in our simulation (10ms), when there are 5 TCP hosts in the network, the MC takes approximately 11.08 ms to switch to the ap2 while this time increased each time we increased the number of TCP hosts to reach 16.10 ms when there are 30 hosts. This is because of the heavy traffic (more FTP traffic) on the wireless channel and the links between hosts and Dst in the network. Moreover, in case of a bit rate of 36 Mbps, when there are 5 TCP hosts in the network, the handover delay was 12.16 ms while this time increasesed to 19.65 ms when there are 30 TCP hosts. However, in case of 24 Mbps bit rate the handover delay increased from 14.26 ms to 21.61 ms when we increased the number of TCP hosts from 5 to 30 hosts. Furthermore, by increasing the number of TCP hosts and using a small channel bit rates, the wireless medium will be more congested. This causes the RTO (Doubling RTO value) and RTT value to increase for the MC. Thus, this will effect the end-to-end delay of the MC.

As we can see in Figure 4.7. as the number of TCP hosts increased in our network from 5 to 30, the end-to-end delay of the MC also increased under different channel bit rates. In Figure 4.7, it is clear that, for a small bit rate of 24 Mbps, as the number of TCP hosts increasesed, the handover delay and end-to-end delay increased sharply. While for larger bit rates of 36 and 54 Mbps, the handover delay and end-to-end delay also increased. However, this increment is lower than the handover delay and end-to-end delay of a bit rate of 24 Mbps.

Average end-to-end delay is composed of five main important values such as: propagation delay, transmission delay, queuing delay, processing delay, and handover delay (In case of mobile node).

This is can be explained mathematically as the following:

$$D_{e2e} = D_{ptop} + D_{ptop} + D_{trans} + D_{queu} + D_{proce} + D_{HO} \tag{4.2}$$

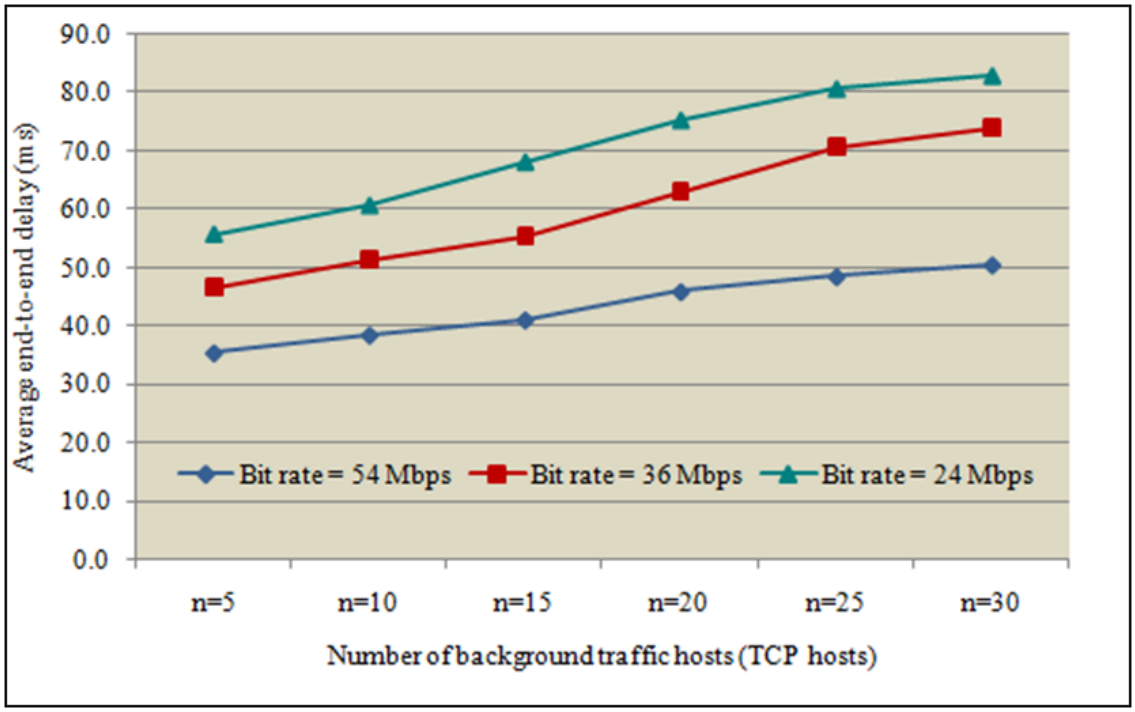

Figure 4.7: Average end-to-end delay of the MC when the intensity of background traffic increased in WLANs under various channel bit rates.

In our simulation, the increased of the TCP hosts lead the wireless channel to be more congested. As a result, this leads the D_{prop}, D_{trans}, and D_{HO} to be increaseed. Another result was that the D_{queu} and D_{proce} increased due to congestion in the R1 and R2. As we can see in Figure 4.7, in case of a 54 Mbps bit rate when there are 5 TCP hosts in the network, the D_{e2e} of the MC was 35.49 ms.This delay increased when the number of the TCP host reach 30 hosts to be D_{e2e}= 50.80 ms . In case of using 36 Mbps when there are 5 hosts in the network, the D_{e2e}= 46.61 ms and when the number of TCP hosts increased to 30 hosts the D_{e2e}= 74.11 ms. However, in case of using 24 Mbps, the $D_{e2e}$was 55.64 ms when there are 5 TCP hosts in the network and this D_{e2e} increased to be 83.30 ms when there are 30 TCP hosts. Tables 4.7, 4.8 and 4.9 present the mean and standard deviation (stdev) value of the handover delay and the end-to-end delay of the MC while increasing the number of TCP hosts in the network under various channel bit rates in the network.

Table 4.7: The mean and standard deviation value of handover delay and average end-to-end delay of the MC with a channel bit rate 54 Mbps

No. of hosts	**Mean D_{e2e}**	**Stdev D_{e2e}**	**Mean D_{HO}**	**Stdev D_{HO}**
5 hosts	35.49 ms	0.334 ms	11.08 ms	0.154 ms
10 hosts	38.55 ms	0.577 ms	12.36 ms	0.236 ms
15 hosts	41.14 ms	0.652 ms	13.68 ms	0.238 ms
20 hosts	46.01 ms	0.717 ms	14.62 ms	0.310 ms
25 hosts	48.58 ms	0.732 ms	15.60 ms	0.341 ms
30 hosts	50.80 ms	0.983 ms	16.10 ms	0.387 ms

Table 4.8: The mean and standard deviation value of handover and end-to-end delay of MC with channel bit rate 36 Mbps

No. of hosts	**Mean D_{e2e}**	**Stdev D_{e2e}**	**Mean D_{HO}**	**Stdev D_{HO}**
5 hosts	46.61 ms	0.474 ms	12.16 ms	0.094 ms
10 hosts	51.51 ms	0.512 ms	13.26 ms	0.121 ms
15 hosts	55.42 ms	0.586 ms	15.53 ms	0.168 ms
20 hosts	63.27 ms	0.624 ms	17.28 ms	0.215 ms
25 hosts	70.65 ms	0.719 ms	18.33 ms	0.336 ms
30 hosts	74.11 ms	0.836 ms	19.65 ms	0.366 ms

Table 4.9: The mean and standard deviation value of handover and end-to-end delay of MC with channel bit rate 24 Mbps

No. of hosts	**Mean D_{e2e}**	**Stdev D_{e2e}**	**Mean D_{HO}**	**Stdev D_{HO}**
5 hosts	55.64 ms	0.534 ms	14.26 ms	0.0912 ms
10 hosts	60.61 ms	0.612 ms	16.36 ms	0.0978 ms
15 hosts	68.12 ms	0.586 ms	18.55 ms	0.144 ms
20 hosts	75.34 ms	0.724 ms	19.28 ms	0.183 ms
25 hosts	80.77 ms	0.838 ms	20.23 ms	0.251 ms
30 hosts	83.30 ms	0.852 ms	21.61 ms	0.286 ms

Packet loss

In this section we evaluate the probability of packet loss for the MC when we increased the number of TCP background traffic hosts under various channel bit rates of 802.11g: 54 Mbps, 36Mbps, and 24Mbps in our simulation. In Figure 4.8, the X-axis shows intensity of background traffic whereas the Y-axis shows the probability of packet loss.

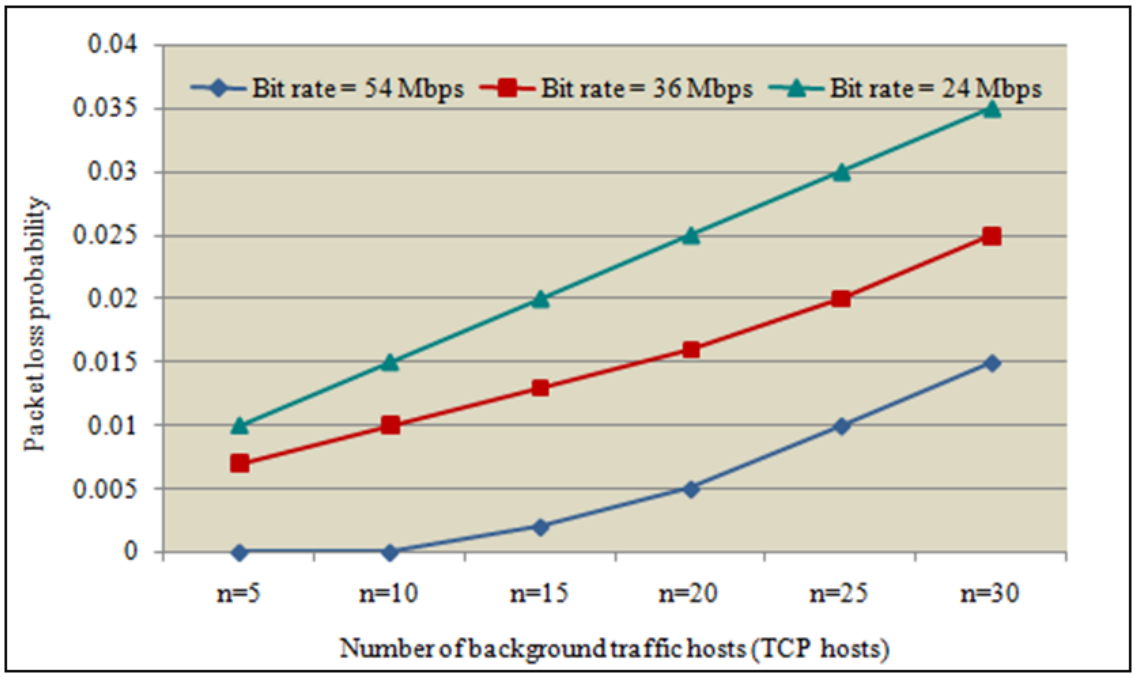

Figure 4.8: the packet loss probability of the MC hosts when the background traffic intensity increased in WLANs under various channel bit rates.

As we can see in Figure 4.8, when we increased the intensity of background traffic by increasing the number of stationary TCP hosts, the network will be more congested as. In case of using 54 Mbps there was a packet loss rate, but this rate was less compared to the packet loss rate in the case of 36 Mbps or 24Mbps. This is because that the background traffic will be exhausted by the large size of the queue buffer space (buffer will overflow). Therefore, this caused more probability of collision and more packet loss rate (MC packets are dropped in the queue buffer).

For example, in case of a bit rate of 54 Mbps, when there are 5 TCP hosts competed to access the channel and sent its data, the probability of packet loss was 0 while when we increased the number of TCP hosts inour semulation to10, 15, 20, 25, up to 30, the probability of loss was slightly increased as following 0, 0.002, 0.005, 0.01, and 0.015. In case of a bit rate of 36 Mbps, when there are 5 TCP hosts in the network, the probability of packet loss was 0.007 Mbps while when we increased the number of TCP hosts to10, 15, 20, 25, up to 30, the probability of loss was increased as following 0.01, 0.013,

0.016, 0.02, and 0.025 respectively. However, in case of a bit rate of 24 Mbps, the packet loss probability was to 0.01, 0.015, 0.02, 0.025, 0.03, and 0.035 when we increased the number of TCP hosts from 5 to 30 hosts. Table 4.10 describe the probability of packet loss under different channel bit rates when the background traffic increased.

Table 4.10: Packet loss probability under various channel bit rates while the number of TCP hosts increased

No. of hosts	**54 Mbps**	**36 Mbps**	**24 Mbps**
5 hosts	0	0.007	0.01
10 hosts	0	0.01	0.015
15 hosts	0.002	0.013	0.02
20 hosts	0.005	0.016	0.025
25 hosts	0.01	0.02	0.03
30 hosts	0.015	0.025	0.035

4.3 Scenario B: Results and Performance Analysis

Average End-to-end Throughput

In this section, we analyze the wireless throughput of MC and background traffic hosts (number of SCTP single homed hosts) for measuring the performance of MC between congested WLANs. In addition, we also estimate how MC will be affected with the intensity of background traffic in the network via using different mobility speed as following: Random Walking Speed (4.5 km/h), Brisk Waking Speed (6.5 km/h), and Random Cycling Speed (15.5 km/h). This is achieved by increasing the number of SCTP hosts. As illustrated in Figure 4.9, the X-axis shows number of background traffic hosts whereas the Y-axis shows the average end-to-end throughput of MC.

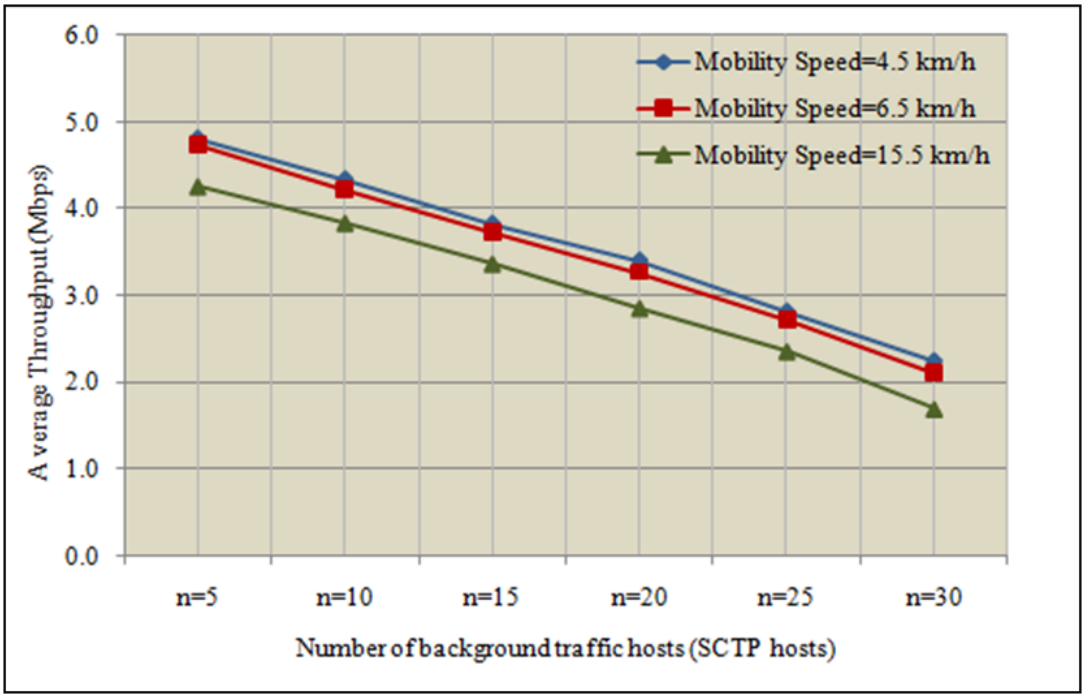

Figure 4.9: Average end-to-end throughput of the MC when the background traffic intensity increased in WLANs under various mobility speed.

In Figure 4.9, it is clear that, since we increased number of SCTP hosts from 5 up to 30 hosts in the network, the MC move at a speed of Random Walking Speed (4.5 km/h) has higher average end-to-end throughput than MC moves at speed of Brisk Waking Speed (6.5 km/h) and/or Random Cycling Speed (15.5 km/h). According to Figure 4.9, when there are 5 SCTP hosts in the network, the average end-to-end throughput of MC was as following: at Random Walking Speed (4.5 km/h) = 4.812 Mbps, at Waking Speed (6.5 km/h) = 4.741 Mbps, and Random Cycling Speed (15.5 km/h) = 4.366 Mbps. Moreover when we increased number of SCTP hosts to be 30 hosts the average end-to-end throughput of the MC dramatically decrease as follows: at Random Walk Speed (4.5 km/h) = 2.256 Mbps, at Brisk Waking Speed (6.5 km/h) = 2.112 Mbps, and at Random Cycling Speed (15.5 km/h) = 1.653 Mbps.

The graph denotes that, by increasing the mobility speeds, the intermediate objects could degrade the strength and quality of receiving signal in wireless communication and the interruption of the links between MC and Dst in the networks increased. This have a negative impact on throughput of wireless medium. Thus, cause to increase the probability of packet loss (as shown in Figure 4.16). Consequently,

the average end-to-end throughput of MC will be decreased. Moreover, fading effects of wireless mobile nodes have to be taken in account. Also a transmitter can experience a different multipath environment when it sends a packet because of fast fading effect. Therefore the average of throughput will dramatically decrease in addition to average end to end delay (see Figure 4.15) and probability of packet los (see Figure 4.16). Tables 4.11, 4.12, and 4.13 summarize the mean and standard deviation(stdev) value of the average end-to-end throughput of the MC under various mobility speeds for all cases of increasing number of SCTP hosts in the networks.

Table 4.11: The mean and standard deviation value of the average end-to-end throughput of the MC with Random Walking Speed (4.5 km/h).

No. of hosts	**Mean**	**Stdev**
5 hosts	4.812 Mbps	113 Kbps
10 hosts	4.343 Mbps	88 Kbps
15 hosts	3.833 Mbps	76 Kbps
20 hosts	3.411 Mbps	58 Kbps
25 hosts	2.824 Mbps	43 Kbps
30 hosts	2.256 Mbps	20 Kbps

Table 4.12: mean and standard deviation value of the average end-to-end throughput of the MC with Brisk Walking Speed (6.5 km/h).

No. of hosts	**Mean**	**Stdev**
5 hosts	4.741 Mbps	98 Kbps
10 hosts	4.223 Mbps	76 Kbps
15 hosts	3.732 Mbps	51 Kbps
20 hosts	3.271 Mbps	35 Kbps
25 hosts	2.720 Mbps	21 Kbps
30 hosts	2.211 Mbps	13 Kbps

Table 4.13: The mean and standard deviation value of the average end-to-end throughput of the MC with Random Cycling Speed (15.5 km/h).

No. of hosts	Mean	Stdev
5 hosts	4.366 Mbps	87 Kbps
10 hosts	3.845 Mbps	73 Kbps
15 hosts	3.375 Mbps	51 Kbps
20 hosts	2.863 Mbps	43 Kbps
25 hosts	2.365 Mbps	37 Kbps
30 hosts	1653 Mbps	21 Kbps

It is more likely, increasing number of SCTP single homed hosts will affect negatively to decrease not only the MC average throughput, but also all other SCTP traffic hosts in the network. In Figure 4.10, it can be seen that, as the number of SCTP hosts increasesd the throughput of single SCTP hosts decrease. However, this decreased is slightly high in case of using Random Cycling Speed (15.5 km/h) compared to Brisk Waking Speed (6.5 km/h) and Random Walking Speed (4.5 km/h).

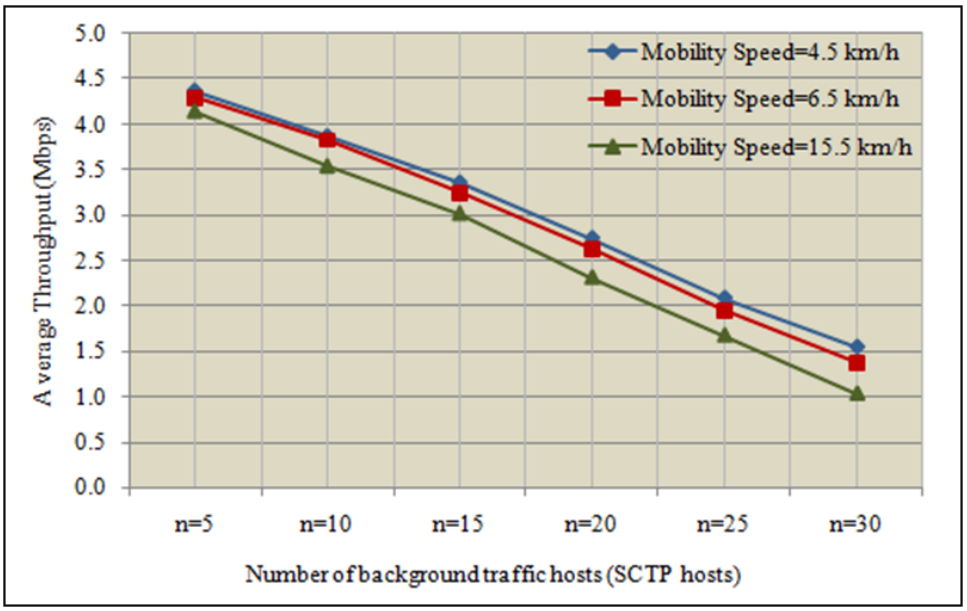

Figure 4.10: The Average throughput of a single SCTP traffic host when the background traffic intensity is increased in WLANs with different mobility speeds

Tables 4.14, 4.15, and 4.16 summarize the mean and standard deviation value of the average throughput of a single SCTP hosts under various mobility speeds in the networks.

Table 4.14: The mean and standard deviation value of the average end-to-end throughput of a single SCTP host with Random Walking Speed (4.5 km/h).

No. of hosts	Mean	Stdev
5 hosts	4.360 Mbps	117 Kbps
10 hosts	3.865 Mbps	84 Kbps
15 hosts	3.355 Mbps	60 Kbps
20 hosts	2.745 Mbps	53 Kbps
25 hosts	2.157 Mbps	38 Kbps
30 hosts	1.650 Mbps	27 Kbps

Table 4.15: The mean and standard deviation value of the average end-to-end throughput of a single SCTP host with Brisk Walking Speed (6.5 km/h).

No. of hosts	Mean	Stdev
5 hosts	4.301 Mbps	87 Kbps
10 hosts	3.834 Mbps	81 Kbps
15 hosts	3.248 Mbps	66 Kbps
20 hosts	2.628 Mbps	59 Kbps
25 hosts	2.054 Mbps	45 Kbps
30 hosts	1.475 Mbps	31 Kbps

Table 4.16: The mean and standard deviation value of the average end-to-end throughput of a single SCTP host with Random Cycling Speed (15.5 km/h).

No. of hosts	Mean	Stdev
5 hosts	4.154 Mbps	114 Kbps
10 hosts	3.543 Mbps	95 Kbps
15 hosts	3.023 Mbps	72 Kbps
20 hosts	2.321 Mbps	56 Kbps
25 hosts	1.786 Mbps	53 Kbps
30 hosts	1.065 Mbps	41 Kbps

Comparisons between the average throughput of the MC and a single SCTP traffic host

As we can see in Figures 4.11, 4.12, and 4.13, the MC with different mobility speeds achieves higher throughput than SCTP traffic host due to the multi-homing feature of MC. Therefore, this is helping the MC to have again bigger value of cwnd equal to 2*MTU and large window size; since it was still decreased by half each time congestion detect in case of using primary path in SCTP single-homed hosts. Therefore, the MC will have a chance to transmit more data which is reason to have higher throughput than SCTP single-homed hosts. This it will be the main reasons prove that SCTP multi-homed host/MC is more robust to achieve a better throughput than SCTP single-homed host .

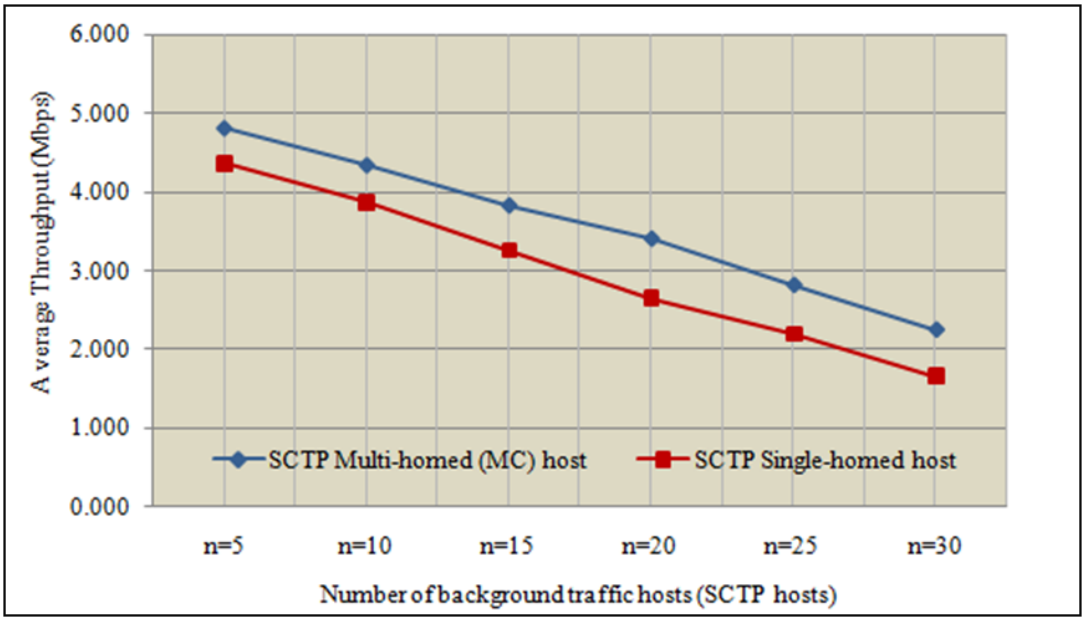

Figure 4.11: The average throughput of both MC (with Random Walk Speed (4.5 km/h)) and SCTP single homed hosts.

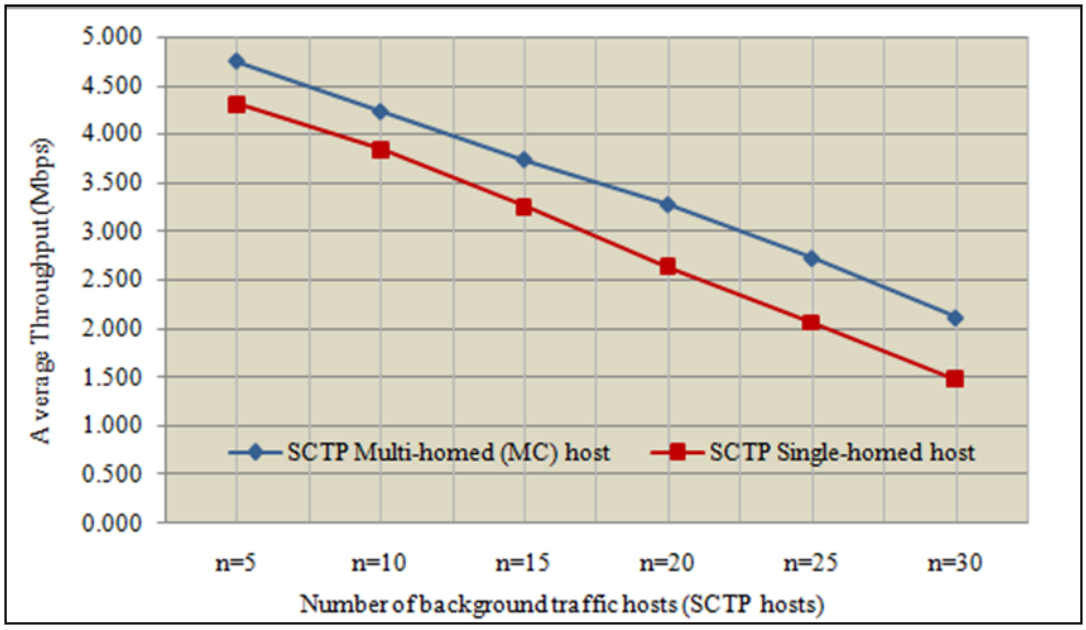

Figure 4.12: The average throughput of both MC (with Brisk Walk Speed (6.5 km/h)) and SCTP single homed hosts.

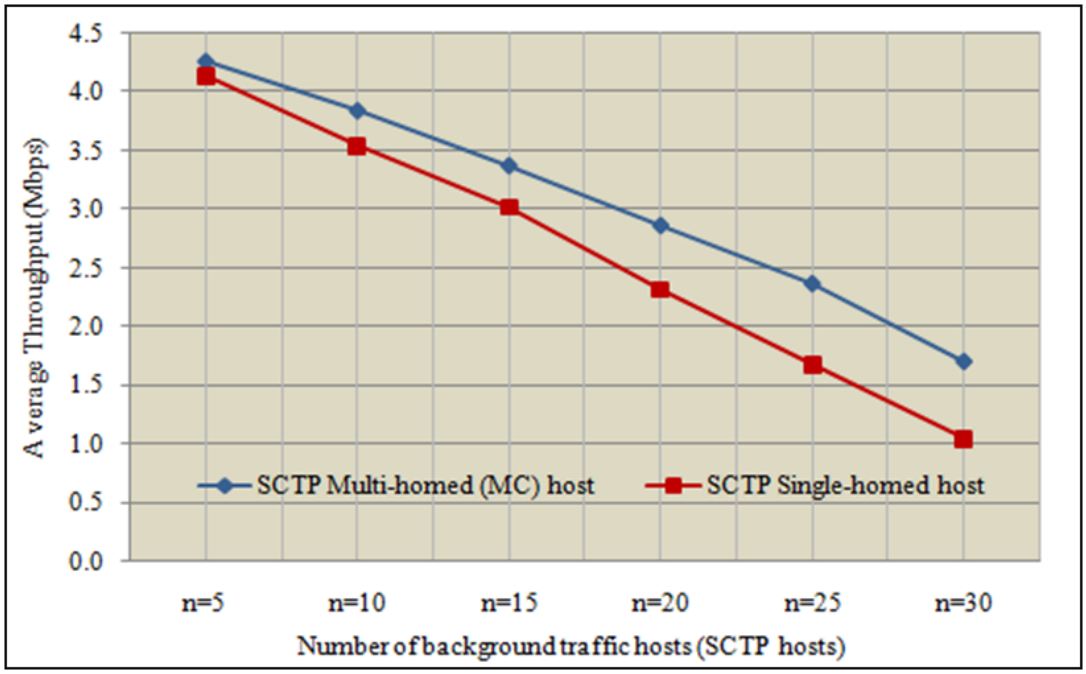

Figure 4.13: The average throughput of both MC (with Random Cycling Speed (15.5 km/h)) and SCTP single homed hosts.

Handover delay and Average end-to-end Delay

In this section we measure the impact of increasing background traffic intensity (number of SCTP single homed hosts) on the end-to-end delay of MC. We also evaluate the average end-to-end delay and handover delay for MC moves among congested networks with different mobility speeds. In Figure 4.14, the X-axis shows the intensity of background traffic whereas the Y-axis shows the handover delay time (ms).

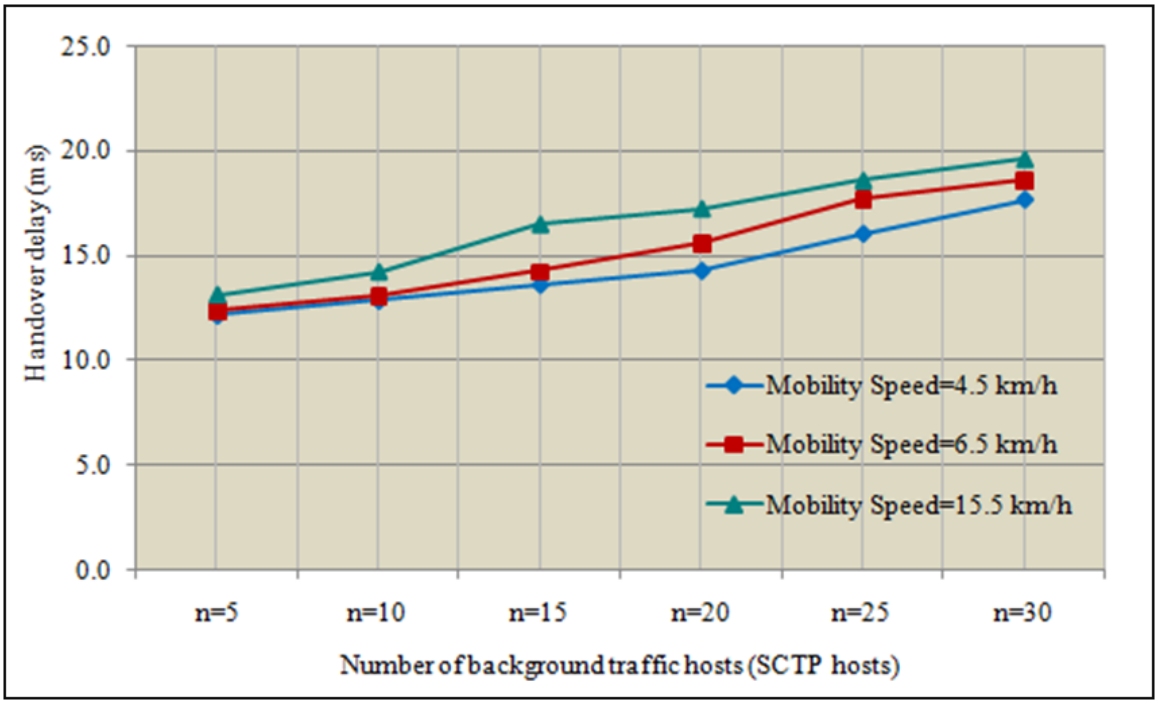

Figure 4.14: Handover delay of MC when the background traffic intensity increased in WLANs under various mobility speeds.

As we can see in Figure 4.14, as the SCTP background traffic hosts increases in the network, the handover delay of MC also increases under various mobility speeds. This means that, the handover delay has been affected by increasing the mobility speeds of the MC. Thus, the HEARTBEAT delay time will increase. This led to increase the handover delay based on the HEARTBEAT delay in our simulation which is set to10ms.

In Figure 4.14. In case of Random Walking Speed (4.5 km/h), when there are 5 hosts in the network the MC takes approximately 12.16 ms to handoff to the ap2. This time will increase each time we increases the number of SCTP hosts to reach to 17.65 ms when there are 30 hosts. This is because the heavy traffic (more FTP traffic) on the wireless channel and the links between hosts and Dst in the network. In case of Brisk Waking Speed (6.5 km/h), when there are 5 SCTP hosts in the network, the handover delay is equal to 12.36 ms while this time increases to 18.65 ms when there are 30 hosts. However, in case of Random Cycling Speed (15.5 km/h), the handover delay increase from 13.16ms to 19.65 ms when we increase the number of SCTP hosts from 5 to 30 hosts.

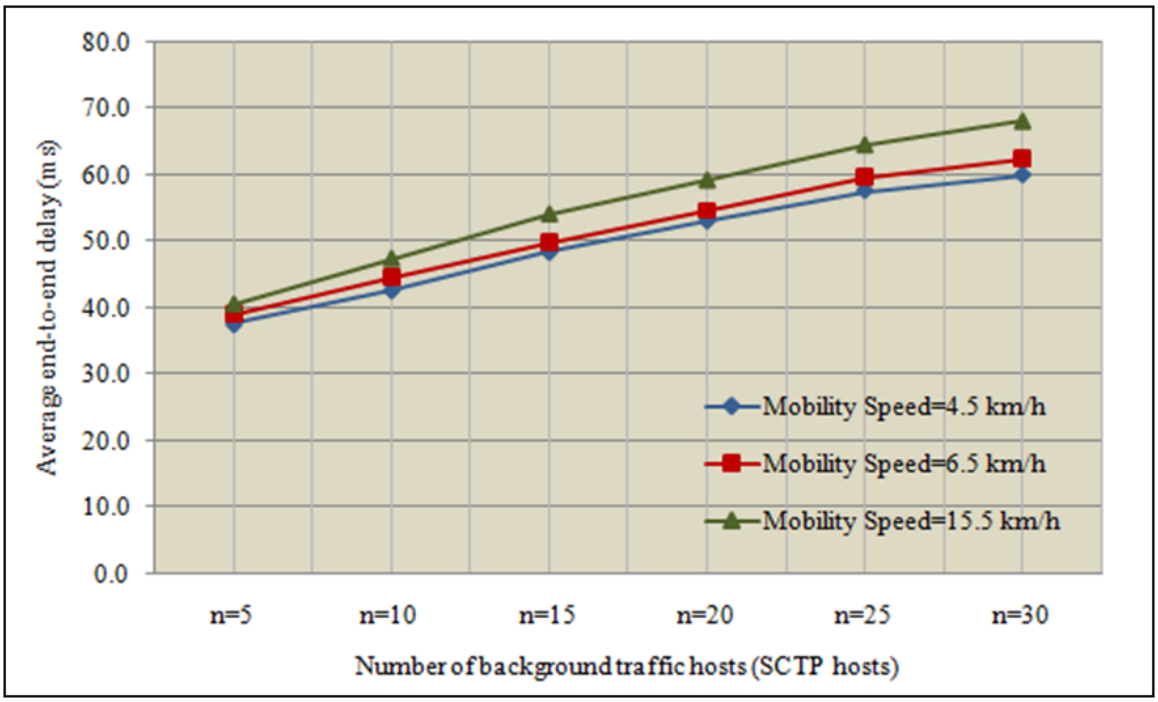

Figure 4.15: Average end-to-end delay of MC when the background traffic intensity is increased in WLANs under various mobility speeds.

It is observed from the Figure 4.15 that as the number of SCTP background traffic hosts increases average end-to-end delays for MC also increases (under various mobility speeds). This due to the fact that, increasing the background traffic intensity leads the wireless medium/channel to be more congested. In addition, due to that, the average end-to-end delay is including many important factors

which effect on the end-to-end delay of MC as mentioned in equation (4.2). Obviously, this leads to increase the D_{prop}, D_{trans}, and D_{HO} (as in Figure 4.14). Also it will increase the D_{queu} and D_{proce} due to congestion in the R1 and R2.

In Figure 4.15, it can be seen, since we increase number of SCTP hosts in the network, the MC with Random Cycling Speed (15.5 km/h) has slightly higher average end-to-end delay compared to MC with Brisk Waking Speed (6.5 km/h) and/or Random Walking Speed (4.5 km/h). In the case of the MC moved at mobility speed of 15.5 km/h, link failure has occurred frequently between MC and Dst. Therefore the packets will take long time to deliver to Dst. Consequently, this will increase the handover and average end to end delay of the MC.

In case of MC moves with Random Walking Speed (4.5 km/h) we can see that, when the number of SCTP background traffic hosts increase in the network from 5 to 30 hosts, the average end-to-end delay of MC has approximately linear increase due to D_{HO} increase (as we can see in Figure 4.14) due to congestion. When there are 5 hosts in the network and the D_{HO}= 12.16 ms the D_{e2e}= 37.66 ms. This average end-to end delay will dramatically increases as the D_{HO} and other factors increases to be D_{e2e}= 60.01 ms when there are 30 hosts in the network and D_{HO}= 17.65ms. Moreover, in case of MC moves with Brisk Waking Speed (6.5 km/h), when there are 5 SCTP hosts and D_{HO}=12.36 ms the D_{e2e}= 39.42 ms while when there are 30 hosts and D_{HO} increases to 18.65 ms the D_{e2e}= 63.50 ms. However, in case of MC moves with Random Cycling Speed (15.5 km/h), the average end-to-end delay of MC when there are 5 SCTP hosts and D_{HO}= 13.16 ms the D_{e2e} = 40.66 ms this increases to be 68.15 ms when there are 30 SCTP hosts in the network and D_{HO}= 19.65 ms.

In conclusion, Figures 4.14 and 4.15 show that, as the number of SCTP background traffic hosts and mobility speeds are increased, the hand over delay and average end-to-end delay for MC will also be increased. Tables 4.17, 4.18, and 4.19 present the mean and standard deviation value of handover delay and end-to-end delay of MC while increasing the number of hosts in the network under different mobility speeds.

Table 4.17: The mean and standard deviation value of the handover delay and average end-to-end delay of the MC with Random Walking Speed (4.5 km/h).

No. of hosts	**Mean D_{e2e}**	**Stdev D_{e2e}**	**Mean D_{HO}**	**Stdev D_{HO}**
5 hosts	37.66 ms	0.483 ms	12.16 ms	0.245ms
10 hosts	42.71 ms	0.586 ms	12.68 ms	0.453 ms
15 hosts	48.42 ms	0.603 ms	13.62 ms	0.465 ms
20 hosts	53.11 ms	0.724 ms	14.28 ms	0.554 ms
25 hosts	57.68 ms	0.839 ms	16.03 ms	0.605 ms
30 hosts	60 ms	0.965 ms	17.65ms	0.635 ms

Table 4.18: The mean and standard deviation value of the handover delay and average end-to-end delay of the MC with with Brisk Walking Speed (6.5 km/h).

No. of hosts	**Mean D_{e2e}**	**Stdev D_{e2e}**	**Mean D_{HO}**	**Stdev D_{HO}**
5 hosts	39.42 ms	0.644 ms	12.36 ms	0.245 ms
10 hosts	44.65 ms	0.687 ms	13.06 ms	0.346 ms
15 hosts	49.80 ms	0.756 ms	14.25 ms	0.464 ms
20 hosts	54.63 ms	0.798 ms	15.63 ms	0.566 ms
25 hosts	59.73 ms	0.808 ms	17.73 ms	0.580 ms
30 hosts	63.50 ms	0.925 ms	18.65 ms	0.612 ms

Table 4.19: The mean and standard deviation value of the handover delay and average end-to-end delay of the MC with with Random Cycling Speed (15.5 km/h).

No. of hosts	**Mean D_{e2e}**	**Stdev D_{e2e}**	**Mean D_{HO}**	**Stdev D_{HO}**
5 hosts	40.66 ms	0.502 ms	13.16 ms	0.262 ms
10 hosts	47.51 ms	0.587 ms	14.26 ms	0.323 ms
15 hosts	54.22 ms	0.623 ms	16.53 ms	0.365 ms
20 hosts	59.31 ms	0.798 ms	17.28 ms	0.417 ms
25 hosts	64.62 ms	0.760 ms	18.63 ms	0.507 ms
30 hosts	68.15 ms	0.924 ms	19.65 ms	0.569 ms

Packet loss

In this section, we measure the impact of background traffic hosts on the packet loss probability for MC. We also evaluate the probability of packet loss for MC moves among congested networks with various mobility speeds. In Figure 4.16, the X-axis shows the intensity of background traffic whereas the Y-axis shows the probability of packet loss.

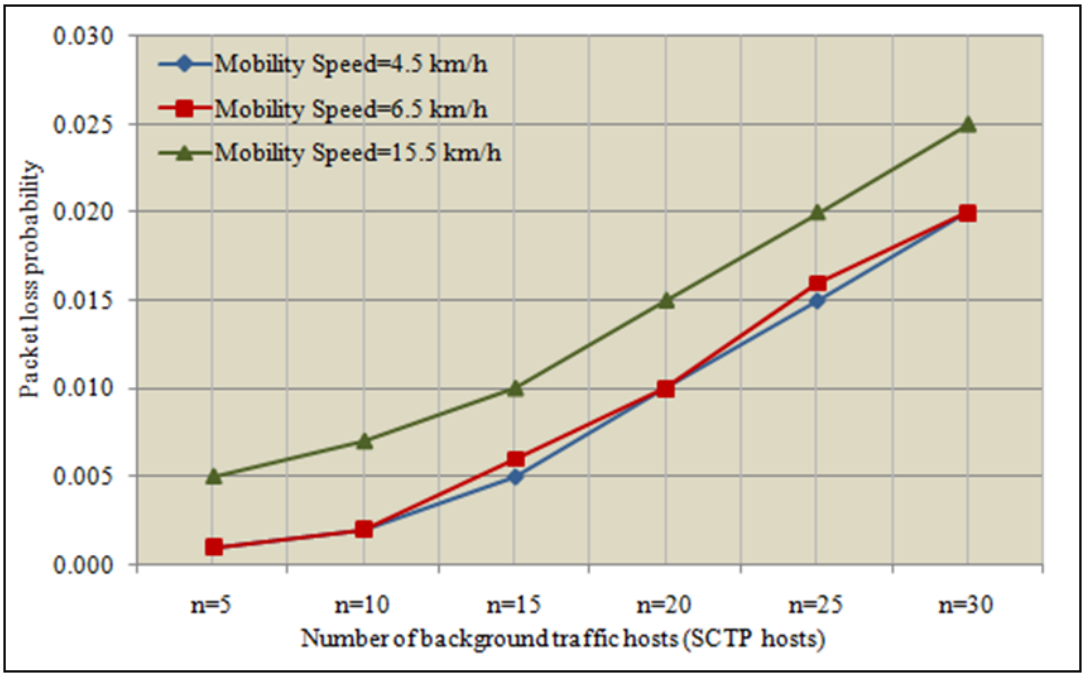

Figure 4.16: The packet loss probability of the MC hosts when the background traffic intensity increased in WLANs under various mobility speeds

It is observed from Figure 4.16 that as the number of SCTP background traffic hosts increases the probability of packet loss for MC is approximately exponentially increases. This due to that, the network will be more congested. Thus, it will increase the probability of collision and then the probability of packet loss rate. In Figure 4.16, it can be seen, since we increase the number of SCTP hosts in the network, the MC with Random Cycling Speed (15.5 km/h) has higher highest loss rate compared to the MC with Brisk Waking Speed (6.5 km/h) and/or Random Walking Speed (4.5 km/h). This is due to that the interruption of the links between MC and Dst in the networks. Therfore, this cause increase in the probability of packet loss.

In case the MC moves with Random Walking Speed (4.5 km/h), the probability of loss was in order as follows: 0.001, 0.002, 0.005, 0.01, 0.015, and 0.02 when the number of SCTP background traffic hosts increased from 5 to 30 hosts in the networks. Moreover, in case of Brisk Waking Speed (6.5 km/h), when there are 5 SCTP hosts in the networks competed to access the channel and sent its data, the probability of packet loss is almost the same compared with Random Walking Speed, it was 0.001 while when we increase the number of SCTP hosts to10, 15, 20, 25, up to 30, the probability of loss was slightly increased as following 0, 0.002, 0.006, 0.01, 0.016 and 0.02 respectively. While in case of MC moves with Random Cycling Speed (15.5 km/h), when there are 5 SCTP hosts in the network, the probability of packet loss was 0.005 while when we increase the number of TCP hosts to10, 15, 20, 25, up to 30, the probability of loss was to 0.007, 0.010, 0.015, 0.02, and 0.025 respectively when we increase the number of SCTP hosts from 5 to 30 hosts. Table 4.20 shows the packet loss probability of the MC when we increased the number of SCTP hosts at various mobility speeds of the MC.

Table 4.20: Packet loss probability of the MC at various mobility speeds while the intensity of background traffic increased

No. of hosts	**54 Mbps**	**36 Mbps**	**24 Mbps**
5 hosts	0.001	0.001	0.005
10 hosts	0.002	0.002	0.007
15 hosts	0.005	0.006	0.01
20 hosts	0.01	0.01	0.015
25 hosts	0.015	0.016	0.02
30 hosts	0.02	0.02	0.025

Chapter 5

Conclusions and Future work

5.1 Conclusions

To conclude our dissertation, two simulation scenarios were carried out in order to evaluate the impact of increasing the SCTP and TCP traffic on the performance of the MC, which moves among two congested WLANs. This evaluation is based on the performance metrics as follows: average end-to-end throughput, handover delay, average end-to-end delay, and packet loss rate. These simulation experiments were also performed to evaluate the performance of the MC under various network conditions which included different channel bit rates and different mobility speeds. Our results show that the performance of the MC was affected due to the intensity of the background traffic. Moreover, as the channel bit rate of SCTP multi-homed host increased the hosts will have better performance than lower bit rate while as the mobility speeds increases the SCTP hosts will have lower performance than using higher mobility speed.

5.2 Future Work

In this dissertation the implementation of SCTP multi homing technology was used to achieve more efficient communication performance between homogeneous networks (WLAN - WLAN) under various conditions networks. However, there are still some aspects that require further research in order to analyze the efficiency and the ability of this multi-homing feature. One of these aspects is multi-

streaming feature which will improve the communication performance between to hosts in the wireless network. This feature is not yet implemented in the OMNet ++ . In the future, it will be very interesting to perform some experiments that evaluate the performance of SCTP multi-homing and multi-streaming simultaneously to achieve higher throughput and reduce unnecessary Delay in the WLANs

Bibliography

[1] Rammohan Bandaru and Debashis Barman. Performance evaluation of sctp as a transport layer protocol. 2011.

[2] Vicuña Nelson, Jiménez Tania, and Hayel Yezekael. Performance of sctp in wi-fi and wimax networks with multi-homed mobiles. In *Proceedings of the 3rd International Conference on Performance Evaluation Methodologies and Tools*, page 71. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.

[3] Better networking with sctp, last accessed May 2014.

[4] Randall Stewart and Chris Metz. Sctp: new transport protocol for tcp/ip. *Internet Computing, IEEE*, 5(6):64–69, 2001.

[5] Shaojian Fu and Mohammed Atiquzzaman. Sctp: state of the art in research, products, and technical challenges. *Communications Magazine, IEEE*, 42(4):64–76, 2004.

[6] James F Kurose and Keith W Ross. *Computer Networking: A top-down approach featuring the Internet*, volume 2. Addison-Wesley Reading, 2001.

[7] Janani Chandrasekaran. Mobile ip: Issues, challenges and solutions. *Rutgers University*, 2009.

[8] Fayza Nada. Performance analysis of mobile ipv4 and mobile ipv6. *Int. Arab J. Inf. Technol.*, 4(2):153–160, 2007.

[9] Weiping He. *Integrated Mobility and Service Management for Future All-IP Based Wireless Networks*. PhD thesis, Virginia Polytechnic Institute and State University, 2009.

[10] T Daniel Wallace and Abdallah Shami. An analytic model for the stream control transmission protocol. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1–5. IEEE, 2010.

[11] Rob Brennan and Thomas Curran. Sctp congestion control: Initial simulation studies. In *Proc. 17th Intl Teletraffic Congress*. Citeseer, 2001.

[12] Andreas Jungmaier and Erwin P Rathgeb. On sctp multi-homing performance. *Telecommunication Systems*, 31(2-3):141–161, 2006.

[13] Wojciech Fraczek, Wojciech Mazurczyk, and Krzysztof Szczypiorski. Hiding information in a stream control transmission protocol. *Computer Communications*, 35(2):159–169, 2012.

[14] Randall Stewart. Stream control transmission protocol. 2007.

[15] Thomas Ravier, Rob Brennan, and Thomas Curran. Experimental studies of sctp multi-homing. *Teltec DCU, Dublin*, 9, 2001.

[16] Shin Maruyama, Michael Tuexen, Randall Stewart, Qiaobing Xie, and Masahiro Kozuka. Stream control transmission protocol (sctp) dynamic address reconfiguration. 2007.

[17] RR Stewart, Q Xie, M Tuexen, and I Rytina. Sctp dynamic addition of ip addresses. Technical report, Internet Draft, IETF, Nov. 2000. draft-stewart-addip-sctp-sigran-01. txt (work in progress), 2000.

[18] Randall Stewart, M Ramalho, Q Xie, M Tuexen, and P Conrad. Sctp partial reliability extension. 2002.

[19] Md Ibrahim Chowdhury and Mohammad Iqbal. A new solution approach for simultaneous mobility issues in seamless handover. In *Computational Intelligence, Modelling and Simulation (CIMSiM), 2012 Fourth International Conference on*, pages 341–346. IEEE, 2012.

[20] Raman Kumar Goyal and Sakshi Kaushal. A survey of msctp for transport layer mobility management. *Journal of Advances in Information Technology*, 4(1):20–27, 2013.

[21] Stream control transmission protocol, last accessed April 2014.

[22] Claudio Casetti. Fundamental concepts and mechanisms of stream control transmission protocol (sctp) multihoming. *Multihomed Communication with SCTP (Stream Control Transmission Protocol)*, 2012.

[23] Jinyang Shi, Yuehui Jin, Hui Huang, and Dajiang Zhang. Experimental performance studies of sctp in wireless access networks. In *Communication Technology Proceedings, 2003. ICCT 2003. International Conference on*, volume 1, pages 392–395. IEEE, 2003.

[24] Sherali Zeadally and Farhan Siddiqui. An empirical analysis of handoff performance for sip, mobile ip, and sctp protocols. *Wireless personal communications*, 43(2):589–603, 2007.

[25] Rajesh Rajamani, Sumit Kumar, and Nikhil Gupta. Sctp versus tcp: Comparing the performance of transport protocols for web traffic. *University of Wisconsin-Madison*, 2002.

[26] Syed Yasmeen Shahdad, Gulshan Amin, and Pushpender Sarao. Multihoming and multistream protocol in computer networks. 2014.

[27] Lin-Huang Chang, Ming-Yi Liao, and De-Yu Wang. Analysis of ftp over sctp and tcp in congested network. *JAIT*, 2007.

[28] Hui Min Weng, Ming He Huang, Hao Wang, Chang Qiao Xu, and Kai Liu. The analysis and simulation of stream control transmission protocol. *Applied Mechanics and Materials*, 433:1795–1799, 2014.

[29] Sheila Fallon, Paul Jacob, Yuansong Qiao, Liam Murphy, Enda Fallon, and Austin Hanley. Sctp switchover performance issues in wlan environments. In *Consumer Communications and Networking Conference, 2008. CCNC 2008. 5th IEEE*, pages 564–568. IEEE, 2008.

[30] Andrew Kelly, Gabriel Muntean, Philip Perry, and John Murphy. Delay-centric handover in sctp over wlan. *Transactions on Automatic Control and Computer Science*, 49(63):1–6, 2004.

[31] T Daniel Wallace and Abdallah Shami. A review of multihoming issues using the stream control transmission protocol. *Communications Surveys & Tutorials, IEEE*, 14(2):565–578, 2012.

[32] Laszlo Bokor, Arpad Huszak, and Gabor Jeney. Novel results on sctp multihoming performance in native ipv6 umts-wlan environments. *International Journal of Communication Networks and Distributed Systems*, 5(1):25–45, 2010.

[33] Guanhua Ye, Tarek Saadawi, and Myung Lee. Sctp congestion control performance in wireless multi-hop networks. In *MILCOM 2002. Proceedings*, volume 2, pages 934–939. IEEE, 2002.

[34] Sourabh Ladha, Paul D Amer, Jr AL Caro, and Janardhan R Iyengar. Improving file transfer in fcs networks. In *Military Communications Conference, 2003. MILCOM'03. 2003 IEEE*, volume 2, pages 944–948. IEEE, 2003.

[35] AN Isizoh, SO Okide, AOC Nwokoye, and CD Ogu. Throughput analysis of ieee802. 11b wireless lan with one access point using opnet simulator. *International Journal*, 2012.

[36] Omnet++user manual version 4.4.1, last accessed March 2014.

[37] András Varga and Rudolf Hornig. An overview of the omnet++ simulation environment. In *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, page 60. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.

[38] Biao Fang, Gaoming Huang, Yongbin Wang, Jun Gao, Chengxu Feng, and Zhen Li. Link 11 network simulation based on omnet++. In *Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering*. Atlantis Press, 2013.

[39] M Köksal. A survey of network simulators supporting wireless networks. *línea: http://www. ceng. metu. edu. tr/˜ e1595354/A% 20Survey*, 20, 2008.

[40] Irene Rüngeler, Michael Tüxen, and Erwin P Rathgeb. Integration of sctp in the omnet++ simulation environment. In *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, page 78. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.

[41] Omnet++user manual-screenshots, last accessed March 2014.

[42] Thomas Dreibholz, Erwin P Rathgeb, Irene Rungeler, Robin Seggelmann, Michael Tuxen, and Randall R Stewart. Stream control transmission protocol: Past, current, and future standardization activities. *Communications Magazine, IEEE*, 49(4):82–88, 2011.

[43] Thomas Dreibholz, Martin Becke, Hakim Adhari, Erwin P Rathgeb, Irene Rüngeler, Robin Seggelmann, and Michael Tüxen. Improvements to the sctp environment in the inet framework. *University of Duisburg-Essen, Institute for Experimental Mathematics, OMNeT++ Code Contribution*, 2012.

[44] Zina Jerjees. Design of interface selection protocols for multi-homed wireless networks. 2013.