TRAFFIC ENGINEERING FOR MULTI-HOMED MOBILE NETWORKS

By

Albert Yuen Tai Chung

THE UNIVERSITY OF
NEW SOUTH WALES

SCIENTIA

MANU ET MENTE

SYDNEY · AUSTRALIA

SUBMITTED IN FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
AT
THE UNIVERSITY OF NEW SOUTH WALES
SYDNEY, AUSTRALIA
MAY 2007

ORIGINALITY STATEMENT

'I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, or substantial proportions of material which have been accepted for the award of any other degree or diploma at UNSW or any other educational institution, except where due acknowledgement is made in the thesis. Any contribution made to the research by others, with whom I have worked at UNSW or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work, except to the extent that assistance from others in the project's design and conception or in style, presentation and linguistic expression is acknowledged.'

Signed ....

Date ....10-5-2007..

COPYRIGHT STATEMENT

'I hereby grant the University of New South Wales or its agents the right to archive and to make available my thesis or dissertation in whole or part in the University libraries in all forms of media, now or here after known, subject to the provisions of the Copyright Act 1968. I retain all proprietary rights, such as patent rights. I also retain the right to use in future works (such as articles or books) all or part of this thesis or dissertation.

I also authorise University Microfilms to use the 350 word abstract of my thesis in Dissertation Abstract International (this is applicable to doctoral theses only).

I have either used no substantial portions of copyright material in my thesis or I have obtained permission to use copyright material; where permission has not been granted I have applied/will apply for a partial restriction of the digital copy of my thesis or dissertation.'

Signed ....

Date ...10 - 5 - 2007...

AUTHENTICITY STATEMENT

'I certify that the Library deposit digital copy is a direct equivalent of the final officially approved version of my thesis. No emendation of content has occurred and if there are any minor variations in formatting, they are the result of the conversion to digital format.'

Signed ....

Date ...10 - 5 - 2007...

To Carmen

Table of Contents

List of Tables

List of Figures

Acronyms

2G Second Generation Mobile Communication System

3G Third Generation Mobile Communication System

AIMD Additive Increase Multiplicative Decrease

BGP Border Gateway Protocol

BS Base Station

CN Corresponding Node

CWD Sender Congestion Window

CoA Care of Address

DNS Domain Name Service

OCEAN On-board Communication, Entertainment, And Information

FTP File Transfer Protocol

GGSN Gateway GPRS Support Node

GPRS General Packet Radio Service

GPS Global Positioning System

GSM Global System for Mobile communications

HA Home Agent

IETF Internet Engineering Task Force

IP Internet Protocol

IPv4 Internet Protocol version 4

IPv6 Internet Protocol version 6

ISP Internet Service Provider

LAN Local Area Network

LDV Link Distribution Vector

LIP Linear Integer Programming

LLL Least Loaded Link

MH-DCP Multi-Homed Data Control Protocol

MPLS Multi-Protocol Label Switching

MR Mobile Router

MRHA Mobile Router's Home Agent

NS2 Network Simulator Version 2

NAT Network Address Translation

NEMO NEtwork MObility

PC Personal Computer

PDA Personal Digital Assistant

PTV Public Transport Vehicles

QoS Quality of Service

RR Round Robin

SNR Signal-to-Noise Ratio

RTT Round Trip Time

TCL Tool Command Language

TCP Transmission Control Protocol

UDP User Datagram Protocol

UMTS Universal Mobile Telecommunications System

WiFi IEEE 802.11

WLAN Wireless Local Area Network

WMAN Wireless Metropolitan Area Network

WWAN Wireless Wide Area Network

WRR Weighted Round Robin

Abstract

This research is motivated by the recent developments in the Internet Engineering Task Force (IETF) to support seamless integration of moving networks deployed in vehicles to the global Internet. The effort, known as Network Mobility (NEMO), paves the way to support high-speed Internet access in mass transit systems, e.g. trains, buses, ferries, and planes, through the use of on-board mobile routers embedded in the vehicle. One of the critical research challenges of this vision is to achieve high-speed and reliable back-haul connectivity between the mobile router and the rest of the Internet. The problem is particularly challenging due to the fact that a mobile router must rely on wireless links with limited bandwidth and unpredictable quality variations as the vehicle moves around. In this thesis, the *multi-homing* concept is applied to approach the problem. With multi-homing, mobile router has more than one connection to the Internet. This is achieved by connecting the mobile router to a diverse array of wireless access technologies (e.g., GPRS, CDMA, 802.11, and 802.16) and/or a multiplicity of wireless service providers. While the aggregation helps addressing the bandwidth problem, quality variation problem can be mitigated by employing advanced traffic engineering techniques that dynamically control inbound and outbound traffic over multiple connections. More specifically, the thesis investigates traffic engineering solutions for mobile networks that can effectively address the performance objectives, e.g. maximizing profit for mobile network operator, guaranteeing quality of service for the users, and maintaining fair access to the back-haul bandwidth. Traffic engineering solutions with three different levels of control have been investigated. First, it is shown, using detailed computer simulation of popular applications and networking protocols (e.g., File Transfer Protocol and Transmission Control Protocol), that packet-level traffic engineering which makes decisions of which Internet connection to use for each and every packet, leads to poor system throughput. The main problem with packet-based traffic engineering stems from the fact that in mobile environment where link bandwidths and delay can vary significantly, packets using different connections may experience different delays causing unexpected arrivals at destinations. Second, a *maximum utility* flow-level traffic engineering has been proposed that aims to maximize a utility function that accounts for bandwidth utilization on the one hand, and fairness on the other. The proposed solution is compared against previously proposed flow-level traffic engineering schemes and shown to have better performance in terms of throughput and fairness. The third traffic engineering proposal addresses the issue of maximizing operator's profit when different Internet connections have different charging rates, and guaranteeing per user bandwidth through admission control. Finally, a new signaling protocol is designed to allow the mobile router to control its inbound traffic.

Acknowledgments

First, I wish to thank my family and friends for their tremendous love and support throughout my life. I would not have been able to complete this journey without your continuous love and encouragement.

Second, I wish to express my gratitude to my supervisor Associate Professor Mahbub Hassan, for his overall guidance throughout my research and spending many hours discussing and reviewing the draft manuscripts of this thesis.

Third, I like to thank the staff in Vodafone Australia, namely: Mr. Michael Woodman, Mr. Ryan Payne, and Mr. Andrew Goldstiver for their value contributions to the definition and regular reviews of this research project. Also, I acknowledge the support of Dr. Salil Kanhere and Dr. Chun Tung Chou from the University of New South Wales; and Dr. Kun Chan Lan and Dr. Lavy Libman from the National Information and Communications Technology Australia (NICTA), for their time and effort in providing very helpful discussions and critique on my research.

Fourth, I wish to acknowledge Irene Chan from the University of New South Wales, for her efforts in collecting the field data for our GPRS measurements project as part of her honors thesis research project.

Next, I wish to express my appreciation to Vodafone Australia, the Australian Research Council (ARC), and University of New South Wales for their generous

financial support in providing me with the Australian Postgraduate Award Industry (APAI) scholarship and Supplementary Engineering Award (SEA) scholarships.

Finally, many thanks to my colleagues: Frederick Yik, William Lau, Wen Hu, Filip Rosenbaum, Dennis Pong, Kevin Lee, and Emily Lee for the great friendships we have established.

Publications

Optimising profit and performance for multi-homed mobile hotspots, Albert Chung and Mahbub Hassan. Proceedings of IEEE International Conference on Communications (ICC), Seoul, Korea, 16-20 May, 2005.

Traffic distribution schemes for multi-homed mobile hotspots, Albert Chung and Mahbub Hassan. Proceedings of IEEE Vehicular Technology Conference-Spring, Stockholm, Sweden, 30 May - 1 June, 2005.

Understanding the effect of environmental factors on link quality for on-board communications, Irene Chan, Albert Chung, Kun Chan Lan, Lavy Libman and Mahbub Hassan. Proceedings of IEEE Vehicular Technology Conference-Fall, Dalls, USA, 25-28 September, 2005.

Chapter 1

Introduction

1.1 Background

Over the past decade, mobile wireless communications have enjoyed a phenomenal success, with both voice and data communication services becoming indispensable for an ever-growing number of people. This applies in particular to passengers on Public Transport Vehicles (PTV), such as regular commuters in metropolitan areas or long-distance travelers, for whom the ability to communicate and access information while on the move is crucial to maintain productivity or provide entertainment during travel. However, current PTVs do not provide communication support for the PTV passengers; hence, mobile end users are restricted to simply connect via a wireless link directly to the static infrastructure (e.g. a base station in a cellular system). As a result, the communication abilities of mobile users are limited to accessing services and applications that are common in the static world, such as telephone calls and Internet access.

Recently, there are strong commercial [1–5] and research interests [6–9] in equipping PTVs with high speed on-board Local Area Network (LAN) to provide Internet

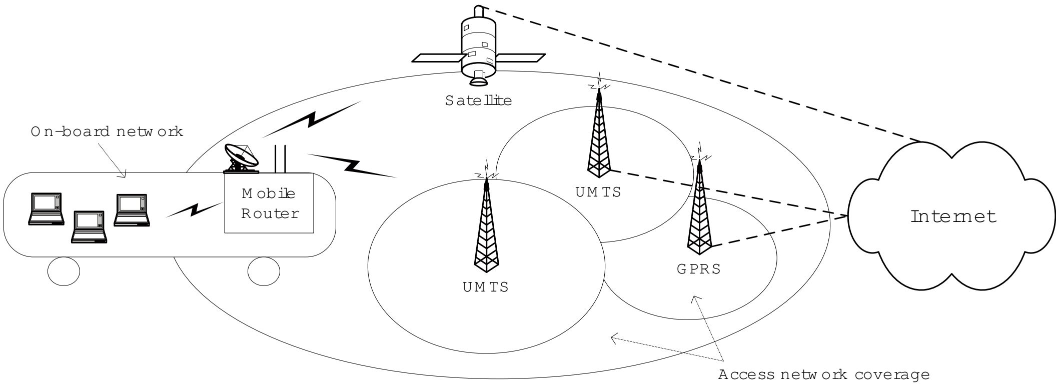

Figure 1.1: On-board communication architecture

connectivity to its passengers. These on-board networks present a new Internet connection paradigm for mobile users. To enjoy Internet services, passengers simply connect their mobile telephones, laptops, or Personal Digital Assistant (PDA) to the on-board Mobile Router (MR), which is connected to the Internet via wireless access technologies. When the system is in transit, the mobility of the entire LAN is managed transparently by the MR using an extension of mobile IP [10]. The on-board network architecture is shown in Figure 1.1.

There are several advantages in the on-board connection paradigm:

- *Single subscription* - The on-board network architecture can be seen as an extension to the highly popular public hotspot model, where users are able to enjoy Internet connectivity in public locations (e.g. cafes, airports, shopping malls etc) with a wireless LAN card. Consequently, on-board network users can enjoy extended Internet coverage in public vehicles without subscribing to other wireless access technologies. Of course, users can still subscribe to other mobile Internet services if they wish to use Internet services in other locations that are not covered by the static and on-board networks.

- *Access technology independency* - For users who only require Internet connectivity in the static and on-board network coverage areas, they do not need to worry about constantly upgrading their subscriptions and/or hardware due to the continuing release of new wireless access technologies. With a single LAN card, on-board network users can basically treat the on-board network as an access black-box and leave the hardware upgrades to the on-board network operator.

- *Transparent mobility management* - For on-board systems that implements a network mobility protocol, such as the NEtwork MObility (NEMO) protocol [6], the MR can handle the mobility management for the entire on-board network. On-board users will be able to enjoy Internet without worrying about performing mobility management for their connections.

- *Efficient bandwidth usage* - We can expect that passengers traveling together in a PTV are likely to access the same popular content items such as news, travel information, and entertainment etc. With an on-board LAN architecture, such items can be cached in an on-board data server and then re-distributed to the on-board users locally, instead of having users retrieve similar content over and over again. Likewise, real-time streaming content (e.g. an Internet radio station or Internet television channel) need only be transferred via the MR once, and can then be multi-cast to the on-board users.

- *On-demand wireless services* - Not everyone will require Internet access at any time or any location while they are on the move. Providing Internet coverage in

PTVs will be enough for most users and on-board network providers can provide on-demand wireless services like the current public hotspot models. This means that users only need to pay for on-demand Internet access services and are not tied up in paying expensive monthly subscriptions.

- *Power consumption* - In the on-board connection paradigm, the on-board user devices need to communicate only locally rather than with a remote wide-area network infrastructure. This translates to a significantly reduced power consumption and consequently to a prolonged battery life.

1.2 Motivation

One of the major challenges for the MR is to handle the large amount of data traffic routed between the on-board network and the Internet, since an on-board network can carry from tens (e.g. buses) to hundreds (e.g. trains, planes) of passengers. As the wireless access links are both bandwidth-limited and are less stable than fixed access links, we proposed to use a technique called *multi-homing*, where the MR is connected to the Internet via *multiple* wireless access technologies (e.g. GPRS, UMTS, CDMA2000, WiMax, Satellite) and/or multiple service providers. There are several benefits of multi-homing:

- *Increased bandwidth* - As wireless access technologies are typically bandwidth limited, the on-board system can increase its aggregate access bandwidth by connecting to the Internet via multiple access links. This is crucial for the on-board network since it needs to handle large amount of data traffic for the on-board users.

- *Ubiquitous Access* - Multi-homing provides a larger wireless Internet service coverage for the on-board system, since each wireless access network has different coverage areas. This is because different wireless access technologies will have different coverage sizes; and, even for networks with the same wireless access technology, the coverage areas will not be the same as different operators will have their own coverage planning strategies.

- *Resilience to network disruptions* - If the on-board network only relies on a single wireless access link for Internet connectivity, the entire system will lose connectivity whenever there is a link outage. This is particularly problematic for wireless access links, since they tend to have stability issues especially when the system is in transit. Thus, if the system is equipped with multiple access interfaces, the system can divert traffic away from failed links to ensure that Internet connectivity is preserved for the on-board users.

- *Wider range of services* - Since different access technologies have different performance characteristics, a multi-homed system can provides a wider range of Internet services to the passengers. For example, if the system only depends on a single satellite link; which is characterized with high propagation delay; the system may have difficulty in providing real-time streaming services for the on-board users. Hence, equipping the MR with multiple access technologies allow the system to serve the requirements for a wider range of applications.

Effective *traffic engineering* is critical for successful realization of multi-homing in the on-board connection paradigm. In the context of on-board systems, we define traffic engineering as the way in which the MR distributes the system data traffic among the multiple wireless access links. For the on-board network architecture; where the entire network is on the move; the challenge with traffic engineering arises from the fact that the mobile multi-homed system must be able to send and receive large amount of user data traffic over the wireless access links. With the movement of the network and the nature of wireless access technologies, the system will have more dynamic link characteristics (i.e. bandwidth and delay) than static multi-homed networks that depend on fixed-line access networks. To maximize the throughput performance for the on-board users, the system needs to adapt to the bandwidth variations in each access link. For the on-board network, we have identified two types of bandwidth variations:

- *Spatial variations* - When the system is in transit, the set of available access networks; i.e. the number of networks available; will not be constant since there are *spatial* variations in the physical network coverage of each wireless access network. For example, the coverage of a Wireless Metropolitan Area Network (WMAN) access technology is highly unlikely to be able to match the coverage of a satellite network due to the physical constraints of such access technology. Also, even with the same access technology, the network service providers have different deployment strategies and hence their coverage areas will not be the same.

- *Temporal variations* - For each wireless access link, the actual bandwidth that is available to the MR will exhibit *temporal* variations due to the contention (for wireless resources) with other users in the same access network. Each access network will have different set subscribers; hence, the temporal bandwidth variations for each network will not be the same in each trip.

To adapt to these network resource variations, the MR needs to perform *dynamic* traffic engineering among the available wireless access networks. When the PTV is in transit, the MR needs to know how to distribute the passengers' data traffic among the available network connections at different points of a trip. Given that there are multiple links associated with the multi-homed systems, there are many possible ways of distributing the data traffic among these interfaces. For example, one algorithm may distribute the traffic in a way that maximizes the throughput of the system, while another algorithm may distribute the traffic in a way that provides a fair bandwidth allocation to the users' data flows. This brings the need to study the design of *traffic engineering algorithms* for the multi-homed on-board network system, which is the main focus of this thesis. The aim of these algorithms is to exhibit certain control over the way in which traffic engineering is performed by the multi-homed on-board system. To compare the performance of the different traffic engineering algorithms, we also identify the important metrics which needs to be considered in traffic engineering algorithm design.

We have identified three possible types of *traffic engineering schemes* for the mobile multi-homed networks architecture; namely, the *user-based*, *flow-based*; and *packet-based* traffic engineering schemes. The difference between these schemes lies in the switching granularity that can be deployed in the multi-homed MR. In this thesis, we will investigate and design traffic engineering solutions for each of these traffic engineering schemes. The selection of traffic engineering scheme is a highly important design issue for the multi-homed on-board system; since as we will discover in this thesis, not all of these traffic engineering schemes are suitable for the on-board network and choosing an incorrect traffic engineering scheme will severely degrade the throughput performance of the users' data flows.

1.3 Contributions

The contributions of this work can thus be summarized as follows:

- Firstly, we showed via detailed computer simulation of popular applications and networking protocols (e.g., File Transfer Protocol (FTP) and Transmission Control Protocol (TCP)), that packet-level traffic engineering which makes decisions of which Internet connection to use for each and every packet, leads to poor system throughput. The main problem with packet-based traffic engineering stems from the fact that in mobile environment where link bandwidths and delay can vary significantly, packets using different connections may experience different delays causing unexpected arrivals at destinations.
- Secondly, we proposed a set of *maximum utility* flow schedulers that aim to maximize a utility function that accounts for bandwidth utilization on the one hand, and fairness on the other. The proposed flow schedulers are compared against previously proposed flow-level traffic engineering solutions and shown to have better performance in terms of throughput and fairness.
- Thirdly, we proposed novel user distribution algorithms that aim to maximize the on-board system operator's profit when different Internet connections have different charging rates, and guaranteeing per-user bandwidth allocation through admission control. We showed that having a prior knowledge on the link bandwidth variations does not provide additional profit to the system, but instead it allows the on-board network to limit the service disruption probability

for the users.

- Finally, as the majority of users' data traffic are in the downlink direction, we designed the Multi-Homed Data Control Protocol (MH-DCP) that allows the MR to exhibit traffic engineering control over the passengers' downlink data traffic. The protocol is specifically designed for both the user-based and flow-based traffic engineering schemes.

1.4 Thesis organization

The rest of the thesis is organized as follow.

In the next chapter, we provide an in-depth background tutorial on multi-homing systems, and review the past research in the different types of multi-homed systems. We also present a thorough classification on the different types of multi-homed systems and compare their characteristics.

Chapter 3 presents the Network Mobility (NEMO) multi-homed architecture, which is the research platform for the work presented in this thesis. We discuss the traffic engineering opportunities associated with this architecture, and discuss the traffic engineering schemes in detail. Also, we highlight the traffic engineering research issues and challenges under this architecture.

In Chapter 4, we investigate the feasibility of using the packet-based traffic engineering scheme in the mobile multi-homed architecture, by simulating the performance of some well known packet-scheduling algorithms and their corresponding flow-scheduling algorithms in a highly popular network simulator.

In Chapter 5, we formulate the flow-scheduling problem and propose several flow-based traffic engineering algorithms that aim to maximize both the link utilization and the fairness of the system. We perform extensive simulations to compare the performance of the proposed algorithms with previously proposed flow scheduling algorithms.

In Chapter 6, we formulate the traffic engineering problem that aims to maximize the profit of the mobile hotspot operator under a volume-based charging model, while maintaining bandwidth guarantees for the on-board users. We propose two user-based traffic engineering schemes and present extensive simulation results to highlight the performance of the proposed algorithms.

In Chapter 7, we discuss the design of the mobile multi-homed downlink control protocol (MH-DCP), which is a signaling protocol that allows the MR to exhibit downlink data traffic control under the flow-based and user-based traffic engineering schemes. We will also present a thorough discussion on the various signaling procedures that will arise in the operation of the protocol.

In Chapter 8, we conclude the thesis by summarizing the results of this thesis, and outline a list of all potential future work.

Chapter 2

Background and literature review

In this chapter, we provide a detailed overview on the concept of multi-homing. We discuss the various multi-homed models that have been proposed and the past related work for each of these models.

2.1 Introduction

In the past, there had been considerable research on equipping users or networks with multiple access links. Some of the reasons for equipping such systems with multiple access links include: increasing the aggregate bandwidth, providing link resilience, extending wireless access coverage, and reducing access costs etc.

In this chapter, the aim is to provide a detailed overview on the concept of multi-homing. We discuss the various multi-homed models that have been proposed and the past related work for each of these models. To do this, we first classify multi-homed systems into three major categories, namely: the static multi-homed network model, the mobile multi-homed user model, and the more recently proposed mobile multi-homed network model. The classification is important since there are different issues and challenges involved with each of the models. For example, in a multi-homed network model, fairness is an important issue for the system since there are multiple users that are trying to send data over the multi-homed system, while the issue does not apply to a multi-homed host model since there is only a single user entity in that system.

The rest of this chapter is organized as follows. In Section 2.2, we provide a detailed discussion on the static multi-homed network model. In Section 2.4, we present the details of the mobile multi-homed host model. In Section 2.4, we discuss the mobile multi-homed network model. In Section 2.5, we conclude this chapter with a summary of its contributions.

2.2 Static multi-homed networks

We define *static* multi-homed networks as non-mobile fixed networks that connects to the Internet via multiple access interfaces. Static multi-homed networks are also commonly referred to as *site* multi-homing. Users in a static multi-homed system typically route their incoming and outgoing data traffic through a multi-homed gateway, which is connected to the Internet via multiple network interfaces. The static multi-homed networks are typically connected to the Internet via fixed wired links, and each network interface may be connected to different access technologies and/or different service providers. An illustration of the static multi-homed system architecture is shown in Figure 2.1.

One of the popular uses of a static multi-homed system is to provide link resilience for the network [11]. In this case, the multi-homed gateway only uses one of the links for traffic routing, and in the case of a link outage the multi-homed gateway will switch the data traffic over to another access link. To ensure that the downlink traffic are

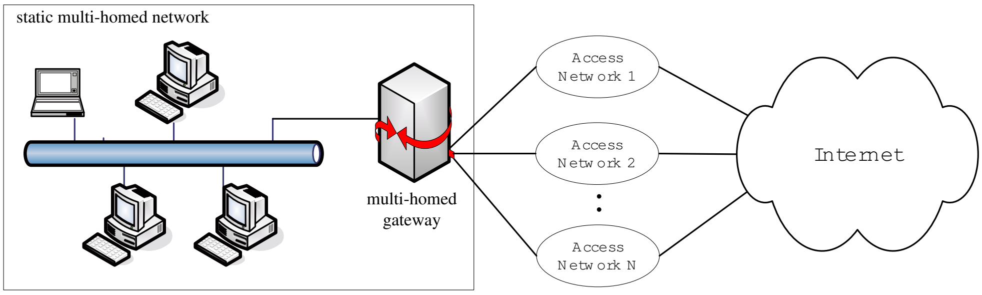

Figure 2.1: Static multi-homed network model

routed over the new access link, the router will advertise the Internet Protocol (IP) address of the new access link using conventional routing protocols (e.g. Border Gateway Protocol (BGP)). In this type of multi-homed system, the multi-homed gateway can only select a single access link in which the aggregate data traffic will be routed over to.

For multi-homed systems that aim to make use of the multiple access links simultaneously [11, 12], one of the tasks of the multi-homed gateway is to split the users' traffic among the multiple network interfaces connected to the Internet. To do this, the gateway needs to make a decision on which interface each data packet should be routed through. In conventional IP routing, the gateway has a more straight-forward task in exhibiting routing control over the uplink traffic (i.e. traffic going from users to the Internet), than the downlink traffic (i.e. traffic going from the Internet to the users). The gateway can control the routing of the uplink packets by forwarding the packets along any one of the available network interfaces, without requiring any modifications or special techniques. The data packets will be forwarded to the Corresponding Node (CN) by the intermediate routers using conventional routing techniques. In contrast, it is not possible to schedule the users' downlink traffic over the multiple access links because in conventional routing, a network (or subnet) is usually only associated with a single IP prefix. In order to make use of the multiple access links for the downlink traffic, the multi-homed gateway can either participate in BGP routing (which supports multi-homing) or make use of existing techniques such as Domain Name Service (DNS) load-balancing which are highly popular for balancing the web traffic among multiple web servers [13]. The main drawback with these approaches is that the multi-homed gateway cannot exhibit control over which access link each individual downlink packet or flow will be routed over from.

For systems based on Network Address Translation (NAT) [14]; a highly popular network configuration among small to medium size networks; the multi-homed gateway can exhibit a finer level of downlink traffic engineering control. To allow the internal users to use private IP addresses, the NAT gateway is responsible for performing network address translation on all uplink packets. The packet address translation involves replacing the IP header of the uplink packets with the gateway's external IP address and a random port number that is chosen to identify this flow. A NAT table is responsible for storing all these flows-to-port mapping. For all incoming packets destined to the gateway's IP address, the gateway performs a lookup on the NAT table using the destination port number and replaces the destination IP address and port number with the user's internal IP address and port number.

The multi-homed NAT gateway can control which interface each *flow* is routed over back to the gateway by selectively performing network address translation over each flow. This is done by encapsulating the uplink packets with any of the available IP addresses mapped to each network interface, and hence the external CNs will see the source address of the packets as one of the external IP addresses selected by the multi-homed gateway. Consequently, in conventional IP routing, all packets for the flow will be forwarded and received over the same interface via the same access network technology. The advantage with this approach is that the system can utilize the multiple access links simultaneously. However, there are three limitations with this approach. Firstly, the network access selection is restricted to only being applied on a per-flow basis. This is because performing selective routing control over each individual packet means packets of the same flow will have different source address encapsulation, which breaks end-to-end flow semantics. Secondly, a flow cannot be *re-scheduled* onto another link during its lifetime, since modifying the table entry during the flow's lifetime means changing the source address of the uplink packets, which again breaks the end-to-end flow semantics. Finally, using the source address to control downlink packets means that the system is restricted to routing both the uplink and downlink traffic over the same network interface, which reduces the system's switching flexibility.

In [15], a new multi-path switching scheme based on *flowlets* was proposed. This scheme aimed to combine the advantages of packet and flow switching; by offering a reasonably fine switching granularity while ensuring out-of-order packet arrivals are minimized. A flowlet is defined as a burst of packets in a TCP flow; where all the packets in a flowlet have an inter-arrival time that is less than the flowlet threshold T. To ensure packet re-ordering does not occur, the inter-arrival time threshold must be at least equal to the latency between the diverging and converging point for the set of parallel paths. The diverging router maintains a hash table containing the packet time-stamps of each flow. For all packets arriving at the diverging router, the router computes a hash on the packet's *flow ID*; which is a combination of the packet's source and destination IP addresses and port numbers; to find the flow's entry in the hash table, and measures the inter-arrival time of the current packet and the previous packet in this flow. If the inter-arrival time is less than the flowlet threshold T, the packet must follow the path currently mapped to this flowlet. Otherwise, the router classifies this as a new flowlet and can schedule this onto a different path without the risk of packet re-ordering. The diverging router performs load balancing with a simple flowlet scheduler that maps each new flowlet to the least utilized path, which is defined as the link with the least amount of data transferred over it. The main limitation with flowlet switching is its dependency on the latency between the diverging and converging points. For fixed core multi-path networks, this value may be stable and easily computed in advance. But for mobile multi-path networks which depend on wireless access links from different service providers, the convergence point of these parallel paths may be different for different locations of a trip and may not be easily computed beforehand. Even if the convergence can be determined on-the-fly, the latency may be highly fluctuating due to the wireless nature of the access links, which increases the chance of packet re-ordering at the end hosts.

A new online distributed traffic engineering protocol was proposed in [16]. The proposed protocol, named TeXCP, is designed to be used in Multi-Protocol Label Switching (MPLS) type networks where there are multiple paths to deliver data traffic from ingress to egress routers. The TeXCP protocol aims to *minimize the maximum load utilization* of all the links in the network, by adaptively moving traffic from over-utilized links to under-utilized links. To maintain load balance among the paths, the egress routers need to actively probe the multiple paths linking each

Ingress-Egress pair. This requires the TeXCP protocol to be supported by all the intermediate routers. The traffic splitting in the network is performed on a per-flow basis, and to prevent out-of-order packet arrivals the authors referred to a new flowlet switching technique [15]. However, many important details such as whether existing flows can be re-mapped to a new path, how the egress routers manages the flow to path mappings, and the effects of re-mapping existing flows to new paths were omitted. Extensive simulations were performed using real packet trace files and the protocol showed that the use of the TeXCP protocol was able to achieve within a few percent of the optimal load balance.

Rost et.al., proposed several hash-based algorithms which performed rate-aware splitting of aggregate traffic over multiple paths [17]. The focus of this work was on multi-path routing in wired core networks. The proposed solution computes a hash based on flow identifiers, which is a unique combination of the source IP address, destination IP address, source port number, and the destination port number. The system splits the traffic over the multiple outgoing links according to the hash value computed for each flow. The advantage of hash-based flow switching is that it provides a more scalable version of flow-based switching, since the router does not need to maintain any per-flow state information. This is particularly important for core networks since there are potentially million of active flows going through a router at any point of time. Also, the flow switching nature ensures the probability of out-of-order packet arrivals is minimized. The disadvantage with this solution is like most flow switching solutions; it does not offer the fine granularity as packet switching and hence leads to a overall load in-balance among the paths. The proposed algorithms aimed to minimize this imbalance by considering the rates of the flows (i.e. the throughput of each link) in its traffic splitting decisions. By continuously adjusting the hash bucket sizes, it ensures that the traffic splitting vector is as close to the optimum splitting vector as possible. Performance metrics for the algorithms includes: measuring the load imbalance based on sample normalized correlation of the current and optimum splitting vectors (i.e. using dot product), and measuring *flow disruption* which is defined as the probability in which the traffic splitting node forwards packets (from an active flow) from one link to another. The system aimed to minimize the flow disruption metric since switching the data packets from one link to another can potentially cause out-of-order packet arrivals, which (the authors claimed) will degrade the TCP performance.

Yamai et.al., proposed a dynamic TCP flow scheduler for static multi-homed networks [18]. The proposed flow scheduler utilizes the connection setup time of the TCP flows to determine the best interface selection metric for each of these flows. To do this, the multi-homed router detects the arrival of the TCP connection setup packets (i.e. SYN packets) sent by the hosts inside the network, and forward duplicate copies of these packets to all the available network interfaces. Upon receiving the first response packet, the multi-homed router can determine which interface has the lowest communication setup time for this particular flow. It will then forward all subsequent packets from this TCP flow to the selected interface. Since the link selection is only performed at the start of the flow, this technique only works well for static networks with relatively stable link characteristics.

The issue of optimizing profit and performance for static multi-homed network was thoroughly investigated in [19]. The work focused on designing offline and online routing algorithms which distributes traffic in a way that maximizes the profit under the *percentile-based* charging scheme. Under this charging scheme, an Internet Service Provider (ISP) records the traffic volume a user generates during every 5-minute time interval. At the end of the complete charging cycle, the percentiles are sorted in ascending order according to the volume transferred over each timeslot, and the system needs to select the charging parameter q. The q value (between 1-99) is used to determine the charging volume, where the q-th percentile of the sorted 5-minute traffic volumes is selected as the charging volume x for the user. This charging scheme is designed for core ISPs networks, which is the context of this research. An online integral (heuristic) flow switching algorithm was proposed and it aimed to maximize the profit of the system by making predictions on the future flow traffic patterns.

2.3 Mobile multi-homed hosts

In the mobile multi-homed host model, mobile users carry communication devices that are equipped with multiple access network connections. Examples of these multi-homed devices includes: mobile phones; PDA; and laptop computers. Figure 2.2 illustrates the mobile multi-homed host architecture.

The mobile multi-homed host model differs to other multi-homed models in many ways. Firstly, the application context of the multi-homed host model is different since there is only a single user in the system. Thus, issues such as maintaining a fair bandwidth share between users and providing service differentiation between users do not apply in the multi-homed host model. Secondly, the traffic demand on the multi-homed host model is highly likely to be less than the multi-homed network models, again due to the fact that there is only a single user in the system. Finally, the mobile multi-homed host model assumes the users will utilize wireless access links

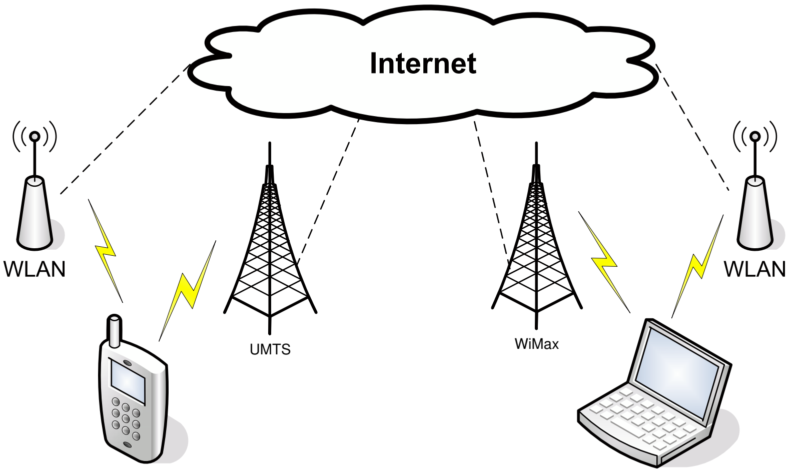

Figure 2.2: Mobile multi-homed host model

for mobility support, which are different to the fixed-line links since they are relatively more bandwidth-limited and have highly fluctuating link characteristics.

A new architecture idea for multi-homed hosts situated in physical proximity to spontaneously share Wireless Wide Area Network (WWAN) bandwidth was proposed in [20]. The proposed solution consists of a proxy-based inverse multiplexer (PRISM), that aims to provide efficient bandwidth aggregation by masking out the different Quality of Service (QoS) characteristics of WWAN links. The work is mainly focused on how the proxy (located in the Internet) schedules the users traffic over the multiple WWAN links. The proposed proxy solution performs packet stripping across multiple WWAN channels. By intelligently controlling the delivery of the TCP acknowledgement packets for the users, the proxy can minimize the probability of packet re-ordering at the receiving hosts. Since packet scheduling over heterogeneous wireless links requires precise link state information; which is difficult to obtain due to wireless channel dynamics; a new packet scheduling algorithm called Adaptive Scheduler (ADAS) is used to maximize bandwidth aggregation for the wireless links. The packet scheduler requires probing into the different TCP packet types, in which for all packet re-transmissions, the proxy forwards the packets over the link with the minimum Round Trip Time (RTT). For all other packets, the proxy sends them over the *least utilized* link, which is defined as the link with the minimum number of packets in-flight. The main drawback with this approach is that the solution is not transparent to the users, as the end hosts are all required to use a modified TCP implementation called *PRISM-TCP*. A prototype was implemented and simulations showed that the PRISM architecture was able to masks out the bandwidth; delay; and loss rate disparities to achieve close to optimum bandwidth aggregation for the users. Note that the experimental results only concentrated on the performance of scheduling a single PRISM flow (over multiple wireless links), which does not highlight the bandwidth competing nature of TCP.

Horde [21] is a new middle-ware that allows user applications to exhibit control over *network stripping*; which is defined as the construction of a high-bandwidth virtual channel from a collection of low bandwidth network channels. The motivation behind this work is driven by the need to provide QoS sensitive medical application (streaming) from ambulance vehicles over existing WWANs. The middle-ware allows applications to exhibit influence over the network stripping policy, and is specifically targeted at wireless channels where the bandwidth and delay characteristics are highly dynamic. The stripping mechanism aims to satisfy the QoS objectives as requested by the application, by determining which ADU (application data unit) will be mapped to each transmission slot (txSlot). Since the txSlots are highly dynamic in terms of QoS and charging rate, the mapping of the ADUs must be performed on-the-fly. To further improve the performance of the applications, the scheduler performs predictions on the channel characteristics. The major limitation with this approach is that it requires applications to be re-written in order to support the interaction between the application and the middle-ware.

A new network layer solution to perform flow splitting across multiple network interfaces at the IP level was proposed in [22]. The solution is transparent to the transport layer, as the IP packets are independently tunneled over to the receiver using multiple wireless paths. Sender-side scheduling mechanisms were proposed to mask the disparity among the multiple links (i.e. bandwidth, RTT) to avoid undesirable behavior such as out-of-order packet arrival at the receiver.

A new transport layer protocol called Reception Control Protocol (RCP) was proposed in [23]. The motivation behind this is that locating the intelligence of a transport protocol at the mobile host that is adjacent to the wireless link(s) can result in distinct performance advantages. A multi-homed extension was proposed, where the protocol relies on a packet scheduler that reduces the probability of out-of-order packet arrivals by continuously monitoring the RTT of the wireless links and scheduling the packets delivery accordingly. The use of the real time RTT values allow the scheduler to make predictions on the packet arrival times.

The recently announced IEEE 802.11 (WiFi) mobile phones [24] are one of the first commercially available multi-homed devices aimed at mobile users. WiFi phones allow users to connect their phone to both the cellular networks and WiFi networks. The motivation behind this is that when a cellular user moves into the coverage of a WiFi network, the phone can switch their connection from the cellular network to the WiFi network. Once a user moves out of the WiFi coverage, the WiFi phone will then switch the connection back to the cellular network. This approach allows cellular operators to extend their coverage using WiFi networks, which are relatively cheaper to deploy than cellular networks. Also, the WiFi networks provide significantly higher bandwidth than the cellular networks, which makes it attractive for data services. In addition, there have been proposals to support *vertical handoffs* [25–27] in these systems, which allows users to preserve existing data or voice sessions when their WiFi phones switches from one network connection to another.

2.4 Mobile multi-homed networks

The mobile network model is a relatively new concept that was specifically designed to equip vehicles with an on-board network, which provides Internet connectivity to the on-board passengers. Due to the movement of the vehicle, the mobile networks must reply on wireless access technologies for Internet connectivity. Consequently, since the wireless access links are both bandwidth limited and have highly fluctuating link characteristics, there have been proposals to connect these on-board networks to the Internet via multiple access links [28, 29]. Equipping such systems with multiple wireless access links allow the system to have a higher bandwidth and more stable link to the Internet. Figure 2.3 illustrates the mobile multi-homed network model.

The mobile multi-homed network architecture differs from the other multi-homed models, as the system needs to handle large amount of data traffic (from a group of users) over several highly dynamic wireless access link. To make efficient use of the multiple access links, the system needs to be able to distribute the users' data traffic over those links. Due to the relatively smaller size of such networks, the system is not

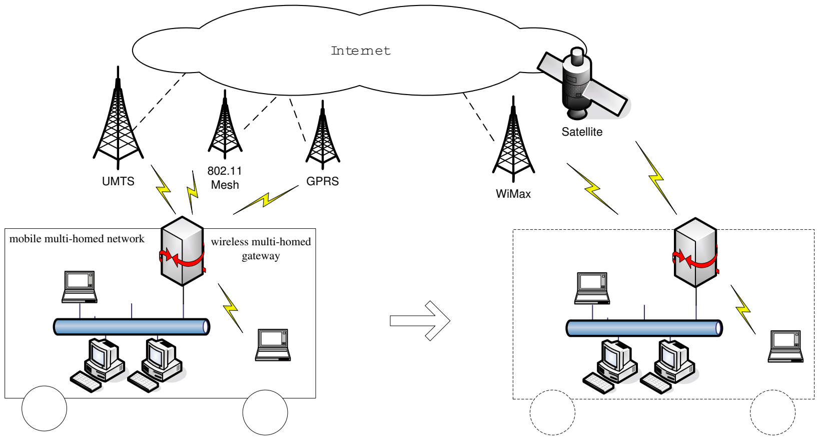

Figure 2.3: Mobile multi-homed network model

expected to be able to participate in BGP routing; hence, this leaves the NAT based solution as the more suitable existing multi-homed routing solution for the mobile multi-homed network model.

As discussed in Section 2.2, the NAT-based multi-homed solution allows the system to distribute the users' data traffic on a per-flow basis. However, the drawback with this approach is that the system does not support changes in the IP addresses of each individual access links; which can occur if the wireless access network changes the IP address of the connection; e.g. changing the Gateway GPRS Support Node (GGSN) for users in a General Packet Radio Service (GPRS) network. In this case, all existing sessions mapped to this network will be terminated since the change of address will break the end-to-end connection semantics. The recently Internet Engineering Task Force (IETF) standardized NEMO protocol is an extension to mobile IP that aims to provide transparent mobility support for users in mobile networks. Multi-homed extensions based on the NEMO protocol have been proposed and this will be thoroughly discussed in Chapter 3.

The Mobile Access Router (MAR), is a wireless multi-homed device that can be placed in moving vehicles (e.g. car, bus, train) to enable high speed data access [28]. The MAR architecture is a NAT-based multi-homed solution where the MR controls the traffic distribution over the multiple links by using different source-NAT addresses for different flows. Flow splitting is not supported in the MAR architecture since it will break the end-to-end connection semantics. The NAT-based solution also requires that the uplink and downlink traffic of a flow must be routed over the same wireless access link. The results presented in this work mainly focused on the comparison of: technology; network; and channel diversity in existing overlapping wireless access networks (GPRS and 802.11). In the presented MAR prototype, a simple round robin TCP flow scheduler was used to achieve load balancing effects. The authors acknowledged the load-balancing limitations of this scheduler, due to the fact that each flow will have different start and finish times (i.e. different transfer sizes). Hence, they made a call for further research in the design of suitable flow schedulers for mobile multi-homed networks.

Chesterfield et.al., discussed how the diversity of cellular links can be exploited to enhance multimedia streaming [30]. To highlight the diversity of WWANs, a measurement test-bed that involves connecting a live GPRS network connected to the Cambridge research lab was built. Also, a second Wireless Local Area Network (WLAN) connection was used to see the effects of technology diversity. Extensive measurements were performed and the following observations were found:

- When the GPRS reliability mode is enabled, there is a significant variation in the propagation times across the link due to the Automatic Repeat Request (ARQ) mechanism. The variation of the packet inter-arrival times is an order of magnitude larger (range between 70-610ms) than typically seen in wired links. If ARQ is disabled, the variation of the inter-arrival times is much tighter (195 - 255 ms). The variance of the inter-arrival times is minimized at the expense of higher packet loss (3.8%) and the loss seems to occur randomly.

- Errors occur in close proximity to each other. It was found that when a packet error occurs, there is a high probability (90%) that the error affects more than one packet. In all of the cases where there are multiple packet errors, the distance between subsequent errors never exceeds three slots.
- Technology diversity provides the most uncorrelated behavior, since there can be an order of magnitude difference in the network performance such as RTT. The most uncorrelated behavior for similar technologies occur between separate *network providers* operating in different frequency spectrums.

Based on the above observations, a new streaming tool which utilizes application based error recovery was proposed. Techniques such as: stripping packets across multiple channels to distribute the effects of sustained loss over the network; designating a single channel for parity packets; and a combination of distributing low bandwidth protected data stream over reliable channels, and distributing high bandwidth unprotected data over unreliable channels were studied. The conclusion drawn from the results is that exploiting uncorrelated performance over separate channels provided the highest performance improvement.

The issue of *ingress filtering* for multi-homed nodes that implement layer-3 mobility protocols such as Mobile IP and NEMO, was thoroughly addressed in [31]. Ingress filtering is a layer-3 technique used to prevent *IP address spoofing*; where users tries to send IP packets to other hosts using a fake source IP address. The ingress filtering router can do this by checking the source address of all the packets that it needs to forward against a list of valid subnet prefixes. The problem with using ingress filtering in a multi-homed environment occurs when a multi-homed node is associated with multiple Home Agent (HA)s. If the multi-homed node tries to send packets to one of the HAs using the Care of Address (CoA) that belongs to another one of its associated HAs; which may occur when one of the access links are inaccessible; the HA that receives the packet may perform ingress filtering and reject that packet. To address this issue, the authors proposed a new IP tunnel re-establishment technique that allows these mobile multi-homed nodes; which includes individual mobile hosts and mobile networks; to achieve ubiquitous connectivity to the Internet.

Paik et.al., outlined the benefits of equipping a mobile network with multiple MRs in [29]. The authors extended the work by proposing a traffic sharing and session preservation architecture in [32]. The proposed framework does not support the simultaneous use of multiple MRs, and hence the aggregate data traffic cannot be split among the multiple MRs. A location-based MR selection algorithm was proposed, where the system selects the use of a MR according to the physical location of the MR in the mobile network.

The concept of Multiple Mobile Routers Management (MMRM) was also discussed in [33]. The proposed framework allows MMRM-enabled mobile users to access the Internet via multiple MMRM-enabled MRs, which include the fixed MRs provided by the mobile network, and the MRs provided by the mobile user themselves (i.e. the Internet connections provided by the mobile users' access network interfaces). By distributing the users' data traffic across multiple paths, the MMRM framework

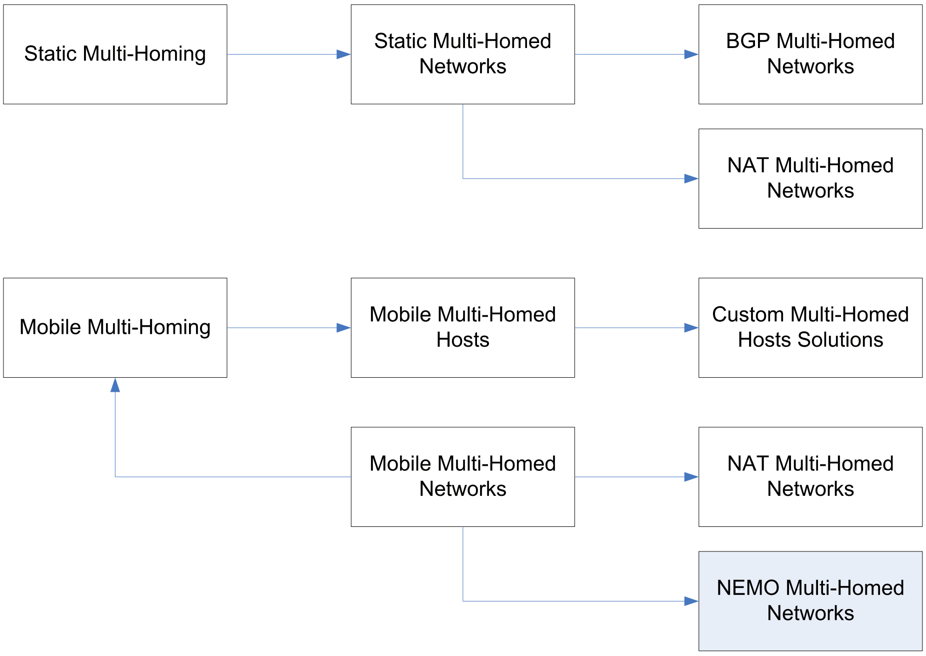

Figure 2.4: Classification of the various multi-homed systems

increases the redundancy and the bandwidth for the mobile users.

Suciu et.al., designed a new network interface selection framework for NEMO-based multi-homed MRs in [34]. The proposed framework includes a profile manager that specifies the performance characteristics and connection establishment instructions for each access link, and a selection decision module that makes decisions on which access link should be used for data communication. The work does not discuss in detail how the on-board system should deal with the network(s) selection problem (i.e. traffic engineering).

2.5 Summary

In this chapter, we provided a thorough background study on multi-homed systems, and the past work which relates to the focus of this thesis. As we can see, there has been research on a wide range of different issues and challenges associated with multi-homed systems; including (and not limited to) layer-3 mobility support, handoff management, traffic engineering, middleware, network selection frameworks, and new multi-homed architectures. Given the wide range of multi-homed contexts, we classified multi-homed systems into three main types, namely: static multi-homed networks; mobile multi-homed hosts; and mobile multi-homed networks and compared the significant issues and differences between these multi-homed models. A detailed classification on the various types of multi-homed systems presented in this chapter is shown in Figure 2.4.

In the next chapter, we will discuss the more recently proposed NEMO multihomed architecture, which is the research platform for the work presented in this thesis. We will highlight the advantages with such systems and compare the differences between our research model and the other multi-homed models discussed in this chapter. We will also highlight the new traffic engineering opportunities, and the research issues and challenges associated with the NEMO multi-homed architecture.

Chapter 3

NEMO multi-homed architecture

In this chapter, we provide a thorough discussion on the NEMO multi-homed architecture, which is the research platform for the traffic engineering solutions proposed in this thesis. We highlight the traffic engineering opportunities, and discuss the research issues and challenges under this architecture.

3.1 Introduction

The mobile network architecture is based on the IETF NEMO protocol [10,35], which is currently the only standardized network mobility protocol designed for mobile networks. It is important to base our research on a standardized protocol, so our proposed solutions can be deployed in a wider range of multi-homed systems than ones that utilizes proprietary technologies. The NEMO protocol provides transparent network layer mobility support for the on-board passengers, which means the users' transport layer sessions can be preserved whenever the mobile network changes the CoA assigned to its access link. The change of CoA usually occurs when the system is in transit; e.g. switching GGSNs in a GPRS network [36].

Recently, the newly established IETF MONAMI6 working group [37] is studying

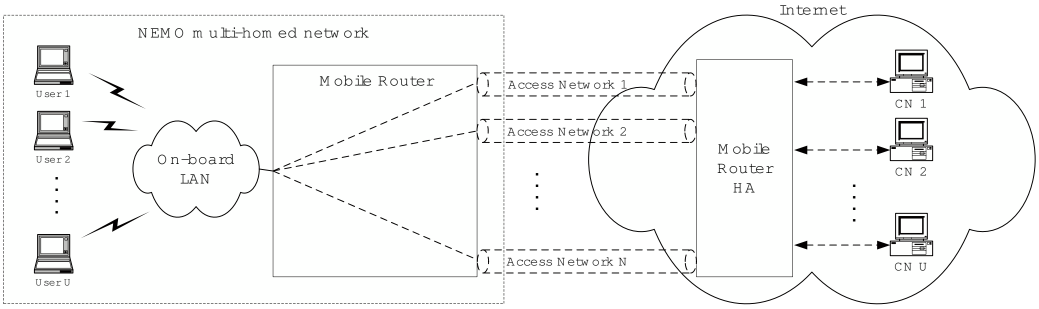

Figure 3.1: NEMO multi-homed architecture

the issues and benefits of using multi-homing in the NEMO architecture. The aim is to equip mobile networks with a multi-homed MR, in order to increase the aggregate bandwidth of the system and provide resilience in the case of access link failures. Since our research is based on this new mobile multi-homed network architecture, the aim of this chapter is to provide an in-depth overview and discuss the traffic engineering issues and challenges associated with the NEMO multi-homed model.

The rest of this chapter is organized as follows. In Section 3.2, we first provide an architectural overview of the NEMO multi-homed system. In Section 3.3, we discuss the traffic engineering opportunities that are inherited from the unique characteristics of the NEMO multi-homed model. In Section 3.4, we highlight the research issues and challenges in the NEMO multi-homed architecture. In Section 3.5, we conclude this chapter with a summary of its contributions.

3.2 NEMO multi-homed architecture

The IETF standardized NEMO protocol [10] is designed to provide transparent mobility to on-board users. In the standard NEMO model; where the mobile network is connected to the Internet via a single wireless interface; the MR is responsible for transferring the users' data traffic to and from a *home agent* that is located in the Internet. The role of the home agent; which we name it as the Mobile Router's Home Agent (MRHA); is to act as the Internet gateway for all the data traffic to and from the mobile network.

In the NEMO system, the on-board passengers are assigned an IP address that belongs to a subnet, that is under the routing responsibility of the MRHA. For all data packets destined to the on-board users, they are first routed to the MRHA. To provide transparent mobility to the on-board users, the MRHA is responsible to keep track of the MR's current CoA while the mobile network is in transit and tunnels the packets to the MR accordingly. Once the MR receives the tunneled packets, it will decapsulate the packets and passed them to the destined on-board user. Since the NEMO protocol is an extension to mobile Internet Protocol version 6 (IPv6); where ingress and egress filtering are implemented in the IPv6 routers; the MR will also need to tunnel all user uplink traffic that are destined to the Internet back to the MRHA. The MRHA will decapsulate these packets and forwards them to the appropriate CNs in the Internet. This process is known as *bi-directional tunneling* in the NEMO research community.

The MONAMI6 IETF working group [37] is focused on extending the NEMO Basic Support Protocol with multi-home support. The group investigates the technical issues and challenges for the NEMO multi-homed architecture. The research presented in this thesis compliments the research performed in the MONAMI6 working group, as we are focused on designing traffic engineering algorithms that are suitable for the NEMO multi-homed network architecture (i.e. controlling how to distribute

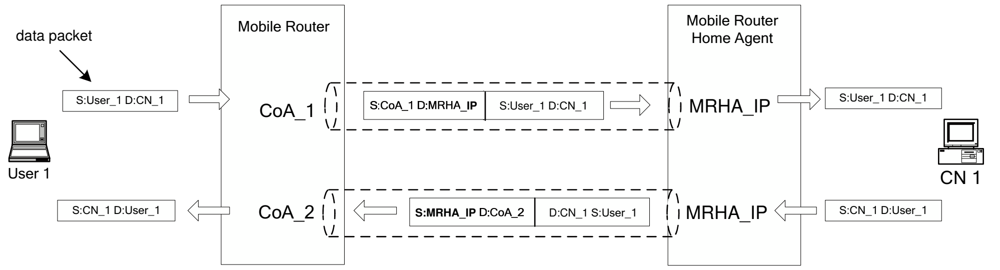

Figure 3.2: NEMO multi-homed bi-directional packet tunneling

the data traffic across the multiple access links). Our proposed traffic engineering solutions cannot be realized without the appropriate underlying mechanisms to support this architecture.

In the proposed NEMO multi-homed architecture; as shown in Figure 3.1; the data traffic between the mobile network and the Internet are routed through the multiple bi-directional IP-to-IP tunnels established between the wireless network links on the MR and the MRHA. For the uplink data traffic, the MR tunnels each packet to the MRHA by encapsulating the packets with any one of the MR wireless access networks current CoA as the source address, and the MRHA's IP address as the destination address. The packet will be forwarded to the MRHA over the selected network interface. When the MRHA receives the packet, it will decapsulate the packet and forwards the packet to the destination specified in the original packet. For downlink traffic, the packet will be first routed to the MRHA since MRHA is responsible for handling the packets that are destined the subnet of the mobile network. The MRHA encapsulates the packet using one of the MR's CoAs as the destination address, and the MRHA's IP address as the source address of the packet. The packet will then be routed to the MR via the wireless access network in the CoA is mapped to. Once the

MR receives the packet, it decapsulates the packet and forwards it to the passenger over the on-board LAN. An illustration of the bi-directional tunneling process in the NEMO multi-homed system is shown in Figure 3.2.

3.3 Traffic engineering opportunities

The traffic engineering problem in the NEMO multi-homed architecture is focused on controlling which wireless access links the users' data packets will be forwarded to. The control over this first or last hop wireless link, is crucial for the QoS in the end-to-end path [38] since these wireless access links are bandwidth-limited and are more error prone than the fixed wired communication links.

The NEMO multi-homed architecture provides a large degree of flexibility of traffic engineering control since the passengers' uplink and downlink data flows can be dynamically routed through any one of the bi-directional tunnels, without affecting the delivery of the packets to the end hosts. Consequently, the NEMO multi-homed architecture presents the following traffic engineering opportunities:

- *Traffic splitting* - The bi-directional tunneling of the users' data traffic allows the multi-homed system to perform different types of traffic splitting; i.e. distributing the packets from a user or data flow among the different access links; without breaking the users' end-to-end flow semantics. This opens up the opportunity for the system to increase the overall link utilization by distributing the data traffic among the wireless access links.

- *Traffic re-scheduling* - The bi-directional tunneling of the users data traffic also opens up the opportunity for the system to dynamically *re-schedule* the data traffic of a user or flow, from one access link to another without affecting the correct delivery of the data packets to the end users. This allows the system to perform traffic engineering that adapts to the dynamic nature of the wireless access links, by re-scheduling the data traffic among the different access links to improve performance or for resilience purposes.

- *Downlink traffic control* - Since the MRHA is responsible for tunneling the users' downlink data traffic to the MR, the NEMO multi-home architecture opens up the opportunity for the MRHA to exhibit traffic engineering control over the users' downlink data traffic. As we will see in Chapter 7, a signaling protocol between the MR and MRHA allows the MR to realize downlink traffic control. This feature allows the system to perform independent uplink and downlink traffic engineering control over the uses data traffic.

- *Repetitive mobility patterns* - In the case where the mobile network system is deployed in PTVs; where the vehicles repetitively traverses fixed routes; the mobility pattern of such system are more predictable than mobile users. Hence, it is likely that the physical characteristics of the access links (i.e. signal quality, interference) should exhibit some sort of correlation between each trip [39]. This opens up the opportunity for such systems to take advantage of this, by utilizing historical data recorded in past trips to assist the system in making predictions on link characteristics (i.e. the expected bandwidth in different locations at different times of the day). The predictions allow the system to be more proactive in its traffic engineering control.

The traffic engineering opportunities provides us with valuable guidance on the design of traffic engineering solutions for the NEMO multi-homed model. As discussed

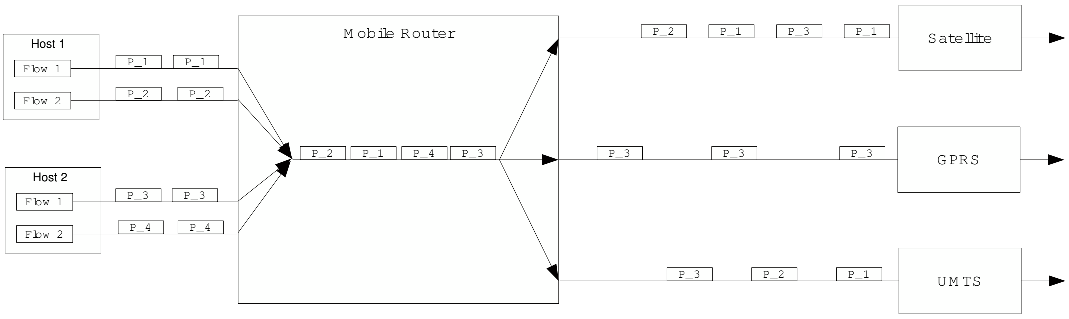

Figure 3.3: Packet-based traffic engineering scheme

above, one of the traffic engineering opportunities is the ability to perform different types of traffic splitting among the wireless access links which are connected to the MR. This is an important traffic engineering design issue for the NEMO multi-homed model; since, as we will discover in this thesis, selecting an unsuitable traffic splitting will severely degrade the throughput performance of the users data flows. Therefore, we have identified three types of *traffic engineering schemes* that can be used in the NEMO multi-homing architecture, and structured our research accordingly. These are namely: the *user-based*; *flow-based*; and *packet-based* traffic engineering schemes, which are similar to the traffic splitting schemes as discussed in [40]. The traffic engineering schemes basically represents the level in which the users data traffic can be split among the multiple network interfaces. The difference between these schemes lies in the granularity of the traffic engineering control that can be exhibited by the MR and MRHA. We will now discuss each of these schemes in detail.

3.3.1 Packet-based traffic engineering scheme

The packet-based traffic engineering scheme; as illustrated in Figure 3.3; represents the finest level of data switching control for the NEMO multi-homed architecture.

This is a flow-splitting scheme, since each packet are routed independently and therefore packets from the same data flow can be distributed over multiple access networks. The packet-based traffic engineering scheme will not affect the end-to-end flow semantics, since the bi-directional tunneling of the data packets will not change the source or destination IP address of the data packets.

The main advantage of the packet-based scheme is that it offers the finest level of traffic engineering control for the routers. This allows the MR or MRHA to make more efficient use of the multiple access links by treating the user data traffic as a single packet stream and evenly distributing these packets among the access links. However, there are several limitations with this approach. Firstly, switching a users' traffic on a per-packet basis significantly increases the risk of out-of-order packet arrival at the end users, which occur when there are performance disparity among the access links. The out-of-order packet arrivals are undesirable since it degrades the throughput performance of TCP flows and affects the quality of real time applications. Secondly, if the system tries to minimize the out-of-order packet arrivals using techniques such as those discussed in [22, 23, 30], the router may need to maintain precise state information of each access network, which may not be easily accessible in the mobile network due its constant movement while in transit.

Unlike the user-based and flow-based traffic engineering schemes, the packet-based traffic engineering scheme is only suitable for controlling the users' uplink data traffic. This is due to the fact that the MR is directly linked to the wireless access networks, which makes it easier to schedule the users' uplink packets across these links. As most packet-based algorithms depends heavily on precise timing to mask out the effects of out-of-order packet arrivals at the end hosts, the MRHA may not be able to

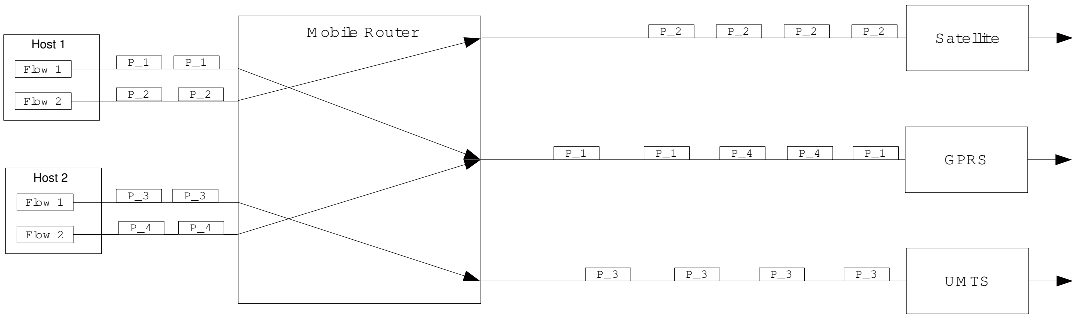

Figure 3.4: Flow-based traffic engineering scheme

exhibit precise scheduling control on users' downlink data traffic since the packets can potentially traverse vastly different paths before being scheduled onto the different wireless access networks. As a result, the downlink packets may not be scheduled over the wireless access links as intended by the MRHA.

In the packet-based traffic engineering scheme, the MR performs traffic engineering control on a per-packet basis. Therefore, the multi-homed MR needs to make instantaneous packet switching decisions. Depending on the different packet scheduling algorithms, the MR or MRHA may need to maintain various types of packet distribution tables.

3.3.2 Flow-based traffic engineering scheme

In the flow-based traffic engineering scheme, the users' data traffic is scheduled over the wireless access links on a per-flow basis. A flow is identified by a combination of the source IP address, destination IP address, source port number, and the destination port number. Therefore, the flow-based switching scheme involves mapping these unique $flowIDs$ to either one of the wireless access networks. Under this scheme, each data flow can be independently routed over different interfaces, i.e. a single user

Source IP	Source port	Destination IP	Destination port	Access link
x.x.x.x	3456	a.a.a.a	10000	Vodafone UMTS
x.x.x.y	7006	b.b.b.b	12000	Satellite
x.x.x.z	10000	c.c.c.c	13000	Vodafone GPRS

Table 3.1: Sample flow switching table

can have multiple concurrent active flows sent over different interfaces. Figure 3.4 illustrates the flow-based traffic engineering scheme.

The advantage with the flow-based traffic engineering scheme is that it offers a medium level of switching control that lies between the packet-based and user-based traffic engineering scheme. This allows the router to have greater flexibility in controlling how the data traffic is distributed over the multiple wireless access networks. It is also relatively simple to implement, as the switching only requires the additional lookup of the packets' IP addresses and port numbers to identify the flow. Similar to the user-based traffic engineering scheme, there is a minimal probability that packets from the same flow will arrive out-of-order since the packets from the same flow are always mapped to a single link. The main disadvantage with the flow-base scheme is that it does not represent the finest level of control, which may result in sub-optimal switching performance. For example, if there is only a single flow in a multi-homed system with 3 access links, a flow-based traffic engineering scheme will under-utilized the system aggregate bandwidth since the scheme will only map the flow to either one of the links in the system.

Similar to the user-based scheme, the flow-based scheme can be used to control both the uplink and downlink traffic, by implementing a *flow switching table* similar to the one shown in Table.3.1 in the MR and MRHA respectively. For all uplink packets, the MR will need to lookup the $flowID$ in the flow switching table to determine which access network to tunnel that packet over. Similarly, the MRHA will need to perform a lookup on its own flow switching table before tunneling the downlink packets to the on-board passengers.

In the flow-based scheme, the traffic engineering problem is basically to determine which flow will be mapped to which network. The MR and MRHA needs to be able to detect the arrival of new flows, and determine which access link the flow traffic should be mapped to.

3.3.3 User-based traffic engineering scheme

The user-based scheme represents the simplest traffic engineering scheme of all. It is a coarse-grain approach, where traffic engineering is performed on a per-user basis. A user is identified by their on-board IP address and therefore the user-based switching scheme only requires mapping the user IP to one of the wireless access networks. This basically means that all flows that belong to a single user will be routed over the same wireless access network, and any change in the traffic engineering will result in transferring all flows belonging to the user from the original network over to the new network. Figure 3.5 illustrates the user-based traffic engineering scheme.

The main benefit of using the user-based traffic engineering scheme is that it is relatively simple to implement. The traffic engineering control is performed on the aggregated requirements of each user. The MR does not need to maintain any per-flow or per-link states, and the switching only requires the router to lookup the user IP address of each packet. Also, as the packets from the same users are mapped to the same link, the packets in each data flow of the users will consequently traverse the same link which reduces the out-of-order packet arrivals at the end hosts. The

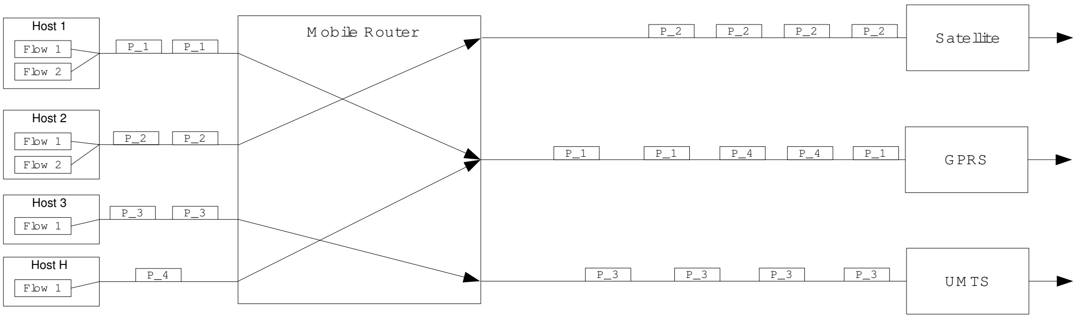

Figure 3.5: User-based traffic engineering scheme

On-board host IP	Access link
x.x.x.x	Vodafone UMTS
x.x.x.y	Satellite
x.x.x.z	Vodafone GPRS

Table 3.2: Sample user switching table

disadvantage with the user-based scheme is that the switching performance may be sub-optimal, due to its coarse grain switching nature.

The user-based traffic engineering scheme may be used to control the data traffic in both the uplink and downlink direction. To implement the user-based traffic engineering scheme in both the uplink and downlink direction, a sample *user switching table* shown in Table.3.2 will need to be implemented in the MR and MRHA respectively. For all incoming uplink packets, the MR will look at the source IP address of the packet and tunnel the packet over the corresponding wireless access network as indicated in the table. In contrast, the MRHA will make a lookup on the destination IP address for all incoming downlink packets, and tunnel the packets to the MR accordingly.

3.4 Issues and challenges

In Chapter 2, we classified the research on multi-homing into three main areas: static multi-homed networks; mobile multi-homed hosts; and mobile multi-homed networks. To highlight the novelty of our research model, the NEMO multi-homing architecture has the following characteristics:

- *Dynamic link characteristics* - The wireless nature of the access links, and the fact that the entire system is on the move means that the link characteristics (i.e. bandwidth and delay) are more dynamic than the static multi-homed networks. It is therefore a challenge for the mobile multi-homed network to mask out these variations in order to provide a more stable and higher bandwidth link to the on-board users.

- *Large data traffic* - The amount of data traffic transferred over the mobile multi-homed network are likely to be significantly more than a single mobile multi-homed user. Therefore, efficient traffic engineering over these dynamic and bandwidth-limited wireless links is a crucial design goal for the MR. With a multi-user system, the traffic engineering design may also need to be aware of additional switching criteria such as providing a fair bandwidth allocation to the on-board users, and performing user admission control to limit the number of users in the system.

A detailed classification on the various types of multi-homed systems is shown in Table.3.3. The challenge with traffic engineering is to maximize the performance metrics for both the mobile network operator and the on-board passengers. For example, the operator may want to maximize the system link utilization or its profit

Multi-homed type	User traffic	Access links	Mobility
static networks	high - multiple users	wired static links	no mobility
mobile hosts	low - single user	wireless dynamic links	random
mobile networks	high - multiple users	wireless dynamic links	repetitive

Table 3.3: Characteristics of different multi-homed models

under a common charging model, while the users may want high per-flow fairness or bandwidth guarantee. This is a highly challenging problem since there are potential trade-offs between the operator and user objectives. In this thesis, we will investigate design traffic engineering solutions that aim to maximize these operator and user objectives for the NEMO multi-homed architecture.

3.5 Summary

In this chapter, we provided an in-depth overview on the NEMO multi-homed architecture; where we focused our discussion on the bi-directional tunneling of the data traffic between the on-board users and their corresponding nodes in the Internet. We have identified that the ability to perform traffic splitting, traffic re-scheduling and downlink traffic control are the new traffic engineering opportunities for the NEMO multi-homed architecture. Also, we have identified the three traffic engineering schemes that can be used in the NEMO multi-homed architecture; namely the packet-based, flow-based and user-based traffic engineering schemes; and provided a thorough comparison on the performance and implementation issues associated with each of these schems.

To motivate our research, we highlighted the new traffic engineering issues and challenges that are associated with the NEMO multi-homed architecture, and compare the differences between the NEMO multi-homed network model with the other previously proposed multi-homed models that were discussed in Chapter 2. Our in-depth comparison shows that the NEMO multi-homed system faces the challenge of switching large amount of on-board users' data traffic over a set of highly dynamic wireless access links.

Chapter 4

Packet-based traffic engineering

In this chapter, we investigate the feasibility of implementing packet-based traffic engineering solutions in the mobile multi-homed network architecture. Our results show that the flow-splitting nature of the packet-based traffic engineering significantly degrades the throughput performance of users' TCP data flows, hence making it unsuitable for the mobile network architecture.

4.1 Introduction

The packet-based traffic engineering scheme offers the finest switching granularity among the various traffic engineering schemes that can be implemented in the NEMO multi-homed architecture. However, as discussed in Section 3.3, the packet-based traffic engineering scheme may introduce the packet re-ordering problem, which affects the performance of the data flows. In this chapter, the aim is to investigate the performance of packet-based traffic engineering in the mobile multi-homed network model. In particular, we discuss some popular packet scheduling algorithms, and examine whether these algorithms are suitable for the mobile multi-homed network model. Furthermore, as the packet scheduling algorithms are flow-splitting approaches, we compare the performance of packet scheduling algorithms against some relevant flow scheduling algorithms to examine the effects of flow-splitting in a simulated mobile multi-homed network model.

The rest of this chapter is organized as follows. In Section 4.2, we look at the design of some popular packet scheduling algorithms that were proposed in the past. In Section 4.3, we discuss the details of the simulation model and the experiment scenarios will be used to model the dynamic behavior of the wireless access links. In Section 4.4, we present a thorough simulation analysis on the performance of the various packet and flow schedulers discussed, to investigate the effects of packets switching in the context of mobile multi-homed networks. In Section 4.5, we conclude this chapter with a summary of its contributions.

4.2 Packet scheduling algorithms

In this section, we present the design of two popular packet scheduling algorithms; namely the Round Robin (RR) and Weighted Round Robin (WRR) packet scheduling algorithms. These algorithms were selected for our simulation analysis since they have been heavily studied in the context of packet queuing and packet scheduling [41–44]. The aim of these packet scheduling algorithms is to balance the traffic load among the multiple access links. We did not consider other more advanced packet scheduling algorithms; for example the weighted fair queuing, deficit round robin, and their many other variants; since the aim of this analysis is to investigate the effects of flow-splitting in the context of mobile multi-homed networks. Therefore, we instead implemented the corresponding RR and WRR flow scheduling algorithms in our simulation study.

4.2.1 Round Robin

The RR packet and flow scheduler schedules every new incoming packet and flow to each link in a round robin fashion. The aim of these scheduling algorithms is to evenly distribute the user traffic among the multiple access links, and to ensure that all the access links are utilized by the multi-homed system.

For the RR packet scheduler implementation, the MR will first be given a list of all the access links in the multi-homed system. For each incoming packet, the MR will schedule the packet to the first link in the list. The MR will then rotate the list of access links to ensure that packets are scheduled to the other links before they are scheduled to the same link again.

In the RR flow scheduler implementation, the MR will again be given a list of all the access links in the multi-homed system. For each incoming packet, it first needs to detect whether the packet belongs to a new or existing flow. If the packet belongs to a new flow, the MR will select the first link in the list, and create a new entry in the flow switching table. All subsequent packets from this flow will then be scheduled according to the link mapped to this flow. For every new flow, the MR will rotate the list of access links in a round-robin manner to ensure that flows are scheduled to the other access links before they are scheduled to the same link again.

4.2.2 Weighted Round Robin

The main limitation with RR scheduling is that it does not consider the link bandwidth disparity in its scheduling decision. This may result in both the under-utilization and over-utilization of the different access links in the system. To deal with this issue, we include the more advanced WRR flow and packet scheduler in our study. The

WRR scheduler is an extension to the RR scheduler, where it utilizes the links' bandwidth information in its scheduling decision. Therefore, instead of blindly scheduling every new packet or flow to each access link in a round robin manner, the WRR scheduler assigns weights to each link and forwards the data onto each link in proportion to the weights assigned. For example; if we have a multi-homed system with 3 links (10Mb/s, 5Mb/s, 1Mb/s); the WRR scheduler will assign the first 10 packets/flows to the first link, the next 5 packets/flows to the second link, and the next packet/flow to the third link. The scheduling of subsequent packets or flows will repeat this distribution pattern.

The implementation of the WRR packet and flow schedulers are similar to the corresponding RR packet and flow schedulers. In the WRR packet and flow schedulers, the MR needs to maintain a count on the number of packets/ flows that have been sent on the current link (in a list of access links). The MR will only move on to the next access link when it has sent the required number of packets or flows to the current link. The required number of packets or flows scheduled to an access link is determined by the weights assigned to each link.

4.3 Simulation model

4.3.1 Simulation architecture

To examine the performance of the packet and flow schedulers that were discussed in the previous section, we performed a set of simulations under a modified version of Network Simulator Version 2 (NS2). NS2 is a discrete event simulator which is widely used by the research community to evaluate network performance. We selected NS2 as our simulation platform because it is an open-sourced simulator,

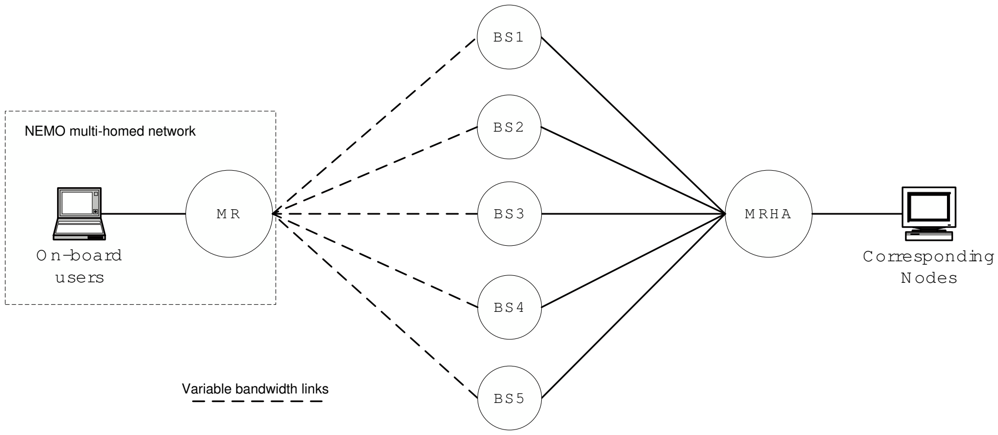

Figure 4.1: NS2 simulation architecture for the NEMO multi-homed model

which allows us to modify the simulator according to our needs. Also, the simulator contains an accurate implementation of TCP, and built-in support for tracing various important performance metrics such as the TCP sender congestion window size, TCP throughput, TCP re-transmissions etc. This allow us to measure the TCP throughput performance of the MR and the end-to-end users in our simulations.

The simulation architecture used in this simulation analysis is shown in Figure 4.1. In this simulation architecture, the on-board users and the MR represents the mobile multi-homed network. The MR is connected to the Internet via the multiple wireless access links, which are represented by the dotted lines between the MR and the various Base Station (BS)s. As we are only concerned with varying the link bandwidth and propagation delays to simulate the dynamic link characteristics, we did not implement any wireless link or physical layer models in the access links. The link dynamics are simulated by varying the link bandwidths and propagation delays during the simulation. The base-stations, MRHA, and the corresponding nodes are all assumed to be located in the Internet and hence they are all connected via links

Parameter	Value
TCP Sender	TCP Newreno
TCP Segment Size	1000 bytes
Maximum TCP Window Size	1000 segments
TCP Receiver	TCPSink
Application layer protocol	FTP
File size	50Mb

Table 4.1: Static NS2 simulation parameters.

with high bandwidth and low delay settings. For simplicity, we did not introduce any other external data traffic in the simulation model. The static simulation parameters used in the NS2 simulations are listed in Table 4.1. Note that the maximum TCP window size value has been set to a relatively high value (of 1000 segments), so that it is bounded by the end-to-end bandwidth delay product rather than the default value (of 20 segments) in the standard NS2 implementation. For the implementation details of our NS2 simulation model, please refer to Appendix A.

4.3.2 Experimental scenarios

Identical links case

We start off our simulation study by looking at the identical links case where there are no link disparities in the system; i.e., all the links have identical bandwidths and propagation delays. The access link parameters used in this simulation scenario are shown in Table 4.2.

The identical links case allows us to measure the performance of the packet and flow schedulers under the absence of link disparity and bandwidth variations. The absence of link disparity means that we expect the packets to arrive in-order at the end hosts, even if the packets are scheduled onto different access links.

Base-station	Bandwidth	Propagation delay
BS1	5Mb/s	20ms
BS2	5Mb/s	20ms
BS3	5Mb/s	20ms
BS4	5Mb/s	20ms
BS5	5Mb/s	20ms

Table 4.2: Link parameters for simulation with identical access links.

To see whether the presence of multiple flows affects the system and flow performance of the schedulers, we performed two sets of experiments under this case. Firstly, we look at the performance of a single flow scenario, where we initiated a 50Mb file transfer over the system between the on-board user and the corresponding node. Secondly, we repeated the first set of simulations with the exception that there are 10 file transfers instead of one.

In the identical links case simulations, we only simulate the performance of the RR packet and flow scheduling algorithms since there are no bandwidth disparity to differentiate the link weights for the WRR packet and flow schedulers. Since the bandwidth values are all identical (i.e. all the link weights are equal to 1), the WRR packet and flow schedulers will therefore schedule all the packets/flows in the exact same way as their corresponding RR schedulers.

Delay disparity case

In the delay disparity case, we introduce propagation delay disparity among the access links. The access link parameters used in this scenario are shown in Table 4.3.

For the packet schedulers, we expect that there may be out-of-order packet arrivals in the delay disparity case since the time it takes for the packets to reach the end

Base-station	Bandwidth	Propagation delay
BS1	5Mb/s	20ms
BS2	5Mb/s	30ms
BS3	5Mb/s	40ms
BS4	5Mb/s	50ms
BS5	5Mb/s	60ms

Table 4.3: Link parameters for simulation with link propagation delay disparity.

hosts will be different for each link. This is because the packet scheduling algorithms performs flow-splitting, where packets from the same flow are scheduled over different wireless access links. In contrast, we predict that the packet re-ordering problem will not occur in the flow schedulers, since packets in the same flow will follow the same end-to-end path. The only effect we may see on the flow schedulers is that some flows may suffer degraded throughput performance when mapped to a higher delay link, since the TCP throughput performance is determined by the product of the bandwidth and delay values of the access link.

In the simulation of the delay disparity case, we will only simulate the single and multiple flows scenarios for the RR packet and flow scheduling algorithms. We expect that the flow transfer time of the RR flow scheduler will be the same as the corresponding simulation scenario of the RR flow scheduler in the identical link case, since the data flow is mapped to the same link parameters in both simulations. Again, we will not simulate the performance of the WRR packet and flow scheduling algorithms since the weights will be the identical in the access links.

Base-station	Bandwidth	Propagation delay
BS1	1Mb/s	20ms
BS2	2Mb/s	20ms
BS3	3Mb/s	20ms
BS4	4Mb/s	20ms
BS5	5Mb/s	20ms

Table 4.4: Link parameters for simulation with link bandwidth disparity.

Bandwidth disparity case

In the bandwidth disparity case, we investigate the effects of bandwidth disparity among the access links. The access link parameters used in this scenario are shown in Table 4.4.

Similar to the delay disparity case, we expect that the packets scheduled by the packet scheduling algorithms in the bandwidth disparity case may experience packet re-ordering problems at the end hosts. This is because the bandwidth difference between the access links means packets scheduled on these links will have different transmission rates. For the flow schedulers, we expect that the flow transfer times will be different for flows mapped to the different links, due to the bandwidth disparity among the access links.

In the single flow scenario, we will only simulate the performance of the RR packet-scheduler, RR flow scheduler, and the WRR packet scheduling algorithms. The WRR flow scheduler will not be included in the single flow scenario since its flow assignment will be identical to the RR flow scheduler. For the multiple flows scenario, we will simulate the performance of the RR and WRR packet and flow schedulers since the bandwidth disparity among the links allows us to assign different link weights to the access links; hence, the RR and WRR flow schedulers will make different flow scheduling decisions.

Bandwidth variations case

In the previous sets of simulations, we looked at the ideal case where the links did not exhibit any bandwidth variations. This is not an accurate representation of the mobile network model since the access links will exhibit bandwidth variations when the system is in transit. Hence, we introduced bandwidth variations among the links to see how both schedulers will perform under these variations. To simulate bandwidth variations, we varied the bandwidth of the links at fixed time intervals, where the size of the interval depends on the number of variations per 1000 simulation seconds (s). For example, if there are 100 bandwidth variations per 1000s in a particular simulation scenario, it means that there is a bandwidth variation in every 10s (from the start of the simulation). At the bandwidth variation times, the bandwidths variations among the links followed a uniform random distribution, where the values take on some fixed-size multiples (i.e. 0.5Mb/s) between zero and the maximum bandwidth of the links.

4.4 Simulation results and analysis

Identical links case

In the single flow case for the identical links scenario, the recorded file transfer time for the RR packet scheduler and the RR flow scheduler were 49.455s and 89.537s respectively. The significant difference between the transfer times shows that the RR packet scheduler is able to make efficient use of the multiple link bandwidths, since the packet scheduler was able to evenly distribute the packets among the access links without experiencing packet re-ordering problems. In contrast, the flow scheduler experienced a much higher transfer time since it was only able to utilize a single link in the system.

To gain a deeper understanding on the behavior of the TCP flows, we plotted the Sender Congestion Window (CWD) trace for the RR packet-scheduler and RR flow-scheduler in Figure 4.2 and Figure 4.3 respectively. Since there were no bandwidth variations introduced in these simulations, the CWD converged to a stable state in both cases. As the RR packet scheduler was able to utilize the bandwidth of the other links, it was able to converge to a higher CWD than the RR flow scheduler.

Next, we examined the effects of transferring multiple flows over both schedulers. We repeated the same simulations as previously; but with 10 FTP flows instead of 1. In this scenario, the RR flow-based scheduler was able to schedule the 10 flows evenly among the links. Hence, similar to the RR packet scheduler, the RR flow scheduler was able to make use of the aggregate bandwidth of the links.

We measured the mean flow duration for the schedulers, and the values were 164.178s and 168.059s for the RR packet scheduler and the RR flow scheduler respectively. It was evident that the performance difference between the two RR scheduling scheme was significantly reduced, due to the fact that the RR flow scheduler was able to make use of the aggregate bandwidth of the links. We also plotted the CWD trace for one of the flows in the RR packet scheduler and the RR flow scheduler in Figure 4.4 and Figure 4.5 respectively. Looking at Figure 4.4, we can see that the introduction of other flows in the system has caused a certain degree of instability in the CWD trace for the TCP flow in the RR packet scheduler. This instability is caused by the minor scheduling time variations of the packets, due to the presence of other competing flows in the system. In contrast, the same reference flow in the RR

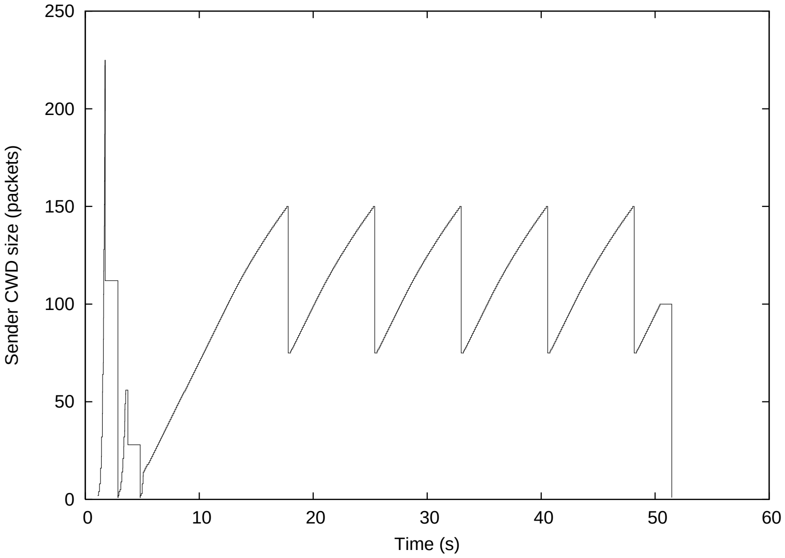

Figure 4.2: The sender congestion window trace for the RR packet scheduler in the identical link case with single flow.

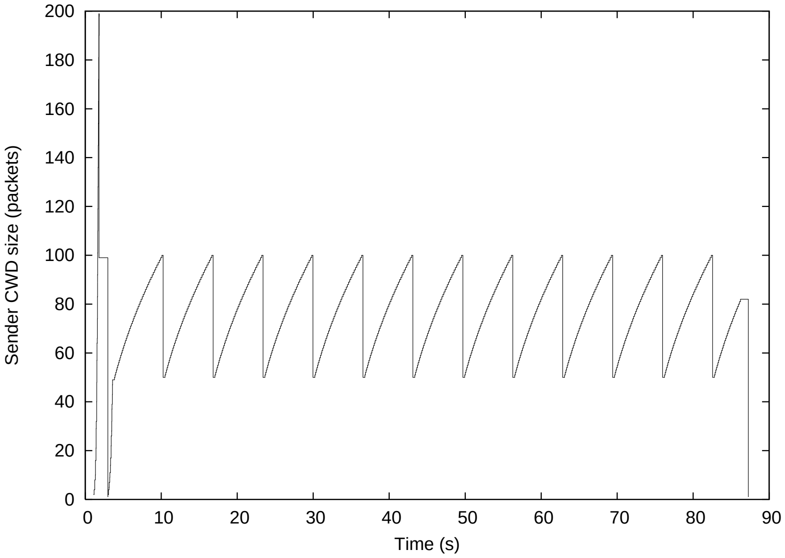

Figure 4.3: The sender congestion window trace for the RR flow scheduler in the identical link case with single flow.

flow scheduler did not exhibit this instability. This illustrates that the performance of the RR packet scheduler is sensitive to the presence of multiple flows.

Delay disparity case

In the single flow case for the link delay disparity scenario, the recorded file transfer times for the RR packet scheduler and the RR flow scheduler are 1668.640s and 89.537s respectively. This is in strong contrast to the identical links case, as the performance of the RR packet scheduler has been significantly degraded as a result of the delay disparities among the access links. Looking at the CWD trace for the RR packet scheduler, as shown in Figure 4.6, we can see that the TCP flow was not able to increase its CWD window to the levels reached by the corresponding flow in the identical links case (Figure 4.2). The reason why the CWD was capped at such a low range is because of the large number of packet re-transmissions in the RR packet scheduler, which recorded 6124 packet re-transmissions compared to the 271 packet re-transmissions recorded for the RR flow scheduler. As each packet are traversing paths with different delay characteristics, the significant number of packet re-transmissions is likely to be caused by the out-of-order packet arrivals at the end hosts. The effect of this is that it severely limited the throughput performance of the TCP flow.

In the multiple flows scenario, the mean file transfer times recorded for the RR packet scheduler and the RR flow scheduler are 905.596s and 169.044s respectively. Interestingly, the performance of the RR packet scheduler improved over the single flow case. The CWD trace show in Figure 4.7 illustrates that the TCP flow in the RR packet scheduler was able to converge to a higher CWD range than the single flow scenario. Again, the total number of re-transmissions had a direct influence on

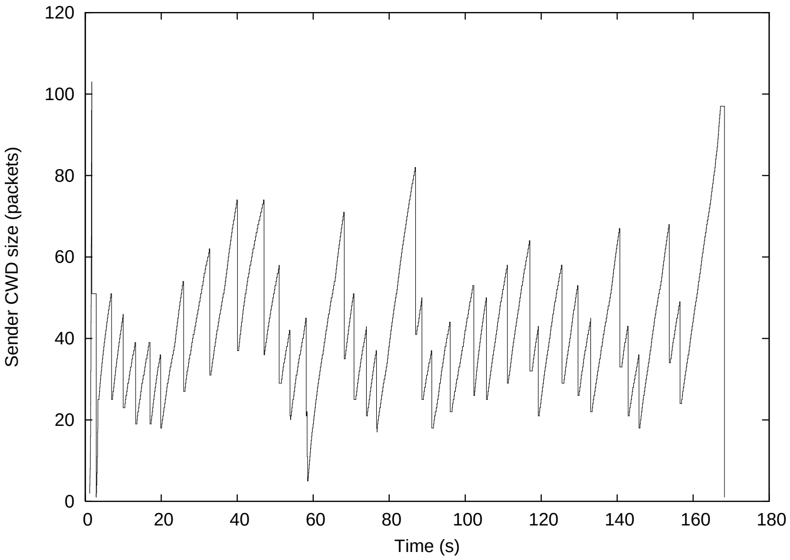

Figure 4.4: The sender congestion window trace for the RR packet scheduler in the identical link case with multiple flows.

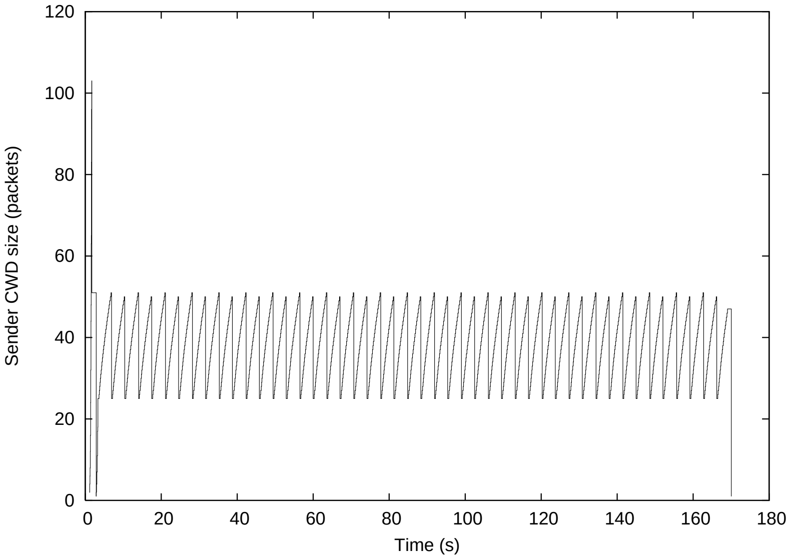

Figure 4.5: The sender congestion window trace for the RR flow scheduler in the identical link case with multiple flows.

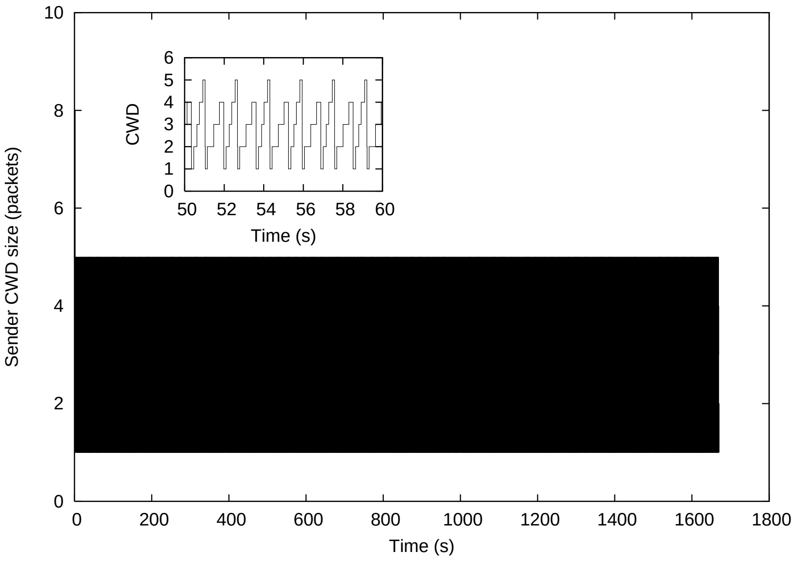

Figure 4.6: The sender congestion window trace for the RR packet scheduler in the link delay disparity case with single flow.

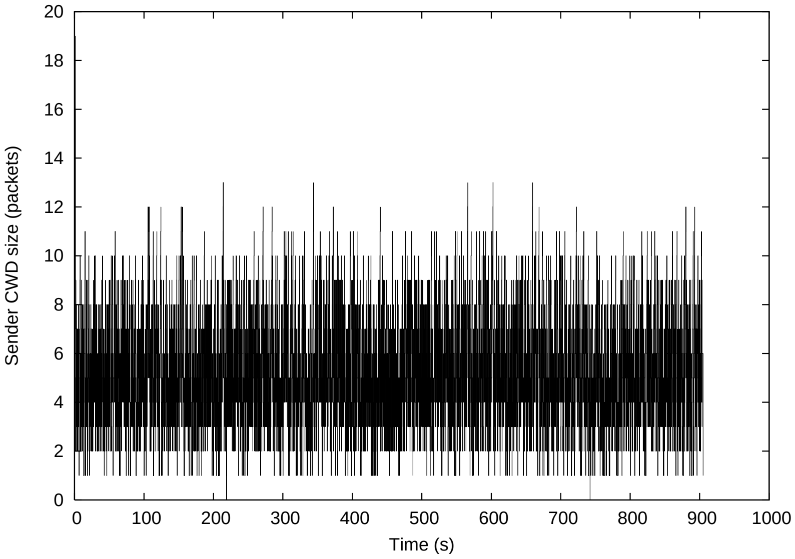

Figure 4.7: The sender congestion window trace for the RR packet scheduler in the link delay disparity case with multiple flows.

Scheduler	RR Packet	WRR Packet	RR Flow	WRR Flow
Mean flow duration	1403.799s	452.982s	569.428s	414.361s
Mean packet re-transmissions	6345.8	4154.73	226.3	212.8

Table 4.5: Mean flow duration and packet re-transmisions for simulation with link bandwidth disparity - multiple flows case.

the throughput performance of the flows, since each flow in the RR packet scheduler had an average of 5191.4 packet re-transmissions (compared to 6124 in the single flow case). The decrease in the number of packet re-transmissions is likely to be caused by the decrease in the number of out-of-order packet arrival at the end hosts, since the presence of multiple flows has increased the inter-packet scheduling time of packets belonging to the same flow.

Bandwidth disparity case

In the bandwidth disparity scenario, we added the WRR packet and flow schedulers to our simulation analysis. For the single flow case, the file transfer times for the RR packet scheduler, the WRR packet scheduler, and the RR flow scheduler were 1348.393s, 568.441s, and 85.259s respectively. By assigning the number of packets according to the link weights, the WRR packet scheduler was able to improve the throughput performance of the RR packet scheduler by more than 50%. In addition, even if the RR flow scheduler only utilized one access link, it was able to transfer the flow by over 6 times quicker than the WRR packet scheduler.

In the multiple flows case, we include the WRR flow scheduler in the simulation and the mean flow transfer times and the mean packet re-transmissions (per-flow) for the packet and flow schedulers are shown in Table 4.5.

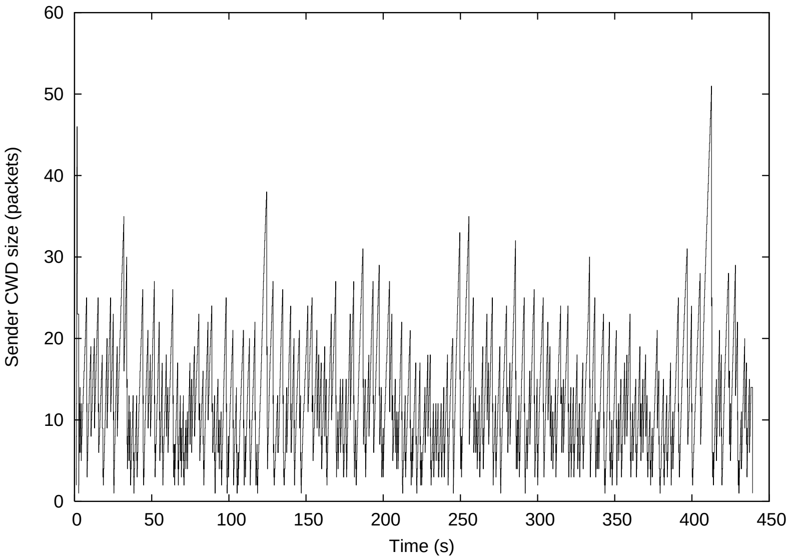

Figure 4.8: The sender congestion window trace for the WRR packet scheduler in the link bandwidth disparity case with multiple flows.

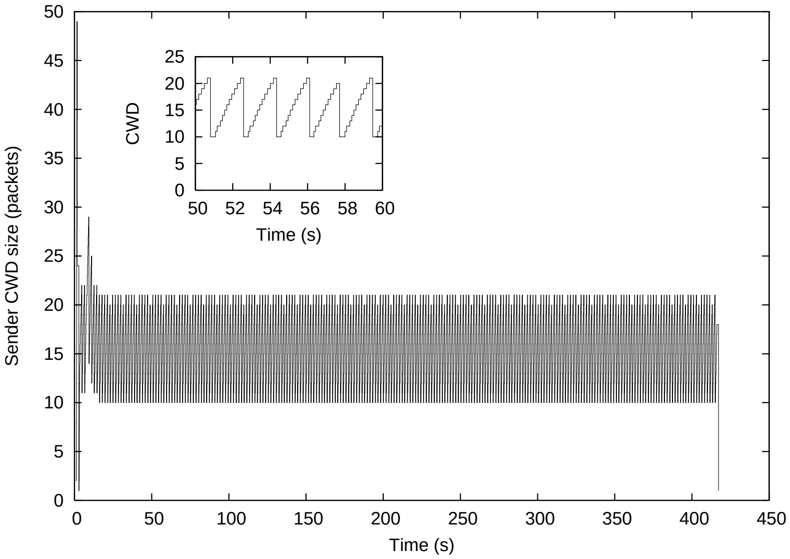

Figure 4.9: The sender congestion window trace for the WRR flow scheduler in the link bandwidth disparity case with multiple flows.

It is evident that the WRR packet and flow schedulers were able to achieve a higher throughput than their corresponding RR schedulers. Hence, we limit our discussion to the WRR schedulers as the characteristics of the RR schedulers were heavily discussed in the previous simulation scenarios.

The WRR flow scheduler was able to offer a lower mean flow duration than the WRR packet scheduler, since the WRR had more than an order of magnitude lower mean packet re-transmissions than the WRR packet scheduler. The CWD traces for one of the TCP flows in the WRR packet and flow scheduler are plotted in Figure 4.8 and Figure 4.9 respectively. Like other flow schedulers, the CWD in the WRR flow scheduler converged to a much steadier state than the WRR packet scheduler. As a result, the flow scheduler was able to achieve significantly lower packet re-transmissions and hence lower file transfer time than the packet scheduler. This illustrates the negative effects of flow-splitting in a multi-homed system with different bandwidth access links.

Bandwidth variations case

In this set of simulations, we introduced bandwidth variations among the access links. In the identical links case, we plotted the relationship of increasing the number of bandwidth variations and the mean flow duration in Figure 4.10 and Figure 4.11; which represented the single flow and 10 flows case respectively. In the single flow case, we can see that the RR packet scheduler was able to offer a lower flow transfer time in the absence of bandwidth variations. But as the number of bandwidth variations increases, the throughput performance of the RR packet scheduler deteriorated quite quickly. Once the number of bandwidth variations went beyond 20 variations per 1000s (i.e. a bandwidth variation scheduled every 50s), the RR flow scheduler was able to outperform the RR packet scheduler in the throughput performance by almost 50%. The throughput performance difference between the two scheduler increased as the number of bandwidth variations was increased.

For the multiple flows case, it is also evident that the RR flow scheduler was able to significantly outperform the RR packet scheduler in throughput performance as the number of bandwidth variations in the system was increased. Apart from the performance difference between the two schedulers, another interesting behavior is that the RR flow scheduler was able to offer a relatively stable throughput performance once the number of bandwidth variations was higher than a certain point (i.e. 30 variations per 1000s). This shows that the RR flow scheduler is much less sensitive to the number of bandwidth variations in the system than the packet schedulers.

We plotted the CWD trace of the RR packet and RR flow schedulers in the multiple flows scenario with maximum bandwidth variations (i.e. 100 variations per 1000s), in Figure 4.12 and Figure 4.13 respectively. The traces show how the throughput performance of the RR packet scheduler was significantly degraded in this simulation run. The bandwidth variations caused significant packet re-transmissions at the end-hosts, which limited the growth of the CWD in the TCP flows.

For the delay disparity case, we plot the number of bandwidth variations versus the mean flow duration time for the RR packet and flow schedulers in Figure 4.14. Similar to the results shown earlier, the RR packet scheduler offered inferior throughput performance even in the absence of link bandwidth variations. As we increased the number of bandwidth variations, the RR packet scheduler offered almost 8 times the file transfer times over the RR flow scheduler.

Finally, we plot the number of bandwidth variations versus the mean flow transfer

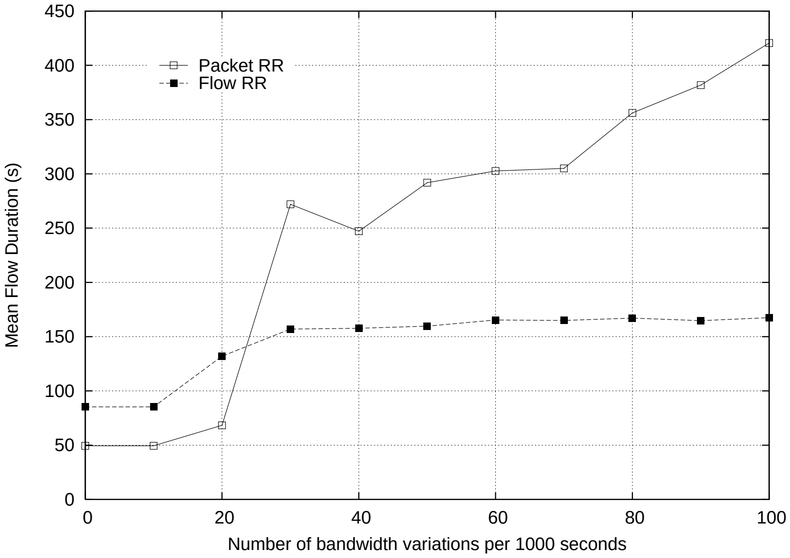

Figure 4.10: The number of bandwidth variations versus flow transmission time, in the identical link case with single flow.

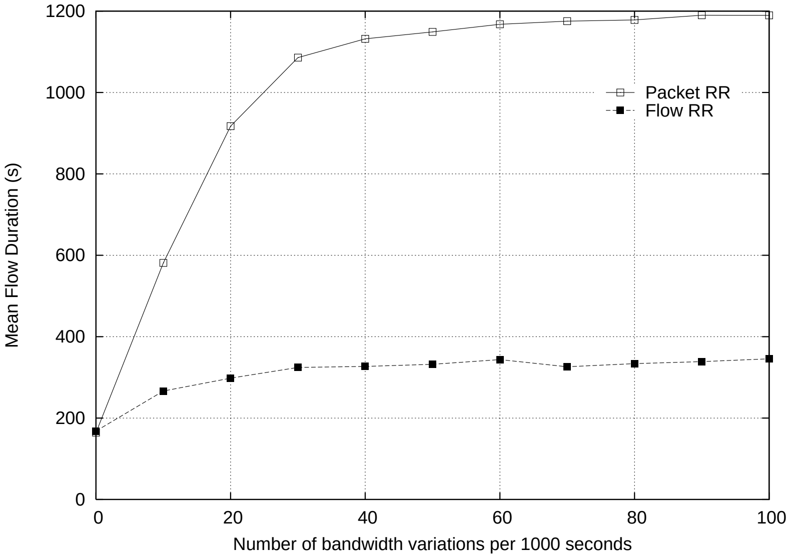

Figure 4.11: The number of bandwidth variations versus flow transmission time, in the identical link case with multiple flows.

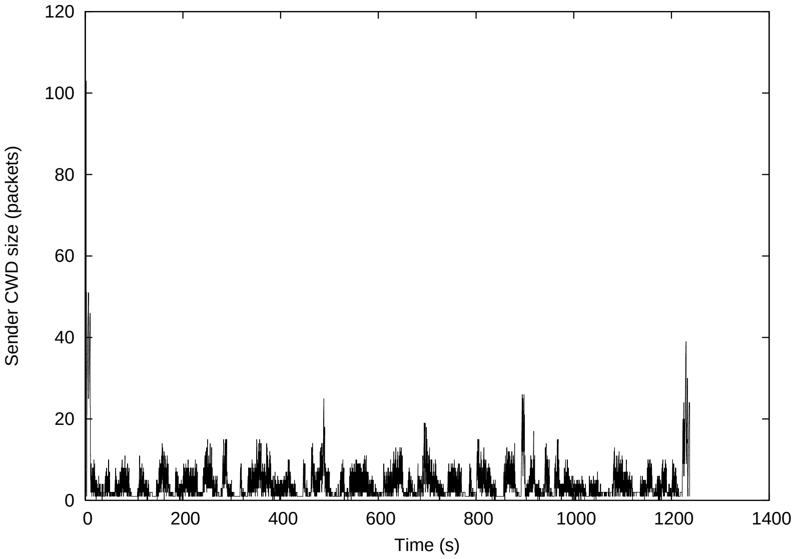

Figure 4.12: The sender congestion window trace for the RR packet scheduler, in the identical link case with multiple flows and 100 bandwidth variations per 1000s.

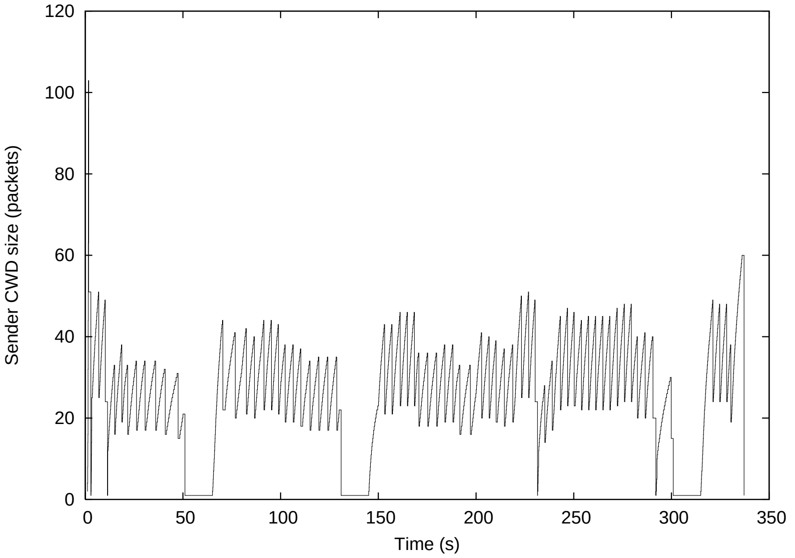

Figure 4.13: The sender congestion window trace for the RR flow scheduler, in the identical links case with multiple flows and 100 bandwidth variations per 1000s.

time for the schedulers in the bandwidth disparity case in Figure 4.15. In the absence of bandwidth variations, the WRR packet scheduler was able to provide a lower mean flow transfer time than the RR flow scheduler. However, once we introduce bandwidth variations, the flow schedulers were able to offer significantly lower flow transfer times than all the other packet schedulers.

4.5 Summary

In this chapter, we showed that the packet-based traffic engineering scheme is not suitable for the mobile multi-homed network architecture. This is due to the fact that the flow-splitting nature of packet-based switching causes significant number of packet re-transmissions; which degrades TCP performance, in situations where there are bandwidth and delay disparities in the wireless access links. Furthermore, the throughput performance of packet-based traffic engineering is further degraded as the number of link bandwidth variations increases.

Given the limitations of the packet-based switching scheme over the mobile multi-homed network architecture, we argue that other traffic engineering schemes that do not perform flow-splitting (e.g. the flow-based and user-based traffic engineering schemes) are more suitable for the mobile multi-homed network architecture. Even though the packet-based switching scheme has a finer switching granularity, we showed that even a simple round robin flow scheduler was able to provide better throughput performance than all the other packet scheduling algorithms discussed in this chapter.

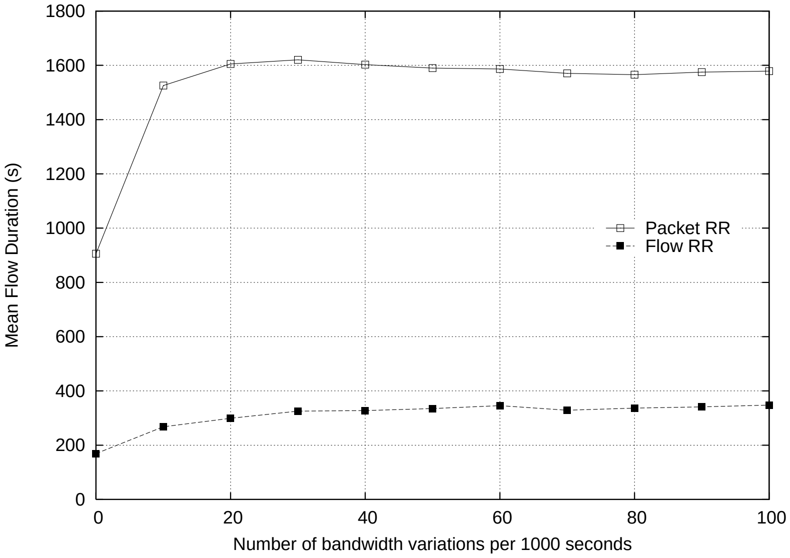

Figure 4.14: The number of bandwidth variations versus flow transmission time, in the link delay disparity case with multiple flows.

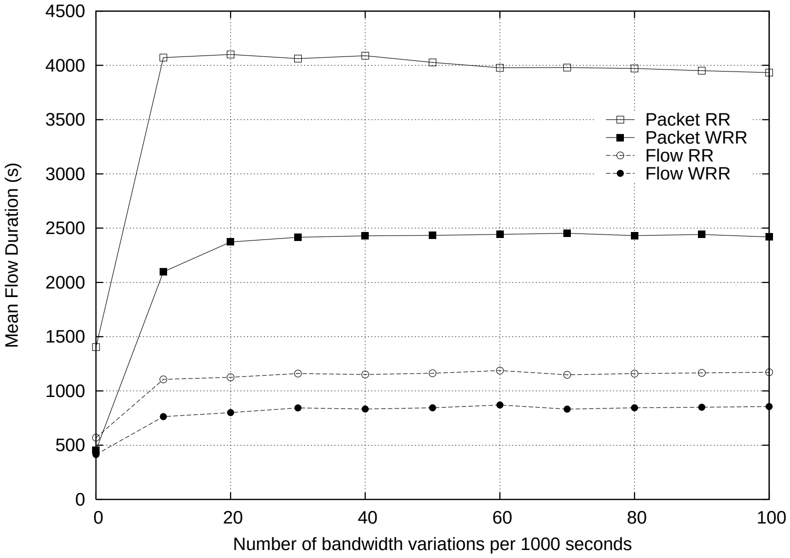

Figure 4.15: The number of bandwidth variations versus flow transmission time, in the link bandwidth disparity case with multiple flows.

Chapter 5

Maximum utility flow-based traffic engineering

In this chapter, we propose a new *MaxUtility* flow scheduler that aims to maximize both the system link utilization and flow fairness for the NEMO multi-homed system. Extensive simulations show that the proposed *MaxUtility* flow scheduler can provide better throughput performance and flow fairness than previously proposed flow schedulers.

5.1 Introduction

Selecting the correct data switching scheme for the NEMO multi-homed system is an important traffic engineering issue. As we saw in Chapter 4, the fine grain packet-based traffic engineering scheme is highly unsuitable for the mobile multi-homed systems, due to the constant packet re-transmissions at the end hosts that were caused by the fluctuating link characteristics in the wireless access links. In our simulation analysis, we showed that even a simple RR flow scheduler was able to out-perform all the packet scheduling algorithms that were discussed in Chapter 4.

The focus of this chapter is to design a flow scheduler that aims to maximize both the system link utilization and flow fairness in the NEMO multi-homed system. Maximizing the system link utilization is an important operator objective, since the system will need to transfer large amount of user data over the bandwidth-limited wireless access links. On the other hand, maximizing the flow fairness is an important user objective as users will want to gain a fair share of the system bandwidth for their data flows. As we will discover in this chapter, there are potential trade-offs between these objectives and hence optimizing both objectives simultaneously is a highly challenging problem.

The rest of this chapter is organized as follows. In Section 5.2, we formulate the optimization problem of maximizing both the system link utilization and flow fairness of the NEMO multi-homed system, by defining a new system utility index which combines both objectives quantitatively. In Section 5.3, we present the details of a new flow assignment algorithm that aims to maximize the system utility index. In Section 5.4, we propose a new flow scheduler that implements link switching, where the system re-schedules existing flows to different access links in order to maintain a high system utility index. In Section 5.5, we further extend the flow scheduler by utilizing link bandwidth predictions to assist the MR in making more proactive flow scheduling decisions. In Section 5.6, we perform an extensive simulation analysis on the proposed flow scheduler and discuss the significant findings in the simulation results. In Section 5.7, we conclude this chapter with a summary of its contributions.

5.2 Problem formulation

The role of the MR in the flow-based traffic engineering scheme is to determine which wireless access link the packets of each flow will be forwarded to, during the lifetime of the flow. In the previous chapter, we looked at the RR and WRR flow schedulers that aimed to statically balance the data traffic load among the access links. However, we argue that load-balancing solely concentrates on the distribution of the data traffic load and does not directly address important operator and user objectives, such as maximizing system utilization and maintaining flow fairness (respectively).

Therefore, we formulate the flow-based traffic engineering problem by defining two new flow scheduling objectives that directly address the operator and user requirements in the NEMO multi-homed model. Firstly, given that the wireless access links have scarce and highly dynamic bandwidth, the operator will want to maximize the utilization of the system's link bandwidth in order to service more users. Secondly, given that there are a potentially large number of users contending for the limited aggregate system link bandwidth, maintaining flow fairness in the system provides each flow with a fair share of the bandwidth resources. In contrast to previously proposed flow scheduling algorithms; which includes the RR flow scheduler [28] and the WRR flow scheduler that were discussed in Chapter 4; we directly address both the user and operator objectives in our flow-based traffic engineering design.

Before we formally define the two traffic engineering objectives, we first highlight the assumptions that were made in the design of the proposed flow scheduler. The flow scheduler was specifically design to schedule TCP flows only, since it had been widely acclaimed that TCP flows make up most of the current Internet traffic [45–48]. Focusing on the scheduling of TCP flows allowed us to make the following assumptions. Firstly, we assume that once a TCP flow is mapped to an access link, the Additive Increase Multiplicative Decrease (AIMD) nature of TCP ensures that the flow will eventually utilize all of the link's available bandwidth. Secondly, when multiple TCP flows are mapped to the same access link, it is assumed that the flows will eventually get a fair share of the link's available bandwidth due to the bandwidth-competing nature of the TCP congestion control mechanisms. With a fair scheduler in place, the assumption about TCP flows equally sharing the link bandwidth; which may not be true for flows with different RTTs; is not necessary.

In addition, the aim of this chapter is to design generic flow scheduling algorithms for the NEMO multi-homed model; hence, we do not make a distinction between scheduling uplink and downlink data flows. The algorithms presented in this chapter assume a generic system model where the flow scheduler is only concerned with scheduling the users' data flows over a set of bandwidth varying access links. If the NEMO multi-homed architecture implements the appropriate mechanisms to support downlink flow-based traffic engineering control (as we will see in Chapter 7), the flow scheduling algorithms presented in this chapter can be used to schedule both the on-board users' uplink and downlink data flows.

We now proceed to provide formal definitions of the flow scheduling objectives. The system link utilization measures how much of the system's aggregate bandwidth has been utilized by the MR. Following our first assumption on TCP flows; where it is assumed that a TCP flow can fully utilize the bandwidth of an access link; the system link utilization is determined by the set of access links that are currently utilized for data transfer in the MR. Hence, we define the system link utilization index as:

$$U = \frac{\displaystyle\sum_{j=1}^{N} z_j b_j}{\displaystyle\sum_{j=1}^{N} b_j} \tag{5.1}$$

subject to

$$z_j = \begin{cases} 1 \text{ if } n_j > 0 \\ 0 \text{ otherwise} \end{cases} \tag{5.2}$$

where U is the system link utilization index; z_j indicates whether a flow has been mapped to link j; N is the number of access links in the system; b_j is the available bandwidth in link j; and n_j is the number of flows mapped to link j.

The flow fairness measures how evenly the bandwidth resources are allocated to each data flow. To calculate the flow fairness in the system, we use a varied form of the highly popular Jain's fairness index [49]. The original form of the Jain's fairness index is defined as:

$$J = \frac{\left[\displaystyle\sum_{i=1}^{K} x_i\right]^2}{K\displaystyle\sum_{i=1}^{K} x_i^2} \tag{5.3}$$

where J is Jain's fairness index; x_i is the bandwidth share for flow i; K is the number of active flows in the system;

Based on the second assumption on TCP flows; where we assumed that TCP flows mapped to the same access link will eventually get a fair share of the link's bandwidth; we modified Jain's fairness index function to allow us to measure the flow fairness based on the Link Distribution Vector (LDV). The LDV is a set of mappings which tells us the number of flows currently mapped to each of the access links. The modified version of Jain's fairness index is defined as:

$$J_n = \frac{\left[\sum_{j=1}^{N} n_j \frac{b_j}{n_j}\right]^2}{K \sum_{j=1}^{N} n_j \left[\frac{b_j}{n_j}\right]^2} \tag{5.4}$$

$$= \frac{\left[\sum_{j=1}^{N} b_j\right]^2}{K \sum_{j=1}^{N} \frac{b_j^2}{n_j}} \tag{5.5}$$

subject to

$$\sum_{j=1}^{N} n_j = K \ , \forall n_j \in \mathbb{N} \tag{5.6}$$

where J_n is the revised Jain's fairness index; K is the number of active flows in the system; N is the number of links in the system; b_j is the current bandwidth in link j; and n_j is the number of flows mapped to link j

Maximizing both the system link utilization and flow fairness for the NEMO multi-homed system is a non-trivial optimization problem, since there is a potential tradeoff between these objectives. For example, consider a system with two access links (link A and link B) where their current available bandwidths are 9MBps and 1MBps respectively. There are two active flows in the system and the MR needs to make a decision on how to map these data flows among the two access links. To achieve maximum flow fairness (i.e. $J_n = 1$), the MR can assign both flows to either link A or link B, where the resulting system link utilization index will be 0.9 and 0.1 respectively. On the other hand, to maximize the system link utilization index (i.e. U = 1), the MR needs to assign a single flow to each link, which will result in a rather low flow fairness index of 0.61. If we consider another example where the current available bandwidths are identical in each link, we can achieve full system link utilization and a maximum flow fairness index by scheduling a flow to each link. Therefore, we can see that there is a potential trade-off between maximizing flow fairness and system link utilization. Given the potential trade-off between these two objectives, we define a system utility function that quantitatively captures both objectives:

$$utility = \alpha\, U + \beta\, J_n \tag{5.7}$$

where α represents the utilization weight factor; and β represents the fairness weight factor.

We aim to maximize both the system link utilization and flow fairness by designing a flow scheduler that aims to maximize the system utility index. The weight factors allow the system to prioritize one objective over another. However, since it is up to the mobile network operator to decide on how to prioritize these two scheduling objective; which may be influenced by a range of cost, political, or management factors; the question of how to set these two weight parameters is an open issue that is beyond the scope of this thesis. For example, the system operator may prioritize system link utilization over flow fairness since the operator may want to put more data through its system in order to serve more users. In contrast, one can argue that an unfair system will reduce user satisfaction, and therefore the mobile operator may decide to set a higher fairness weight factor in the system in order to maximize user satisfaction. Since these are non-technical issues, we therefore assume that α and β are both equal to one in future discussion. However, please note that the flow scheduling algorithms proposed in the rest of this chapter can all be applied even if the weight factors are set differently.

5.3 Flow scheduling without link switching

In this section, we discuss the design of the newly proposed *MaxUtility* flow scheduler. The role of the *MaxUtility* flow scheduler is to schedule new flows in a way that aims to maximize the system utility index.

For all incoming packets arriving at the flow scheduler, the system first performs a lookup in the flow switching table to see if there is an existing entry for the *FlowID* of this packet, which is a unique combination of the packet's source IP address, destination IP address, source port number, and destination port number. If there is an existing entry in the flow switching table for this *FlowID*, the MR will forward the packet to the access link that is mapped to this flow. If the packet is from a new flow, the MR will trigger its *flow assignment algorithm* to decide which access link this new flow should be mapped to. It will then create a new entry in the flow switching table, so that all subsequent packets from this flow will be forwarded to the selected access link.

In contrast to the RR and WRR flow schedulers that were discussed in the previous chapter, the proposed *MaxUtility* flow scheduler makes dynamic flow assignment decisions based on the current state of the access links. Using this information, the *MaxUtility* scheduler utilizes our newly proposed *MaxUtilityAssign* flow assignment algorithm that aims to assign new flows which maximize the current system utility index. The pseudo-code for the proposed *MaxUtilityAssign* flow assignment algorithm is shown in Algorithm 1.

The *MaxUtilityAssign* flow assignment algorithm first calculates the *expected system utility index* for each access link. The expected system utility index is calculated based on the assumption that the system has admitted the new flow to the

```
Data   : New flow with $FlowID$; Flow Switching Table: FST; Link Bandwidth
         Table: LBT
Result : Flow Switching Table: FST
Initialize maxUtility = 0, maxUtilityLink = NA;
Extract current set of $n_j$ from FST, store in $curLDV$;
for each $link \in LST$ do
    Increase $curLDV[link]$ by one;
    Compute $temUtilization$ from $curLDV$ and LBT;
    Compute $temFairness$ from $curLDV$ and LBT;
    Let $temUtility = temUtilization + temFairness$;
    Reduce $curLDV[link]$ by one;
    if $temUtility >= maxUtility$ then
        Set maxUtility = $temUtility$;
        Set maxUtilityLink = $link$;
    end
end
Add new mapping (FlowID to maxUtilityLink) to FST;
Return FST;
```

Algorithm 1: MaxUtilityAssign flow assignment algorithm

selected link in the current system state. The input of the computation are the current bandwidth values of the available access links; which is represented by the *LinkBandwidthTable*; and a modified LDV where we increased the number of flows mapped to the selected link by one. The scheduler calculates the expected system utility index for each access link, and assigns the new flow to the access link which generates the highest expected system utility index. By assigning flows that maximize the expected system utility index, we expect that the *MaxUtility* flow scheduler will provide a higher system link utilization and flow fairness for the system.

Since the *MaxUtilityAssign* flow assignment algorithm is only required to compute the expected system utility index for each access link, the algorithm has a computation complexity on $O(N)$. This can be considered a $O(1)$ algorithm (i.e. constant computation complexity), as it is unlikely that a NEMO multi-homed system will be

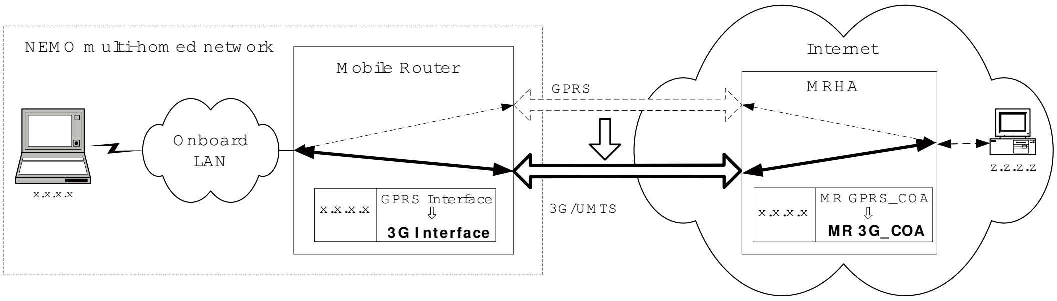

Figure 5.1: Flow re-scheduling in the NEMO multi-homed architecture

attached to a large number of access networks.

5.4 Flow scheduling with link switching

5.4.1 Motivation

As discussed in Chapter 3, one of the traffic engineering challenges for the NEMO multi-homed system is to adapt to the dynamic link characteristics of the wireless access links. Also, since the MR performs flow assignment upon the arrival of a new flow (i.e. before data traffic are sent by the user), the MR will not be able to determine the flow duration at the flow assignment phase. With these dynamic characteristics, the flow scheduler may not be able to maintain a high system utility index by just maximizing the system utility index in its flow assignment phase. Therefore, in order to adapt to the network resource variations and the flow dynamics, we proposed the use of *link switching* in the flow scheduler. Link switching allows the flow scheduler to re-schedule existing flows to different access links in order to maximize the system utility index. The bi-directional tunneling nature of the NEMO multi-homed architecture provides the MR with the capability to re-schedule users' data flows without breaking the end-to-end flow semantics. Figure 5.1 illustrates the flow re-scheduling process.

One of the benefits of the flow-based traffic engineering scheme is that out-of-order packet arrivals are minimized since the MR does not perform flow-splitting. However, when the MR re-schedules a flow from one link to another, the packets may be re-ordered since there are disparities in the performance characteristics of each access link [17]. The out-of-order problem will occur if the packets scheduled on the new link arrive at the end-hosts ahead of the packets scheduled on the original link. But since most current TCP implementations have a receiver buffer at the end hosts, there needs to be a certain amount of out-of-order packet arrivals at the end hosts to cause the TCP sender to trigger packet re-transmissions; and in the worst case, causes the TCP sender to go through the slow-start phase again. Thus; in contrast to the previous assumptions that all flow re-scheduling causes out-of-order packet arrivals at the end hosts [17]; we believe that there are only certain cases of flow re-scheduling which will affect the throughput performance of the data flows. To clarify this view, we simulated various flow re-scheduling schedules scenarios (with different link disparities) in NS2 to see its effects on the performance of TCP flows. The simulation architecture is shown in Figure 5.2. Our focus was to investigate whether a *CWD-Reset* occurred during the flow re-schedule. We define a CWD-Reset as the case where the size of the TCP sender congestion window drops to 1. This is the worst case scenario for TCP flows, since the TCP sender will need to go through the slow-start phase again and consequently, TCP throughput will be severely degraded.

In the flow re-scheduling simulations, we first initiated a 50MB file transfer (using the FTP and TCP protocol) between a user and its corresponding node from time 1s. A relatively large file size was chosen to allow the TCP congestion control mechanism

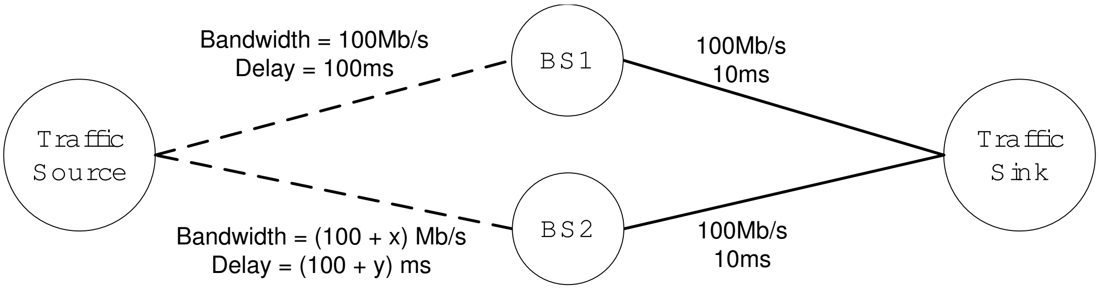

Figure 5.2: NS2 simulation architecture for the flow re-scheduling simulations

to stabilize before the flow is re-scheduled; which allows us to study the effects of packet re-ordering on the actual TCP protocol. At the start of the file transfer, the MR forwarded the data traffic from this flow onto link 1. At time 50s; where the TCP congestion control CWD value has stabilized to within a predictable range; the MR re-scheduled the flow by forwarding the subsequent packets to link 2. The simulations were terminated when the file had been successfully transferred.

A 3-dimension plot of the number of CWD-Resets versus the links bandwidth and propagation delay disparities is shown in Figure 5.3. The bandwidth and delay parameters that are shown in the x and y axis respectively, are the relative differences between the corresponding parameter between link 2 and link 1. For example, a bandwidth value of +1Mb means that the bandwidth parameter set in link 2 is 1Mb higher than that of link 1, and a delay value of -10ms means that the delay value of link 2 is 10ms less than that of link 1. As we can see from the graph, there are basically two regions in the graphs. Since we used identical simulation parameters in all of the simulation runs (except for the bandwidth and delay variables), the relative difference in the number of CWD-resets show us which type of flow reschedules will have an effect on the TCP flows. More precisely, the higher region in the graph

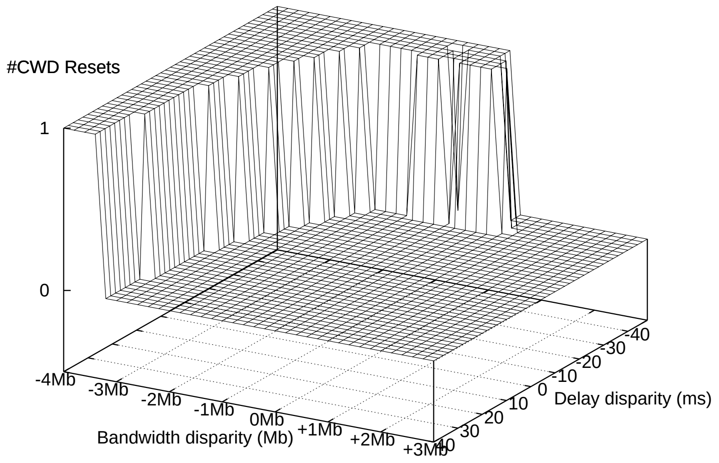

Figure 5.3: The number of CWD-Resets for re-scheduling a flow to links with different link disparities

shows the re-scheduling scenarios where the user's TCP flow experienced a CWD-Reset, hence causing it to have 1 more CWD-reset than the other flow re-scheduling scenarios. What we can conclude in this graph is that there are only a limited number of flow re-scheduling cases in which the user's TCP flow will be affected.

To determine which types of flow re-schedule scenarios will cause a CWD-reset for the users' data flows, the MR needs to take into consideration the end-to-end flow parameters of each individual user. These parameters include: propagation delay; bandwidth; queuing conditions; and receiver buffer size for every flow. The end-to-end parameters are not easily accessible in the MR. For example, to gain each flow's end-to-end propagation delay parameters, the MR will need to actively probe [50–52] the end points of the flow (i.e. the end-to-end users). Even if the probing mechanisms are implemented in the MR, the MR will need to constantly probe these end-to-end

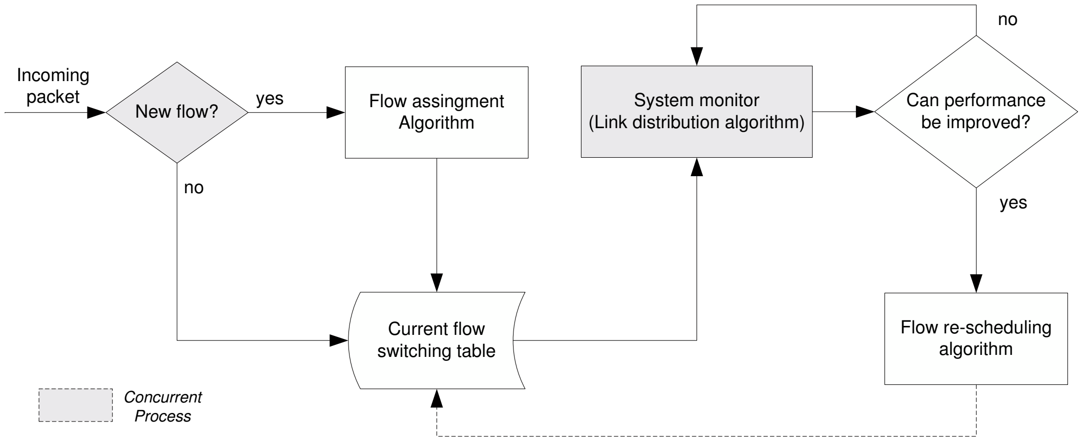
Incoming packet
New flow?
yes
no
Flow assingment Algorithm
Current flow switching table
System monitor (Link distribution algorithm)
Can performance be improved?
no
yes
Flow re-scheduling algorithm
Concurrent Process

Figure 5.4: Proposed flow scheduler framework

parameters since the values will be varying continuously when the system is in transit. Therefore, we believe that it will be highly un-scalable for the MR to probe such information for all the active flows in the system. Also, it is not worth the efforts for the MR to do this in order to avoid a small subset of flow reschedule cases where the user will experience a CWD-reset.

5.4.2 Link switching framework

The dynamic nature of the wireless access links and non-predictable flow departures means that the system utility index will fluctuate during the trip. To maintain a high system utility index, we propose an adaptive flow scheduler which periodically monitors the system utility index and re-schedules the existing flows in situations where the system utility index can be improved. The design of the link switching flow scheduler framework is shown in Figure 5.4.

In the proposed link switching flow scheduler framework, the MR continuously monitors the incoming packets and triggers the *flow assignment algorithm* upon the arrival of a new flow. The flow assignment algorithm determines which link the new flow will be mapped to, and creates an entry in the flow switching table so that subsequent packets will be forwarded to the selected link. Concurrent to this, the system periodically monitors the system utility index and determine whether the system utility index can be improved by re-scheduling the existing flows. To do this, the system uses a*link distribution algorithm* to calculate a LDV that maximizes the system utility index in the current system state. The computed LDV represents what the link distribution algorithm considers as the desired set of mappings that will maximize the system utility index in the current system state. If the link distribution algorithm computes a LDV that produces a higher system utility index than the current utility index, the system will trigger its *flow re-scheduling algorithm* to re-schedule all the existing flows from the current LDV to the desired LDV. The modular design of the framework allows the system to select various implementations of the flow assignment algorithm, link distribution algorithm, and flow re-scheduling algorithm that are independent of one another. As we have already proposed the *MaxUtilityAssign* flow assignment algorithm in the previous section, we will design a new link distribution algorithm and flow re-scheduling algorithm in the next subsection.

5.4.3 Link switching algorithms

Link distribution algorithm

To determine whether re-scheduling existing data flows can improve the system utility index, the system first needs to utilize a link distribution algorithm to compute a *desired* LDV that maximizes the current system utility index. Once this is computed, the system can use the resultant system utility index to determine whether re-scheduling the existing flows will improve the current system utility index.

To find the optimal LDV that produces the highest system utility index in the given system state, one can go through all possible combinations of LDV (i.e. n_j) and select the LDV which produces the highest system utility index. This has a computation complexity of approximately $O(N^K)$, which makes it unsuitable for real-time computation since the value of K (i.e. the number of active flows) can be very high. We can significantly reduce the LDV solution space by solving Equation.(5.5) with Lagrangian relaxation:

$$f(n_j) = K\sum_{j=1}^{N}\frac{b_j^2}{n_j} + \lambda(\sum_{j=1}^{N} n_j - K) \ , \forall n_j > 0 \tag{5.8}$$

Solving the partial derivatives,

$$\frac{\mathrm{d}f}{\mathrm{d}n_j} = -\frac{b_j^2}{n_j^2} + \lambda = 0 \tag{5.9}$$

$$\frac{\mathrm{d}f}{\mathrm{d}\lambda} = \sum_{j=1}^{N} n_j - K = 0 \tag{5.10}$$

$$\lambda = \frac{b_j^2}{n_j^2} \Rightarrow n_j = \frac{b_j}{\sqrt{\lambda}} \tag{5.11}$$

$$\sum_{j=1}^{N} n_j = \frac{\sum_{j=1}^{N} b_j}{\sqrt{\lambda}} = \frac{\sum_{j=1}^{N} b_j}{\frac{b_j}{n_j}} \tag{5.12}$$

$$\therefore n_j = K\frac{b_j}{\sum_{j=1}^{N} b_j} \ , \forall n_j > 0 \tag{5.13}$$

The above computation produces the ideal allocation of n_j, on the assumption that each access link will be utilized. To find the maximum possible system utility index in the current system state, the system now only needs to compute the optimal

LDV (using Equation 5.13) for all possible combinations of access link selections, and choose the LDV that provides the maximum system utility index. We call this the *MaxUtilityLDV* approach. There are only $O(2^N - 1)$ number of access link selections and therefore, this approach has significantly reduced the entire solution space from to $O(N^K)$ to $O(2^N - 1)$. Given that we expect the number of links attached to the MR to be relatively low in relation to the computation complexity (i.e. N less than 10), this approach will be suitable for real time computation.

However, the computed LDV can be fractional in the above computation. Therefore, the system may need to convert the solution to an integer solution if the computed LDV contains fractional values. This is because the flow-based traffic engineering scheme does not allow mapping partial flows onto the different links (i.e. flow-splitting). To convert the fractional LDV to an integer LDV, we propose a LDV round off heuristic *LdvRoundOff* that aims to find the integer LDV which yields the highest system utility index. The details of the *LdvRoundOff* heuristic is shown in Algorithm 2.

The intuition behind the proposed *LdvRoundOff* heuristic is to search for the closest integer solution around the optimal fractional LDV. The quadratic nature of the utility function means that the optimal integer LDV solution will be very close to the optimal LDV solution, and hence the idea is to greedily select the integer LDV around the optimal fraction LDV solution that yields the highest system utility index. The heuristic first greedily adjust the fractional values in the optimal LDV, and then selects the integer LDV which yields the highest system utility index for the system.

The combined computational complexity for the *MaxUtilityLDV* approach and the *LdvRoundOff* heuristic is approximately $O(2^N + N - 1)$, which is equivalent to

```
Data    : Fractional link distribution vector (FLDV), Link Bandwidth Table
          (LBT)
Result  : Integer link distribution vector (ILDV)
Initialize numFractionalFlows = 0, numCeiledFlows = 0;
for each $n_j \in FLDV$ do
    numFractionalFlows += $n_j$;
    numCeiledFlows += ceil($n_j$);
    Store ceil($n_j$) in ILDV;
end
Let w = numCeiledFlows - numFractionalFlows;
for $i \leftarrow 1$ to $w$ do
    Let maxUtility = 0, maxLinkDistVector = NA;
    for each $n_j \in ILDV$ do
        Decrease $n_j$ in ILDV by one;
        Compute $temUtilization$ from ILDV and LBT;
        Compute $temFairness$ from ILDV and LBT;
        Let $temUtility = temUtilization + temFairness$;
        if $temUtility >= maxUtility$ then
            Let maxUtility = $temUtility$;
            Let maxLinkDistVector = ILDV;
        end
        Increase $n_j$ in ILDV by one;
    end
    Set ILDV = maxLinkDistVector;
end
return ILDV;
```

Algorithm 2: Heuristic for rounding off optimal fractional link distribution vector

a complexity of $O(2^N)$. As discussed previously, we do not expect this approach to have a high computation complexity since it is unlikely that the MR will be equipped with a large number of links. However, if the number of links is large enough to make it unsuitable for real time computations, we proposed another heuristic that greedily selects links that maximizes the utility index. The details of this heuristic is shown in Algorithm 3.

```
Data    : Link Bandwidth Table (LBT)
Result  : New Link Distribution Vector
Initialize maxUtility = 0, maxLinkDistVector as empty map, selectedLinks as empty vector;
Sort the links in LBT in descending current bandwidth value and store in sortedLinks;
for each link $\in$ sortedLinks do
    add link to selectedLinks;
    Compute curLDV from selectedLinks using maxUtility heuristic;
    Compute temUtilization from curLDV and LBT;
    Compute temFairness from curLDV and LBT;
    Let $temUtility = temUtilization + temFairness$;
    if $temUtility >= maxUtility$ then
        Set maxUtility = temUtility;
        Set maxLinkDistVector = curLDV;
    end
end
return maxLinkDistVector;
```

Algorithm 3: Heuristic for maximizing utility function

The heuristic works by first sorting the link in descending bandwidth value. The algorithm selects the link with the highest bandwidth in the sorted list and computes the optimal LDV and the corresponding system utility index using the LDV round off algorithm for the selected link. It then adds the next highest bandwidth link from the sorted list to its selected link list, and computes the LDV and system utility index using the *LdvRoundOff* heuristic. If the current system utility index is higher than the previously computed utility index, it will continue adding links to its selected link list from the sorted list. Otherwise, it will cease the computation and use the selected link as the target LDV. The proposed heuristic has a complexity of $O(N)$.

Flow re-scheduling algorithms

If the link distribution algorithm determines that the system utility index can be improved, the existing flows will need to be re-scheduled according to the new LDV. Since the LDV only represents the number of flows that are mapped to each access link, there are many possible ways in which the existing flows can be re-scheduled from the current LDV to the new LDV. Therefore, we designed a new re-scheduling algorithm that aims to minimize the number of link switches for the existing data flows. We call this the *MinLinkSwitch* flow re-scheduling algorithm and the details of this algorithm is shown in Algorithm 4.

The *MinLinkSwitch* flow re-scheduling algorithm goes through each entry in the existing flow switching table and tries to maintain the same access link mapped to each $FlowID$ whenever possible. This will allow the system to minimize the number of flow re-schedules. If the flow can be mapped to its original access link, the algorithm will create an identical entry for this flow in the new flow switching table. Otherwise, the flow is added to a list of the *unmappedflowslist* as the flow cannot be mapped to its original access link. After the algorithm goes through every entry in the existing flow switching table, the algorithm will go through the flows in the *unmappedflowslist* and assign a valid access link for each of these flows. Once the new flow switching table is fully created, subsequent packets of the existing paths will be forwarded to the access links according to the new flow switching table.

```
Data     : Old Flow Switching Table: OldFST; Target Link Distribution Vector:
           maxLDV
Result   : New Flow Switching Table: NewFST
for each $flowID \in OldFST$ do
    set existLink = OldFST[$flowID$];
    if maxLDV[existLink] > 0 then
        decrease maxLDV[existLink] by one;
        NewFST[$flowID$] = existLink;
    end
    else
        add $flowID$ to unmapList;
    end
end
for each $flowID \in unmapList$ do
    for each $linkID \in maxLDV$ do
        if maxLDV[linkID] > 0 then
            set NewFST[$flowID$] = $linkID$;
            decrease maxLDV[linkID] by one;
            break;
        end
    end
end
return NewFST;
```

Algorithm 4: Flow re-scheduling algorithm

5.5 Predictive flow scheduling

One of the unique characteristics associated with PTVs; which is the main candidate for deploying the NEMO multi-homed systems; is that these vehicles repeatedly traverse the same fixed route. As pointed out several times in this thesis, the main challenge with flow scheduling in the NEMO multi-homed system model comes from the highly dynamic link characteristics, that is caused by the movement of the vehicle and the nature of wireless links. Therefore, the aim of this section is to see whether the system can take advantage of the PTVs' fixed route characteristic to improve its flow scheduling performance.

Since the MR in the PTVs traverse fixed routes, it is logical to try and make predictions on the link characteristics in each trip. This can be performed by analyzing the data and trends recorded in past trips. We have made preliminary efforts to predict the link characteristics over a live GPRS network and the details are presented in Appendix C.

Figure 5.5 illustrates a sample bandwidth prediction graph for a single link, where the predicted link bandwidth is plotted against the predicted time interval of a PTV route. Instead of using the *experienced bandwidth* as we did in the previous section (i.e. the *LinkBandwidthTable*), the flow scheduler can base its flow assignment and re-assignment decisions on the *predicted bandwidth* of each access link. Rather than reacting to degraded system utility index as a result of the link dynamics, the flow scheduler can utilized the predicted bandwidth values to make proactive flow scheduling and re-scheduling decisions that maximize the system utility index.

A limitation with using the predicted link bandwidth is that the actual bandwidth values may sometimes be too volatile for the system to make accurate fine

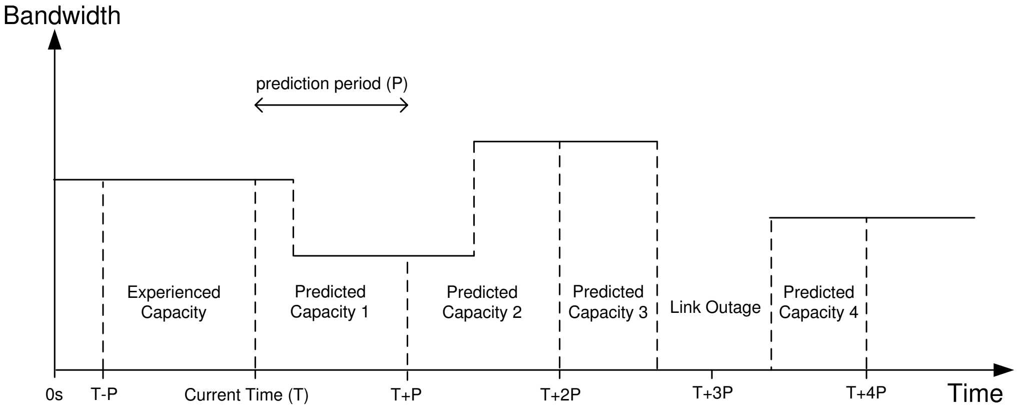

Figure 5.5: A sample bandwidth prediction graph

grain predictions on it. In fact, even if the predictions are accurate, the system will be constantly re-scheduling its flows if it tries to increase its system utility index according to the bandwidth variations. Hence, instead of using the predicted bandwidth values, we proposed the use of *predicted capacity* instead. The term capacity represents the amount of data that a link can send over a given time period. This can be predicted by analyzing the throughput statistics measured in past trips; or if the predicted bandwidth to time graph is given; it can be calculated by integrating the time-bandwidth function. By monitoring the system utility index at regular intervals, and making flow assignment and flow re-scheduling decisions based on the predicted capacity over a fixed period of time (i.e. the monitoring period), the system will avoid being over-sensitive to the actual predicted bandwidth values.

5.6 Simulation results and analysis

To simulate the performance of the proposed flow schedulers, a flow scheduling framework for the mobile multi-homed network architecture was implemented in the NS2

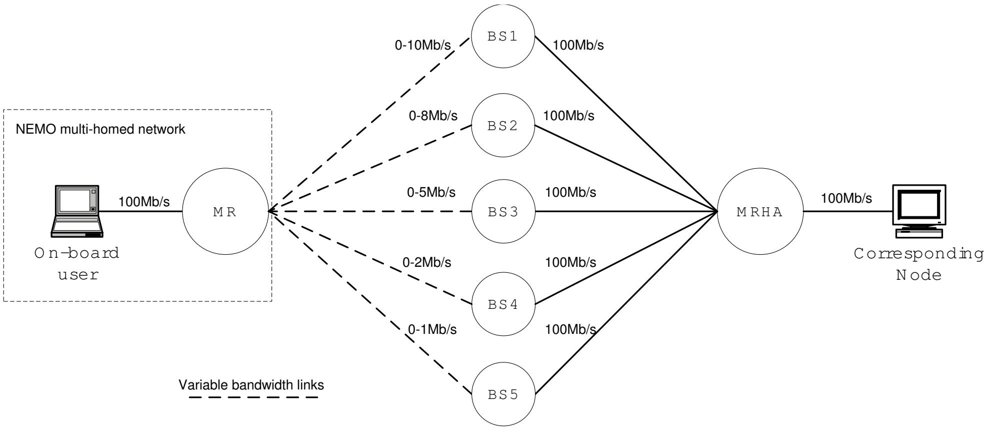

Figure 5.6: NS2 simulation architecture for the flow schedulers

simulation software [53]. The simulation model consists of a multi-homed system connected to 5 access networks, whose range of bandwidth values are specified in the simulation architecture shown in Figure 5.6. The proposed flow schedulers are implemented in the MR, and the data traffic is modeled by sending data flows from the on-board users to their CNs. To simulate the effects of network mobility, the bandwidth of the access links were varied randomly from zero (representing outage) to the maximum bandwidth of each access link. The bandwidth variations followed a uniform distribution and the times between each variation (i.e. inter-variation times) were exponentially distributed. For a fair comparison of the flow schedulers, we used the same set of parameters for the simulations of the different schedulers under the same simulation scenario.

5.6.1 Flow scheduling without link switching

In the first set of experiments, we compared the performance of our proposed *MaxUtility* flow scheduler with the RR flow scheduler (as studied in [28]); the WRR flow scheduler; and the Least Loaded Link (LLL) flow scheduler. The RR and WRR flow schedulers were discussed in Chapter 4 and hence we will not discuss their implementation details again.

In contrast to the RR and WRR schedulers; where the schedulers do not utilize any system state information in its scheduling decisions; the LLL flow scheduler makes its scheduling decisions based on the current state of the flow scheduler. For every new flow, the LLL flow scheduler assigns the new flow to the link with the minimum number of flows currently mapped to it. The least loaded link can be derived from the current flow switching table by counting the number of flows mapped to each access link, and selecting the one with the minimum number of flows mapped to it. In the case where there are multiple links with the same minimum load, our implementation of the LLL flow scheduler assigns the new flow to the link with a higher maximum bandwidth value.

To simulate on-board data traffic, users randomly started file transfers of fixed-size (10Mb) files to their corresponding nodes. The flow start times were exponentially distributed between 0-500 seconds and the simulations ceased when all the data flows were transferred successfully. To obtain reasonable confidence in the simulation results, we performed 30 sets of experiments for each simulation scenario and used the mean of the results in our analysis.

Figure 5.7 shows the mean flow transfer time versus increasing system load for the *MaxUtility* flow scheduler and the other schedulers. In this set of experiments, our focus was on the flow scheduling performance; therefore, the flow re-scheduling or bandwidth prediction features were not implemented in the *MaxUtility* flow scheduler. As expected, the RR flow scheduler had the highest mean flow transfer times among

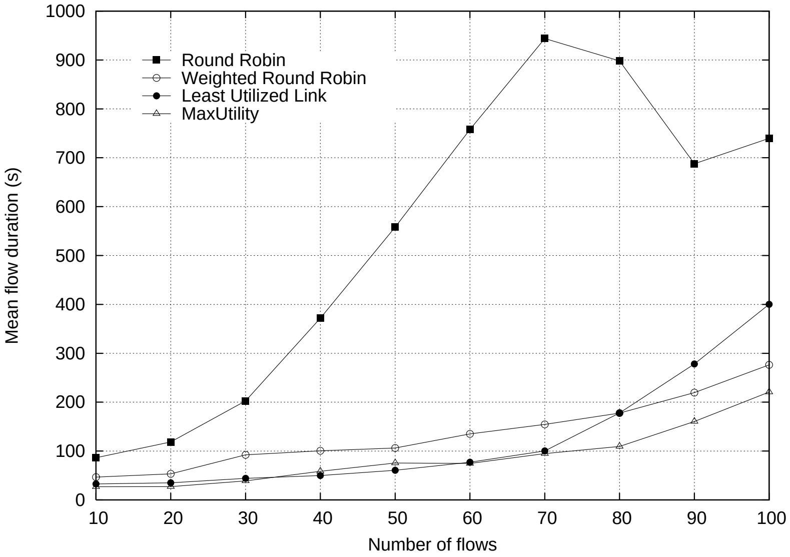

Figure 5.7: Number of flows versus mean flow transfer times for the RR, WRR, LLL, and MaxUtility flow schedulers

the flow schedulers. As the load was increased, the mean flow transfer time of the RR flow scheduler also increased significantly. The WRR flow scheduler had a lower mean flow transfer time than the RR flow scheduler in all scenarios since the WRR flow scheduler was able to map more flows to the higher bandwidth links.

The LLL flow scheduler showed some interesting behavior. In the low to medium load scenarios (i.e. from 10 flows to 70 flows), the LLL flow scheduler matched the performance of the *MaxUtility* flow scheduler since it was able to make use of the unutilized links whenever possible. But as the load increased from 70 flows to 100 flows, the mean flow time increased significantly since the scheduler basically performed RR scheduling when there are multiple flows mapped to each access link. Even without implementing the link switching and bandwidth prediction schemes, the *MaxUtility* flow scheduler outperformed all the other flow schedulers in most of the cases. As the

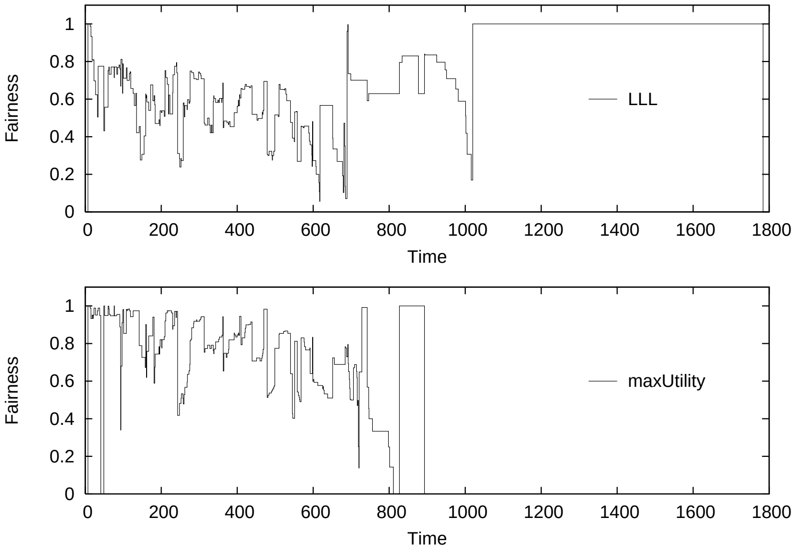

Figure 5.8: Flow fairness index trace for LLL and MaxUtility schedulers in a sample 100 flows simulation run

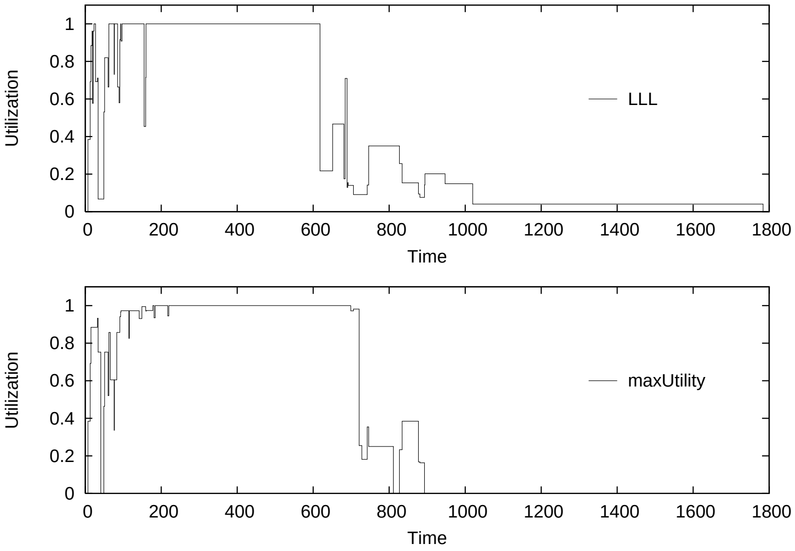

Figure 5.9: System link utilization index trace for LLL and MaxUtility schedulers in a sample 100 flows simulation run

load increased, the *MaxUtility* scheduler was able to maintain a similar mean transfer times for the data flows by making better use of the access links.

Figure 5.8 and Figure 5.9 show traces of the flow fairness and system link utilization index for the LLL and *MaxUtility* flow scheduler respectively, for one of the simulation runs in the 100 flows scenario. The simulation termination times (i.e. the x-axis values) were different for both sets of graphs since the *MaxUtility* flow scheduler was able to transfer all the flows in approximately half the time of the LLL flow scheduler. The utilization traces showed a similar trend for the two flow schedulers, where both were able to maintain a maximum utilization during the first 600 seconds of the simulation. As the starting times of the data flows were uniformly distributed between 0-500s, it is evident that both flow schedulers were able to utilize every access link when the traffic demand was high. In contrast, the flow fairness index trace of the *MaxUtility* flow scheduler was significantly higher than the LLL flow scheduler during the same period. This is because the LLL flow scheduler assigned more flows to the BS5 link than the *MaxUtility* flow scheduler; hence, the flow fairness index of the LLL flow scheduler was severely penalized. In fact, the LLL flow scheduler was assigning the flows to the access links in a RR manner in the situations where it needs to schedule multiple active flows. After 1000s, when the *MaxUtility* scheduler had finished transferring all the data flows, the LLL flow scheduler maintained a low utilization index and a constant maximum flow fairness index of 1. This is because the LLL flow scheduler was still transferring all the flows that were mapped to BS1, which had a very little bandwidth compared to the other links. This significantly increased the transfer times for the remaining flows that were mapped to BS1. In contrast, the *MaxUtility* flow scheduler only scheduled a single flow to BS1 and hence it was able to provide a much lower transfer time for this flow. This highlights the importance of enforcing flow fairness in the system, where the disparity among the flow duration times will be minimized.

5.6.2 Flow scheduling with link switching

In the second set of experiments, we analyzed the performance and behavior of implementing flow re-scheduling in the *MaxUtility* flow scheduler. Unlike the previous set of experiments; where we simulated the transfer of multiple small fixed-size flows in the system; we instead simulated the transfer of one large 100Mb flow for both the *MaxUtility* flow scheduler and the *MaxUtility-R* flow scheduler with flow re-scheduling capabilities. This allowed us to a take a closer look at the impact of flow re-scheduling on the TCP flow performance of both schedulers. The *MaxUtility-R* flow scheduler monitored the system utility index every second and it triggered the re-scheduling algorithms whenever it detects that the system utility index can be improved. The simulation parameters were consistent in the simulations of both schedulers and the simulations were terminated after the data flow had been transferred successfully.

The link selection and flow throughput trace for the *MaxUtility* and *MaxUtility-R* schedulers are plotted in Figure 5.10 and Figure 5.11 respectively. The link selection trace is used to show the access links that were utilized by the flow schedulers for data transfer during the flows lifetime. As expected, the *MaxUtility* flow scheduler mapped the data flow to the same link (i.e. BS5) throughout the simulation since re-scheduling was not implemented. In contrast, the *MaxUtility-R* flow scheduler adapted to the link bandwidth variations by regularly re-scheduling the data flow in order to maximize the system utility index.

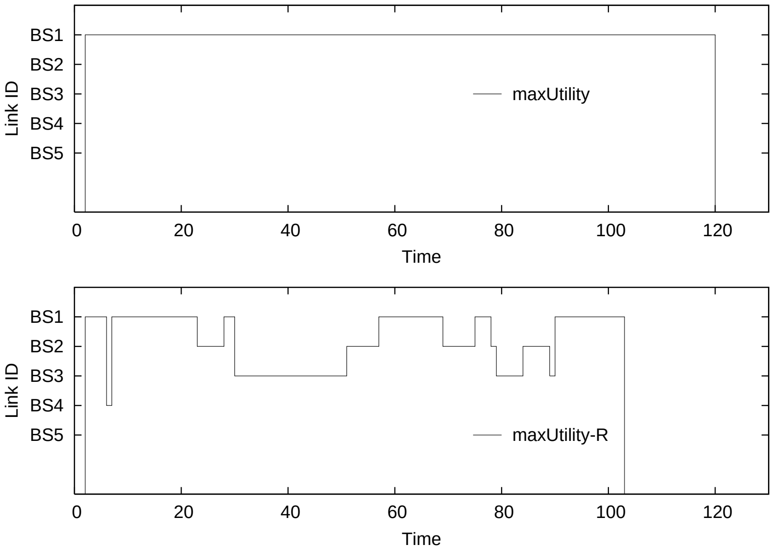

Figure 5.10: Link selection for MaxUtility and MaxUtility-R schedulers

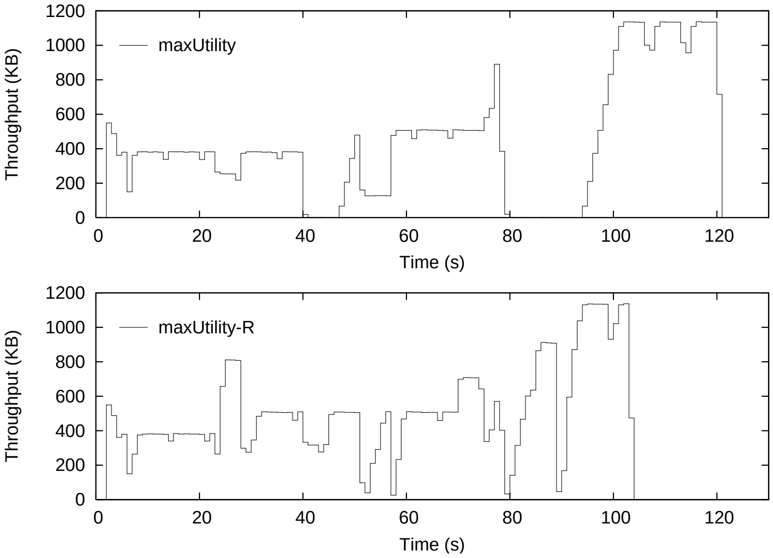

Figure 5.11: Throughput for MaxUtility and MaxUtility-R schedulers

In the link selection trace, it showed that the *MaxUtility-R* flow scheduler re-scheduled the flow from BS1 to BS4 at time 6s, and then re-scheduled the flow back to BS1 at time 7s. However, the throughput trace showed that the *MaxUtility* scheduler had a higher throughput than the *MaxUtility-R* scheduler during this period. This is because the *MaxUtility-R* reacted to the slight bandwidth glitches in the BS1 link, by re-scheduling the flow to BS4 which had a higher bandwidth at that point of time. The system only utilized the higher bandwidth link for a very short period of time (i.e. 1s) before re-scheduling the data flow back to BS1, and hence the TCP flow was not able to make use of the higher bandwidth link.

At time 23s, the link selection graph shows that the *MaxUtility-R* scheduler re-scheduled the flow from BS1 to BS2, and switched it back to BS1 at time 28s. This time, the throughput of *MaxUtility-R* was doubled the throughput value of the *MaxUtility* flow scheduler during this period, since the TCP flow had more time to utilize this higher bandwidth via its AIMD congestion control mechanism. This illustrates the effectiveness of flow re-scheduling, where the system was able to provide a higher throughput to the data flow by shifting the data traffic from a low bandwidth link to a higher bandwidth link.

At time 79.16s and 89.03s, the throughput trace of the *MaxUtility-R* scheduler showed significant performance improvement over the *MaxUtility* scheduler since there were prolonged link outages in BS1 during that period. The *MaxUtility-R* avoided the outage by scheduling the data flow to BS3 and BS2 during this period. Since the simulation trace ended after the flow had been successfully transferred, we can see that the flow transfer time of *MaxUtility-R* was approximately 20% lower than the *MaxUtility* scheduler.

5.6.3 Predictive flow scheduling

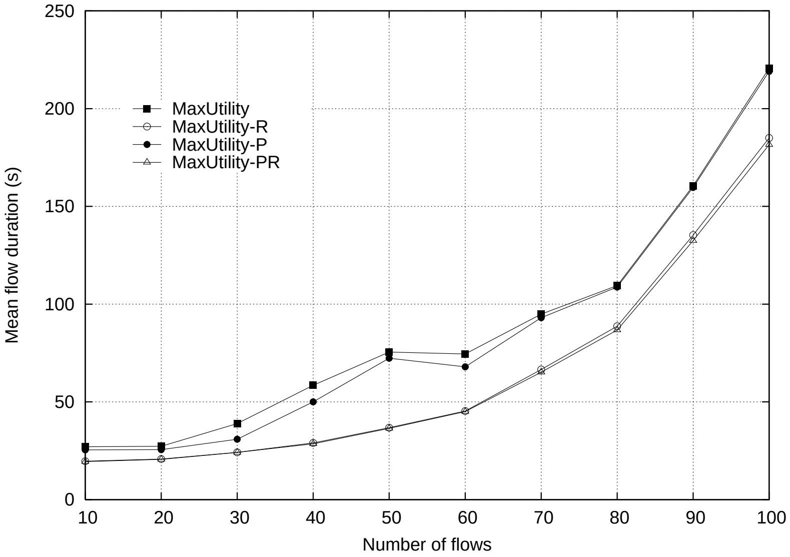

Figure 5.12: Number of flows versus mean flow transfer time for the MaxUtility, MaxUtility-R, MaxUtility-P and MaxUtility-PR flow schedulers

In the third set of experiment, we simulated the performance of the *MaxUtility-PR* flow scheduler, where both the flow re-scheduling and bandwidth prediction features were implemented in the *MaxUtility* flow scheduler. To simulate bandwidth prediction, we assume that the *MaxUtility-PR* scheduler had access to the expected time-to-bandwidth graph from the current time (T) to the prediction interval ($T+P$). The prediction period and the system monitoring period were both set to 1 second, and the predicted capacity was calculated by integrating the time-to-bandwidth graph for the prediction period.

The experiments were performed using the same user data traffic parameters as the first set of experiments. We simulated the performance of the *MaxUtility*, *MaxUtility-P*, *MaxUtility-R*, and *MaxUtility-PR* flow schedulers and their mean flow transfer time are shown in Figure 5.12. From the results, it is clearly evident that the implementation of flow re-scheduling provided a higher reduction on the mean flow transfer time than the implementation of link prediction in the *MaxUtility* flow scheduler, since flow re-scheduling allowed the system to adapt to the dynamic bandwidth in the access links. The utilization of link bandwidth predictions in the *MaxUtility-P* and *MaxUtility-PR* flow schedulers, provided marginal reduction on the mean flow transfer time for their respective *MaxUtility* and *MaxUtility-R* non-predictive flow scheduling counterparts.

The flow fairness index trace and the system utilization index trace of the *MaxUtility-R* and *MaxUtility-PR* flow schedulers; for one of the simulation runs in the 100 flows scenario; are shown in Figure 5.13 and Figure 5.14 respectively. For the *MaxUtility-R* flow scheduler, there are many performance *glitches* (i.e. temporary performance drops) in both the flow fairness index and system utilization index traces. These glitches may be caused by a sudden change in the link bandwidth values, or the termination of a particular flow. Since the *MaxUtility-PR* did not implement any flow size prediction mechanisms, the flow fairness index and system utilization index traces of the *MaxUtility-PR* shows that the use of bandwidth predictions allowed the *MaxUtility-PR* to reduce both the size and number of performance glitches in the system. The *MaxUtility-PR* flow scheduler was able to achieve this by making proactive flow scheduling and re-scheduling decisions based on the predicted capacity of the access links.

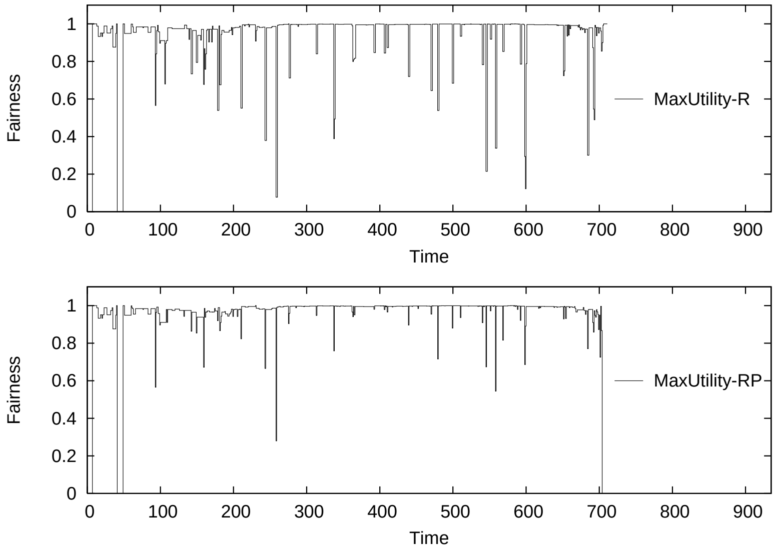

Figure 5.13: Flow fairness index for MaxUtility-R and MaxUtility-PR schedulers in a sample 100 flows simulation run

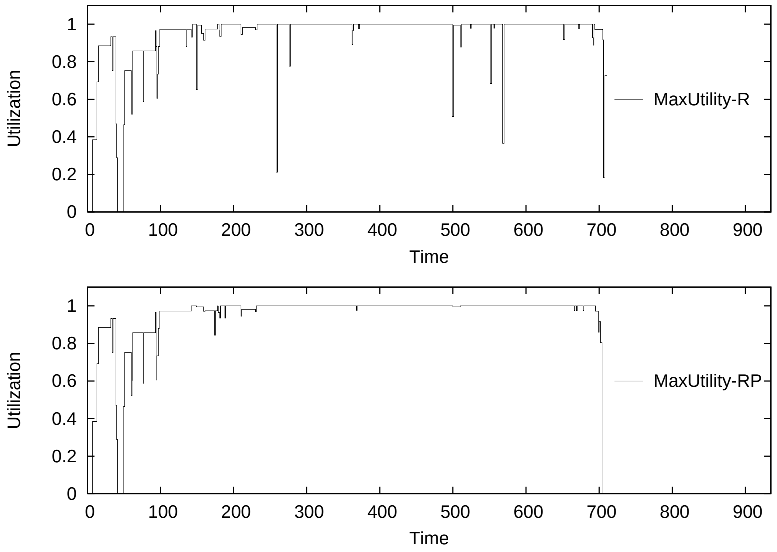

Figure 5.14: System link utilization index for MaxUtility-R and MaxUtility-PR schedulers in a sample 100 flows simulation run

5.7 Summary

The contributions of this chapter can thus be summarized as follows:

- We formulated the optimization problem of maximizing both flow fairness and system link utilization under the NEMO multi-homed system, and showed that there is a potential tradeoff between these two objectives. This trade-off is captured quantitatively by defining a utility function that accounts for system link utilization on the one hand, and flow fairness on the other.

- We proposed a *MaxUtility* flow scheduler, which employed an optimal flow assignment heuristic for maximizing the utility function. Given the dynamic nature of the access links, we proposed a *MaxUtility-R* flow scheduler which regularly monitors the system and triggers a re-scheduling algorithm in the event of sub-optimal performance. In contrast to the views in previous literature [17], we also showed that in most of the cases, flow re-scheduling is not harmful to TCP flows.

- Furthermore, to take advantage of the fact that PTVs regularly traverse fixed routes, we proposed a *MaxUtility-PR* flow scheduler which utilizes the link throughput predictions in its flow scheduling and flow re-scheduling decisions.

- By performing an extensive set of simulations in NS2, the results shows that our proposed *MaxUtility* flow scheduler was able to achieve significantly a higher system throughput and flow fairness than previously proposed flow schedulers for the NEMO multi-homed systems. Also, the implementation of flow re-scheduling was able to provide a larger reduction on the flow transfer times than the utilization of link throughput predictions in our proposed *MaxUtility* flow scheduler.

Chapter 6

Profit optimization with user-based traffic engineering

In this chapter, we present the design of two new user-based traffic engineering solutions that aims to maximize the profit of the mobile hotspot operator under the highly popular volume-based charging model, while maintaining bandwidth guarantees for the on-board users. Our results shows having a prior knowledge on the link bandwidth variations does not provide additional profit to the system, but instead it allows the mobile hotspot to limit the service disruption probability for the users.

6.1 Introduction

Previous research on multi-homed systems, including the work presented in the flow-based traffic engineering algorithms proposed in Chapter 5, mainly focused on designing traffic engineering solutions that only maximizes the technical performance of the system and users. To the best of our knowledge, the issue of profit and performance optimization for the mobile hotspot operator has not been addressed before. We contend that, given the proliferation of wireless service providers with diverse charging rates, profit and performance optimization will be a key design requirement for traffic distribution computations in mobile hotspots.

In this chapter, we propose and compare two user-based traffic engineering schemes: *in-transit* and *pre-transit*, which aims is to maximize the profit of the mobile hotspot operator under a common charging model, while providing acceptable level of service to the users. The *in-transit* scheme assumes no prior knowledge of potential network changes that might occur during the trip and computes optimum traffic distribution on-the-fly whenever a change is detected. In contrast, the *pre-transit* scheme exploits *a priori* knowledge of all possible network changes along a known and repetitive route of a PTV and computes the whole set of traffic (re)distributions before the trip starts. We provide results from a detailed simulation study of these two schemes under different user demands.

This chapter is structured as follows. In Section 6.2, we provide an in-depth discussion on the mobile hotspot model. Section 6.3 provides the problem formulation for profit and performance optimization under the mobile hotspot model. Section 6.4 presents the design of our proposed *in-transit* and *pre-transit* user-based traffic engineering schemes. Section 6.5 provides details of our simulation platform and highlights the significant findings in our results. In Section 6.6, we conclude this chapter with a summary of its major contributions.

6.2 Modeling hotspots in public transports

6.2.1 Public transport user model

Our focus is primarily on deploying mobile hotspots in PTVs. To model the characteristics of such systems, we formulated a system model illustrated in Figure 6.1.

The multi-homed mobile hotspot is assumed to be equipped with N network

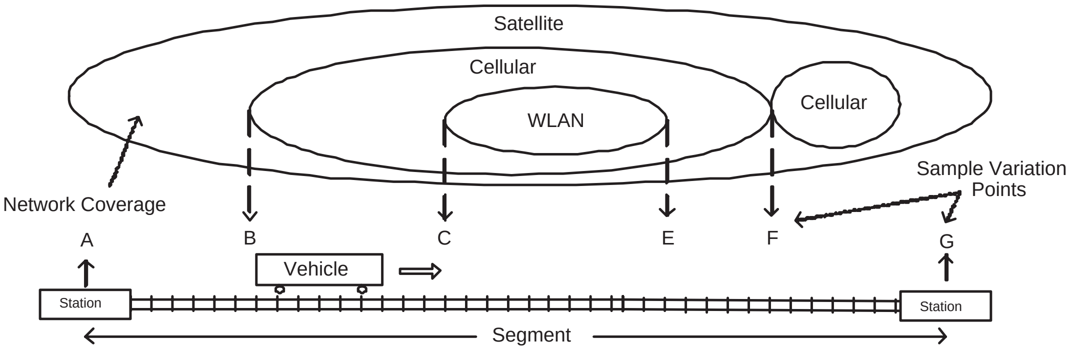

Figure 6.1: Mobile hotspot system model

interfaces, which can be used to connect to N different access technologies provided by either the same service provider, or several different service providers. Different service providers of the same access technology will also be considered multi-homing since each service provider will deploy a different network infrastructure. The mobile hotspot is assumed to be running a NEMO protocol [6], which provides transparent network mobility to the on-board passengers. Users can only board the vehicle at each *station*. The route between each station is defined as a *segment*. As the vehicle travels along each segment, it will experience coverage and resource variations which affect the set of available networks and the networks' available bandwidths. There are K number of these locations which are called *variation points*. Each variation point has a resource vector $\mathbb{B}_n$, which represents the available bandwidth of each network at that location.

6.2.2 Quality of service model

In the multi-homed mobile hotspot, there are S service classes that are available to the passengers. The service classes are similar to the home broadband (i.e. ADSL, cable) plans where users in each service class have a bandwidth limit (e.g. 56KBps to 1.5MBps). These are represented by the service data rate vector $\mathbb{D}_s$. When a passenger boards the mobile hotspot at each station, they must indicate their service class preference, which will be valid for the whole segment. The total number of requests in each service class is represented by the requirements vector $\mathbb{R}_s$.

6.2.3 Cost and charging model

Our research is based on the *total volume charging* scheme [54, 55]. In this charging scheme, the access cost is calculated by applying a non-decreasing linear function over the total volume used in that particular network. The scheme is simple yet highly popular, as the charging model for most wired and wireless Internet access network providers are based on some variants of the volume based charging scheme [19, 56, 57]. In our charging model, we assume that all the cost functions are expressed in terms of price per unit of data transfer (e.g. cents per kilobyte).

The mobile hotspot offers S number of service classes to the passengers. To differentiate the various service classes, we follow an approach similar to most home broadband (i.e. ADSL, cable) plans [57] where the data rate of each user in a service class is capped to a certain speed (e.g. 56KBps, 512KBps, 1.5MBps etc). The maximum data rates for each service class are represented by the service data rate vector $\mathbb{D}_s$, and the elements are sorted in ascending order such that users with a higher service index s represents that they have a higher maximum data rate. The MR is assumed to be capable of exhibiting user rate control over the access networks similar to the gateways shown in [58, 59].

In our model, we assume that all the available wireless access networks will follow the total volume based charging scheme. The network access cost vector $\mathbb{C}_n$, represents the costs incurred (paid by mobile hotspot operator) for transferring each unit of data on an wireless access network n. The vector $\mathbb{C}_n$ is sorted in ascending order, and can contain identical values as there is no restriction preventing networks to have identical charging functions.

Similarly, users in each service class are also charged according to the total volume based charging scheme. The user service charge vector $\mathbb{U}_s$ refers to the charges (paid by users) for transferring each unit of service class s data via the mobile hotspot. The vector $\mathbb{U}_s$ is also sorted in ascending order but unlike $\mathbb{C}_n$, elements in $\mathbb{U}_s$ cannot contain identical values as we want to differentiate the service classes strictly on its costs and maximum data rates. Hence, users in a service class with a higher maximum data will *always* pay a higher cost per data unit (transferred over the mobile hotspot) than a user on a lower maximum data rate service class.

6.3 Problem formulation

6.3.1 Profit function

We define the *profit function* of the mobile hotspot operator to be the sum of the differences between the user service costs and the network access cost in each network. This basically means that mapping each unit of data traffic of service class s to network n, will incur a different profit for the mobile hotspot operator. The profit function of each network can be represented by:

$$\alpha_{s,n} = u_s - c_n \tag{6.1}$$

where $\alpha_{s,n}$ represents the mobile hotspot's profit function for transferring each unit of data service class s via access network n; u_s is the user service charge for service class s; and c_n is the network access cost for network n.

In the total volume based charging model, the actual profit obtained will depend on the volume of different service class traffic passing through each network. Therefore, the profit functions for the mobile hotspot operator are:

$$p_k = \sum_{s=1}^{S} \sum_{n=1}^{N} \alpha_{s,n} v_{s,n,k} \tag{6.2}$$

$$p_T = \sum_{k=1}^{K} \sum_{s=1}^{S} \sum_{n=1}^{N} \alpha_{s,n} v_{s,n,k} \tag{6.3}$$

where p_k represents the profit generated between each variation point k; p_T represents the total profit generated for the entire trip; and $v_{s,n,k}$ represents the total volume transferred over network n for passengers in service class s between variation point k and $k+1$

The total profit P for the mobile hotspot operator will be the sum of all the elements in the following profit matrix:

$$P_{s,n} = \begin{pmatrix} \alpha_{1,1}v_{1,1} & \alpha_{1,2}v_{1,2} & \cdots & \alpha_{1,N}v_{1,N} \\ \alpha_{2,1}v_{2,1} & \alpha_{2,2}v_{2,2} & \cdots & \alpha_{2,N}v_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{S,1}v_{S,1} & \alpha_{S,2}v_{S,2} & \cdots & \alpha_{S,N}v_{S,N} \end{pmatrix}$$

where $v_{s,n}$ represents the total volume of service class s traffic transferred over network n in the entire segment.

6.3.2 Profit objectives

In the user-based traffic distribution scheme, profit maximization in the total volume based charging model is a challenging problem. As we have discussed in our charging model, the profit functions depends on the amount of traffic transferred over the wireless access networks. In the user-based scheme, the MR computes the traffic distribution vectors *before* traffic is actually transferred over the wireless access networks. Therefore, both schemes cannot compute the optimal profit maximization traffic distribution vector without having a prior knowledge on the traffic volume sent by each session. This brings the need to design some online profit maximization heuristics for MRs using the user-based switching scheme.

However, as the profit functions depend on the volume of data traffic admitted onto each network, optimum profit maximization is not possible since the passengers' data usage pattern cannot be known in advance. Therefore, we propose a profit maximization heuristic that determines the *best* number of users (in each service class) to be *mapped* to each network (i.e. *user traffic distribution*). At each variation point, the mobile hotspot computes a user traffic distribution vector $\mathbb{X}_{s,n}$ which determines how many users of each service class s should be mapped to each access network n from the current variation point to the next. Assuming all users exhibits a similar usage pattern; where the actual usage U of each user can be modeled by some probability distribution; the proposed profit maximization heuristic should perform well.

Assuming all users transfer the same volume of traffic over the mobile hotspot, the total *estimated profit* P' will be the sum of all elements in the following matrix:

$$P'_{s,n} = \begin{pmatrix} \alpha_{1,1}x_{1,1} & \alpha_{1,2}x_{1,2} & \dots & \alpha_{1,N}x_{1,N} \\ \alpha_{2,1}x_{2,1} & \alpha_{2,2}x_{2,2} & \dots & \alpha_{2,N}x_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{S,1}x_{S,1} & \alpha_{S,2}x_{S,2} & \dots & \alpha_{S,N}x_{S,N} \end{pmatrix}$$

where $x_{s,n}$ represents the number of service class s users mapped to network n in the entire segment.

6.3.3 Performance objectives

To ensure each passenger get an acceptable level of service, the mobile hotspot operator should aim to provide the advertised maximum data rate (in each service class) to every user on-board. As the mobile hotspot does not have knowledge of the resource vectors in advance, it is virtually impossible for the mobile hotspot operator to guarantee that each user will get their advertised data rate during the entire trip.

An interesting issue will be on the admission control of both schemes. Admission control is performed at each station to determine the number of passengers admitted onto the mobile hotspot, to ensure there are sufficient resources to serve the group of users who are admitted onto the mobile hotspot. The admission vector $\mathbb{A}_s$ represents the total number of users admitted in each service class. To measure the performance of various admission control schemes, we introduce two metrics: *service disruption probability*, and *admission blocking probability*. Firstly, a *service disruption* occurs when a passenger's data traffic cannot be mapped to any one of the mobile hotspot's access network connections, due to insufficient resources. Secondly, an *admission block* occurs when a passenger's request for data service is rejected by the admission control scheme. The mathematical definitions of these two metrics are:

$$\beta_k = \sum_{s=1}^{S}(a_s - \sum_{n=1}^{N}(x_{s,n,k})) \tag{6.4}$$

$$\gamma = \frac{\sum_{k=1}^{K} \beta_k}{K \sum_{s=1}^{S} a_s} \tag{6.5}$$

$$\delta = 1 - \frac{\sum_{s=1}^{S} a_s}{\sum_{s=1}^{S} r_s} \tag{6.6}$$

where β_k represents the number of service disruptions in variation point k; γ is the service disruption probability; δ is the admission blocking probability; a_s is the number of service class s users admitted; and $x_{s,n,k}$ represents the number of service class s users mapped to network n at variation point k.

6.4 Profit maximization schemes

The problem of profit maximization with given resource and request knowledge is a classical Linear Integer Programming (LIP) problem. Hence, the proposed algorithms presented in this chapter will be heavily based on LIPs. The difference between the algorithms lies in the way the computations are performed, and the constraints imposed on the models.

We have identified two schemes in computing the traffic distribution. In the *in-transit* computation scheme, the MR computes the on-the-fly traffic distribution decisions at each variation point. This scheme is suitable in situations where the coverage and resource variations can only be detected when the mobile hotspot physically reaches the variation point. In contrast, the *pre-transit* computation scheme computes a *set* of traffic distribution decisions at each station, for all the variation points in the segment. This scheme assumes the mobile hotspot to have a prior knowledge on the coverage and resource variations. The resource vectors can be obtained either using resource discovery [60] and/or resource reservation [61–63] protocols; or to predict the network variations by using data mining techniques on data recorded in past trips. The *pre-transit* scheme is only realistic for PTVs as they repeatedly traverse the same route.

6.4.1 In-transit computation

The *in-transit* profit maximization algorithm (*in-transit-PM*) computes a traffic distribution vector which aims to maximize the estimated profit at each variation point. As the *in-transit* scheme assumes the mobile hotspot to have knowledge on the resource vector on arriving at each variation point, the proposed algorithm requires the mobile hotspot to compute the following LIP model at every variation point to obtain the traffic distribution vector $\mathbb{X}_{s,n}$:

$$\max \sum_{s=1}^{S} \sum_{n=1}^{N} \alpha_{s,n}.x_{s,n} \tag{6.7}$$

subject to

$$\sum_{s=1}^{S} (x_{s,n}.d_s) \leq b_n \ , \forall n = 1, \ldots, N \tag{6.8}$$

$$\sum_{n=1}^{N} x_{s,n} \leq r_s \ , \forall s = 1, \ldots, S \tag{6.9}$$

where S is the number of service class; N is the number of networks; $\alpha_{s,n}$ represents the profit function; $x_{s,n}$ represents the number of service class s users mapped to network n at current variation point; d_s represents the service data rate; b_n represents the available bandwidth for network n; and r_s represents the number of service class s requests at each station.

The mobile hotspot can effectively control the service disruption probability by enforcing an admission control policy based on the resource vector in the first variation point. The proposed admission control extension (*in-transit-PM-AC*) requires the mobile hotspot to compute the *in-transit* LIP model (6.7 - 6.9) at the first variation point. In subsequent variation points, the MR will compute the same LIP model with constraint (6.9) replaced with:

$$\sum_{n=1}^{N} x_{s,n} \leq a_s \ , \forall s = 1, \ldots, S \tag{6.10}$$

where a_s is the admission vector (i.e. the $\mathbb{X}_{s,n}$ computed at the first variation point).

6.4.2 Pre-transit computation

The *pre-transit* profit maximization algorithm (*pre-transit-PM*) computes a traffic distribution plan which aims to maximize the total profit in each segment. As the *pre-transit* computation assumes the mobile hotspot to have prior knowledge on the resource vectors in the segment, the mobile hotspot will perform the computation once all the users have boarded the vehicle. To obtain the traffic distribution plan; i.e. the set of $\mathbb{X}_{s,n,k}$; the mobile hotspot will compute the following LIP at each station:

$$\max \sum_{k=1}^{K} \sum_{n=1}^{N} \sum_{s=1}^{S} \alpha_{s,n} . x_{s,n,k} \tag{6.11}$$

subject to

$$\sum_{s=1}^{S} (x_{s,n,k} . d_s) \leq b_{n,k} \tag{6.12}$$

$$\forall n = 1, \ldots, N, \forall k = 1, \ldots, K$$

$$\sum_{n=1}^{N} x_{s,n,k} \leq r_s \tag{6.13}$$

$$\forall s = 1, \ldots, S, \forall k = 1, \ldots, K$$

where S is the number of service class; N is the number of networks; K is the number of variation points; $\alpha_{s,n}$ represents the profit function; $x_{s,n,k}$ represents the number of service class s users mapped to network n at variation point k; d_s represents the service data rate; b_n represents the available bandwidth for network n; and r_s represents the number of service class s requests at each station.

The admission control policy can be performed by taking into consideration all the bandwidth vectors in the computation. This allows the mobile hotspot operator to eliminate service disruptions, by enforcing a strict admission control policy in each segment. This can be achieved by adding the following constraint to the previous LIP:

$$\sum_{n=1}^{N} x_{s,n,k} = \sum_{n=1}^{N} x_{s,n,k+1} \tag{6.14}$$

$$\forall s = 1, \ldots, S, \forall k = 1, \ldots, K-1$$

This constraint basically eliminates service disruptions by restricting the sum of the admitted user requests to be equal at every variation point of the segment. We call this the *pre-transit* profit maximization with zero-service-disruption algorithm (*pre-transit-PM-ZSD*).

6.5 Simulation results and analysis

6.5.1 Simulation model

We implemented the proposed traffic distribution algorithms in C++, which computes the traffic distribution vectors based on a given system model as described in this chapter. The implementation is linked to a commercial LIP optimization solver called CPLEX [64]. We performed a set of experiments which investigated the effect of increasing user demand on the profit and admission control performance dynamics of the mobile hotspot. Our simulated system model consists of five access networks, in which the network and service charges were fixed in all experiments. For each network, resource variation was drawn from a uniform random distribution with thirty two steps between zero and 10MBps. For each experiment, we conducted many simulation runs and reported the average value to achieve acceptable confidence levels.

6.5.2 Profit performance

In Figure 6.2, we can see the *in-transit-PM* and *pre-transit-PM* computations have identical profit performance, which shows the knowledge of the resource vectors in the *pre-transit* scheme does not contribute any additional profit to the system. This is because the objective function is not dependent on future resource vectors. In the admission control extensions, we see that both algorithms also have identical profit performance when there are sufficient networks (resources) available to the mobile

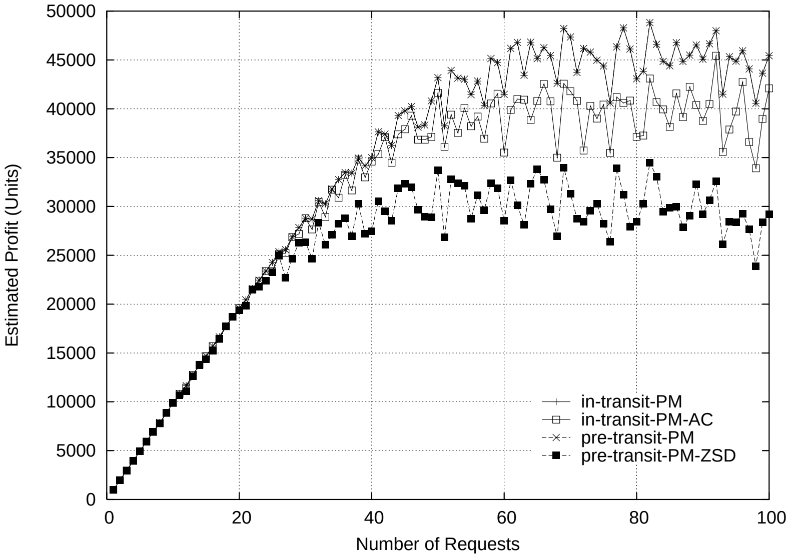

Figure 6.2: Number of requests vs estimated profit

hotspot. However, as the number of requests increases to a level (i.e. 30 requests) where there are insufficient resources to serve all of the requests, the *in-transit-PM-AC* computation out-performs the *pre-transit-PM-ZSD* algorithm. This behavior is due to the different admission control capabilities of both computation schemes. In cases where the number of user requests exceed the amount of available resource, the *pre-transit-PM-ZSD* computation's admission control will admit less passengers than the *in-transit-PM-AC* computation, since it needs to enforce a tighter constraint to ensure the service disruption probability will be zero in all its computations. As a result, the *in-transit* computation can achieve higher profit performance as it admits more users than the *pre-transit* computation.

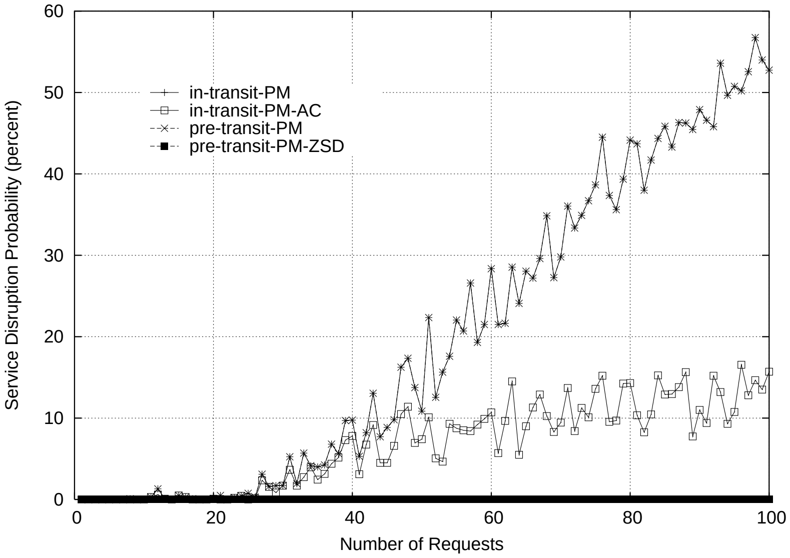

Figure 6.3: Number of requests vs service disruption probability

6.5.3 Service disruption

Next, we measured the service disruption probability for both computations. For this metric, we will not discuss the *pre-transit-PM-ZSD* algorithm, since it is able to enforce zero service disruption probability in all computations. In Figure 6.3, we can see the *in-transit-PM* and *pre-transit-PM* algorithms have identical service disruption probability throughout the experiments, as both of them do not attempt to provide any kind of admission control in their computation. The *in-transit-PM-AC* computation is able to provide zero service disruption probability, in cases where there are sufficient network resources available to the mobile hotspot. As the number of user requests increases, the service disruption probability increases. But the increase seems to be no-linear and there appears to be an upper bound (i.e. 15%) on the

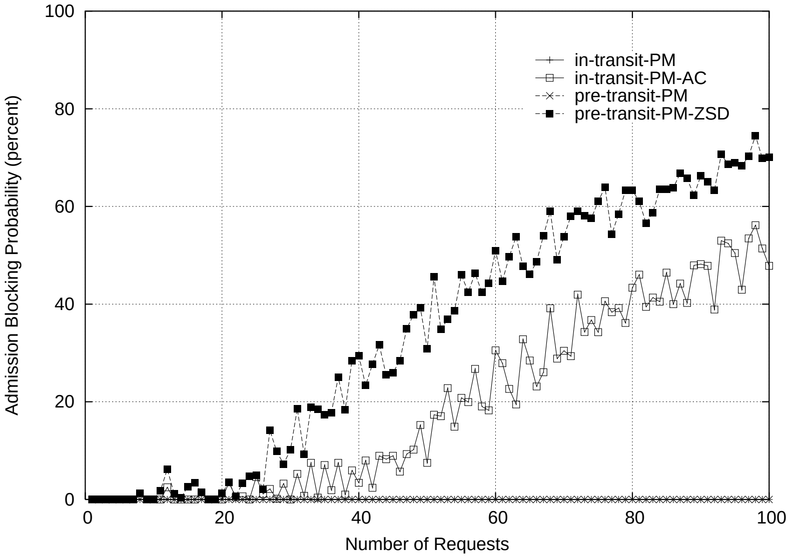

Figure 6.4: Number of requests vs admission blocking probability

service disruption probability. This is due to admission control exhibited by the *in-transit* computation at the first decision point. The significance of this finding is that it allows the *in-transit* computation to provide passengers with an expected level of service disruption. Consequently, the mobile hotspot can try to balance the trade-off between the increased profit shown in Figure 6.2, with the bounded service disruption probability shown in Figure 6.3.

6.5.4 Admission blocking

Finally, Figure 6.4 shows the admission blocking probability for both computations. As expected, the *in-transit-PM* and *pre-transit-PM* algorithms will have a zero admission blocking probability as both will accept all user requests. The *pre-transit-PM-ZSD* algorithm has a higher admission blocking probability than the *in-transit-PM-AC* algorithm, due to the tighter admission control imposed by the computation.

Another interesting trend is that both the *in-transit-PM-AC* and *pre-transit-PM-ZSD* computation have similar admission blocking probability gradients as the number of user requests increases. The difference between the values in these admission blocking probability functions, directly contributes to the difference in profit performance and service disruption probability discussed previously.

6.6 Summary

We have formulated a public transport mobile hotspot model to study the performance of two proposed user-based traffic engineering schemes. The performance dynamics of these schemes in terms of profit and service disruption were studied through simulations. There were three major contributions in this chapter:

- First, we presented the design of a new mobile hotspot model for public transports, and formulated an optimization problem which aims to maximize the profit of the mobile hotspot operator in the volume-based charging model, while satisfying the users' bandwidth requirements.

- Secondly, we proposed the *in-transit* and *pre-transit* user-based traffic engineering schemes, which aim to optimize both the profit and performance of the mobile hotspot system.

- Thirdly, we performed extensive simulations to highlight the performance and issues of the proposed algorithms, and showed that algorithms in the *pre-transit* scheme does not provide additional profit to the system, but instead allows the mobile hotspot to limit the service disruption probability for the users. Service disruptions can be completely avoided at the expense of lower profit, if all network changes are known a priori at the start of the trip by exercising strict admission control.

Chapter 7

Inbound traffic control

In this chapter, we present the design of the Multi-Homed Downlink Control Protocol (MH-DCP). The protocol allows the MR to exhibit traffic engineering control over the on-board users' downlink data traffic, under the flow-based and user-based traffic engineering schemes in the NEMO multi-homed architecture.

7.1 Introduction

The bi-directional tunneling of the users' data packets across the wireless access links, allows the NEMO multi-homed network model to independently switch the on-board users' uplink and downlink data traffic over the wireless access networks. Packets from the same data flow can be sent over different uplink and downlink paths without breaking the flow's end-to-end semantics. This opens up the opportunity for the MR to exhibit traffic engineering control for the on-board users' downlink data traffic. There are two main reasons why the mobile hotspot would like to exhibit traffic engineering control over the users' downlink data traffic:

- *Link asymmetry* - Most of the currently available wireless access networks (e.g.

GPRS, Third Generation Mobile Communication System (3G)) possess disparity in the uplink and downlink bandwidth. Since the performance metrics such as system utilization and flow fairness are directly related to the bandwidth of the access links, the system may need to perform independent uplink and downlink traffic engineering control [65–67] in order to maximize these performance metrics. For example, if the system implements the *MaxUtility* flow scheduler, the MR may need to maximize two independent system utility indices for the uplink and downlink data traffic.

- *Flow asymmetry* - In a lot of current user applications, the data traffic flows are asymmetric in terms of the amount of data sent in the uplink and downlink direction [68]. Applications which exhibit this characteristic include web surfing, file transfer, and one-way multi-media streaming etc. The throughput difference in each direction means that mapping the imbalanced uplink and downlink data in the same access network may result in good switching performance in one direction, and poor switching performance in the other. Therefore, the system may need to tailor its traffic engineering solutions independently for the uplink and downlink direction.

To exhibit downlink traffic engineering control, the MR needs to send its traffic engineering instructions to the MRHA so that the MRHA can switch the on-board users' downlink data traffic accordingly. To support this, we proposed a new signaling protocol named Multi-Homed Data Control Protocol (MH-DCP) that allows the MR to control the on-board users' downlink data traffic in the NEMO multi-homed architecture. In this chapter, we present the design of the signaling messages and discuss the signaling procedures for the different operation scenarios that will arise when the MR performs downlink traffic engineering control.

The proposed MH-DCP protocol supports both the user-based and flow-based traffic engineering schemes. It was not designed to be used with the packet-based traffic engineering scheme since it will be highly un-scalable for the protocol to send downlink control messages on a per-packet basis. As the traffic engineering solutions proposed in this thesis were designed to be independent of the data traffic direction, the MH-DCP protocol allows the MR to apply any of the user-based and flow-based traffic engineering algorithms that were proposed in this thesis, to control the on-board users' downlink data traffic.

Since the MH-DCP protocol supports both the user-based and flow-based traffic engineering schemes, we will use the term *session* to describe either a flow or user as the downlink control unit in the MH-DCP protocol. The term *session switching table* will also be used to represent either the user switching table or the flow switching table according to the context of the traffic engineering scheme discussed.

The outline for the rest of this chapter is as follows. In Section 7.2, we present the design details of the various message formats used in the MH-DCP protocol. In Section 7.3, we provide an in-depth discussion on the MH-DCP signaling procedures for the various operation scenarios that will arise when the MR performs downlink traffic engineering control. Finally, in Section 7.4, we conclude this chapter with a summary of its contributions.

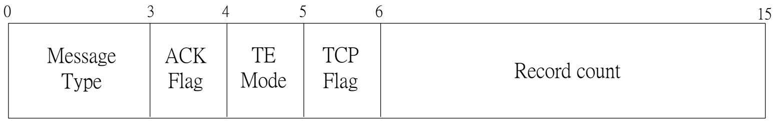

Figure 7.1: MH-DCP message header

7.2 Message design

The general message header for the MH-DCP protocol is illustrated in Figure 7.1. All the signaling messages used in the MH-DCP protocol share this common header format. The MH-DCP message header contains 4 main fields: *message type* field, *traffic engineering mode* (TE-mode) field, *TCP-flag* field, *acknowledgment flag* (ACK-flag) field, and *record count* field.

The message type field stores a numerical message ID that is used to identify one of several message types used in the MH-DCP protocol. The 4 bits field supports the definition of 16 different message types in the MH-DCP protocol. At the moment, there are only 6 message types defined for the current version of the MH-DCP protocol, and the mappings between the message ID and their corresponding message types are defined in Table 7.1. Details on the usage of each of these messages will be discussed in the protocol operation section.

The TE-mode field is used to identify whether the user-based or the flow-based traffic engineering scheme is used. Table 7.2 provides a list of all the traffic engineering mode mappings that are defined in the current version of the MH-DCP protocol. The TE-mode field is also used to assist the decoding of the session records in the message payload. For example, as we will discuss in the next section, both the session initialization and session re-schedule messages utilized *session-to-link* records to

Message ID	Message type
0	Acknowledgement
1	Session initialization
2	Session re-schedule
3	Session termination
4	Downlink update
5	Session table update
6-15	Unused

Table 7.1: MH-DCP message ID field types

Traffic engineering ID	Traffic engineering scheme
0	User-based scheme
1	Flow-based scheme

Table 7.2: MH-DCP traffic engineering modes

control the downlink traffic on a per-session basis. Depending on the traffic engineering scheme used, the session-to-link records are different in size. Hence, the MRHA must read the TE-mode field in the MH-DCP message header before it can read the session-to-link records correctly.

For the user-based traffic engineering scheme, the session-to-link record contains a mapping between an on-board user IP, and the IP address and port number of the access link in which the session is mapped to. Note that the MH-DCP protocol utilizes 128 bits IPv6 addresses, which results in a session-to-link record size of 272 bits (or 34 bytes). The session-to-link record format for the user-based traffic engineering scheme is shown in Figure 7.2.

For the flow-based traffic engineering scheme, the MR needs to write the entire $flowID$; i.e. the flow's source IP, source port number, destination IP, and destination

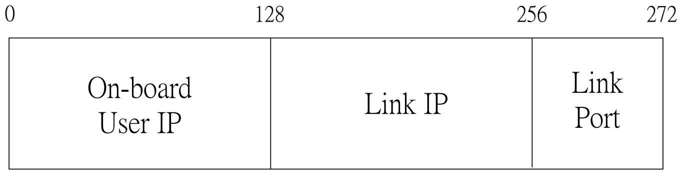

Figure 7.2: MH-DCP session-to-link record for user-based traffic engineering scheme

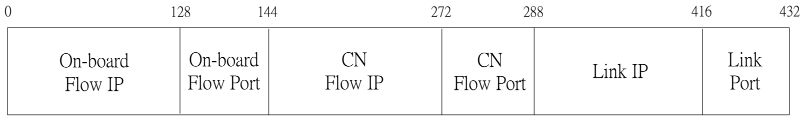

Figure 7.3: MH-DCP session-to-link record for flow-based traffic engineering scheme

port number; in the record which results in a total session-to-link record size of 432 bits (or 54 bytes). The session-to-link record format for the flow-based traffic engineering scheme is shown in Figure 7.3.

The MH-DCP signaling protocol uses User Datagram Protocol (UDP) as its default transport layer protocol. The system has the option of using TCP by setting the TCP-flag in the message header. For the highly adaptive flow-based traffic engineering solutions such as the *MaxUtility-R* and *MaxUtility-PR* flow schedulers; which maximizes the system utility index by regularly re-scheduling the existing flows to different access links; the recommendation is to use UDP as the underlying transport layer protocol. The advantage with using UDP is that the protocol does not require any connection establishment and termination procedures. This makes it more suitable for the highly dynamic traffic engineering solutions such as the *MaxUtility-R* and *MaxUtility-PR* flow schedulers, which will utilize a lot of MH-DCP messages to update the session switching tables in the MRHA.

The limitation with UDP is that data reliability is not implemented. For the flow-based traffic engineering solutions, MH-DCP message reliability may not be a crucial factor since losing MH-DCP messages will only result in sub-optimal downlink traffic performance for a relatively short period of time. In contrast, the less dynamic solutions such as the *in-transit* and *pre-transit* user-based traffic engineering schemes may choose to use TCP as its underlying transport layer protocol, since TCP provides full reliability support that ensures the MH-DCP messages arrive at the MRHA as intended. The safe delivery of the MH-DCP messages is important in this case, since the coarse switching granularity of the user-based traffic engineering scheme may magnify the effects (and the duration) of degraded traffic switching performance as a result of lost MH-DCP messages.

If the MR uses UDP as its underlying transport layer protocol, it has an additional option of enabling an acknowledgment mode that provides a very basic form of MH-DCP message reliability. The acknowledge mode only supports one-way message reliability for the MH-DCP messages sent from the MR, since the MRHA does not send any traffic engineering control messages to the MR. To enable the acknowledgment mode, the MR will need to set the ACK-flag in the MH-DCP message header for all the messages it wished to be acknowledged by the MRHA.

The record count field is used for *record aggregation* purposes. Record aggregation allows the MH-DCP protocol to transfer multiple message records inside one MH-DCP message. The record count field stores the number of records that are transferred in this message, and 10 bits are used to support the aggregation of 1024 records in each MH-DCP message.

Finally, MH-DCP messages has the option to append message specific data in

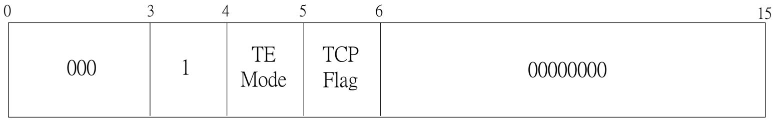

Figure 7.4: MH-DCP acknowledgment message

the *message payload* of the MH-DCP message. Apart from the acknowledgement message, all the other message types defined in this version of the MH-DCP utilizes the message payload. The message payload is appended after the message header, in which the format and size of the message payload will differ between the different message types and the TE-mode used.

7.3 Protocol operation

In this section, we will provide a detailed discussion on the signaling process for the different operation scenarios in the MH-DCP protocol. We will start off our discussion on the protocol operation of the MH-DCP acknowledgment mode, which provides one-way message reliability for the messages sent by the MR to the MRHA. We then discuss the signaling procedures required for the downlink session control during various stages of the session lifetime. Finally, we will discuss the signaling procedures involved with aggregate downlink traffic engineering control.

7.3.1 Message acknowledgement

To enable the acknowledgement mode, the MR needs to set the ACK-flag in the message header of all the MH-DCP messages it sends to the MRHA. Upon receiving the

MH-DCP messages with the ACK-flag set, the MRHA will need to send an acknowledgement message back to the MR. The format of the MH-DCP acknowledgement message is shown in Fig 7.4.

By enabling the acknowledgement mode setting in the MH-DCP message header, the MR has the option to selectively choose which MH-DCP message requires to be acknowledged by the MRHA. This allow the MR to prioritize the delivery of certain control messages. For example, if the MR only wish to change the downlink switching table for a single session, it may disable the ACK-flag in the message header since the lost of the MH-DCP message may only have minimal effects on the system performance. On the other hand, if the MH-DCP message have significant effects on the user downlink traffic; i.e. updating the entire downlink session switching table; it may decide to switch the acknowledgement mode on for this message.

If the ACK-flag is set for a particular MH-DCP message, the MR will need to start a timeout timer after sending the message over one of its wireless access links. If the MR does not receive an acknowledgement message from the MRHA within a timeout period, the MR will need to re-send the previous message to the MRHA. The message re-transmissions may be triggered by two possible cases. Firstly, if the original MH-DCP message was lost, the re-transmitted message will allow the MRHA to perform the instructions that were specified in the lost MH-DCP message from the MR accordingly. In the second case, if the MH-DCP acknowledgement message sent by the MRHA was lost, the MRHA will receive a duplicate of the previous MH-DCP message. In this case, the duplicate message will not affect the downlink traffic control in the MRHA since the MRHA had already processed the instructions in

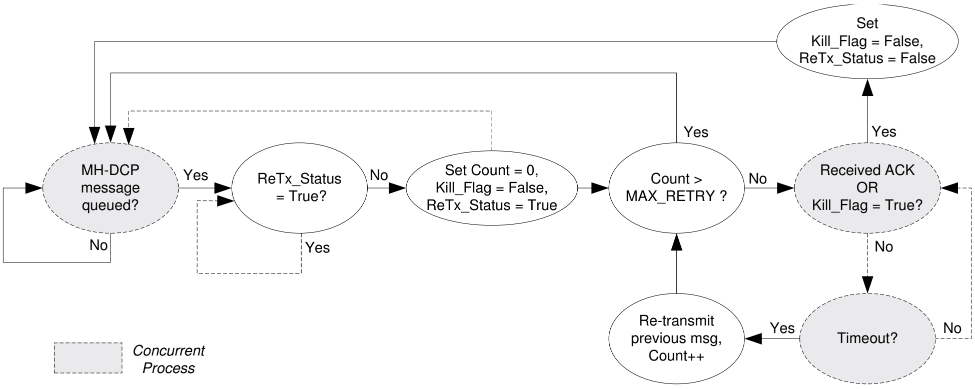

Figure 7.5: State diagram of the MH-DCP message acknowledgment mechanisms

the previous message. Hence, even if the MRHA does not recognized that this re-transmitted message is a duplicate of the previous message, the processing of the duplicate message will not lead to any actions taken by the MRHA.

To prevent the MR in re-transmitting excessive number of MH-DCP messages to the MRHA; which can occur if the re-transmitted MH-DCP messages or the re-transmitted acknowledgement messages are lost continuously; the MR may need to restrict itself to re-transmitting only a certain number of the same MH-DCP message to the MRHA. To achieve this, the MR may need to maintain a *re-transmission counter* for every re-transmission messages it sends to the MRHA. Furthermore, if there is a new MH-DCP message scheduled in the MR during the re-scheduling process, the MR should give strict priority to the new message by giving up the re-transmission process immediately. This ensures that the MRHA does not process out-dated MH-DCP messages caused by the message re-transmission process. Figure 7.5 provides a full state diagram that captures the details of all the acknowledgement features discussed.

7.3.2 Downlink session control

In this section, we provide a detail discussion on the signaling process for the downlink data traffic control during various stages of a session lifetime. To provide a logical view on the use of the MH-DCP protocol operation for downlink traffic engineering control, we divide the process into three stages:

1. *Session initialization* - For all new sessions, the MR and MRHA needs to decide which wireless access network the new session should be mapped to in the uplink and downlink direction. Once the MR makes a decision on this, it will need to initialize its own uplink session switching table and communicate the downlink control instructions to the MRHA, so that the MRHA can update its downlink session switching table accordingly.

2. *Session re-schedule* - While the mobile hotspot is in transit, the control module may wish to re-schedule a data session from one access network to another. To re-schedule a session's downlink traffic, the MR needs to inform the MRHA so that it can modify its session switching table accordingly.

3. *Session termination* - When a session terminates, the MR and MRHA should remove the session record in the uplink and downlink session switching tables. This ensures that the session switching tables only contain records for the active sessions, which will reduce the session lookup times for all incoming uplink and downlink data traffic.

We will now discuss the protocol operation for each of these stages in detail.

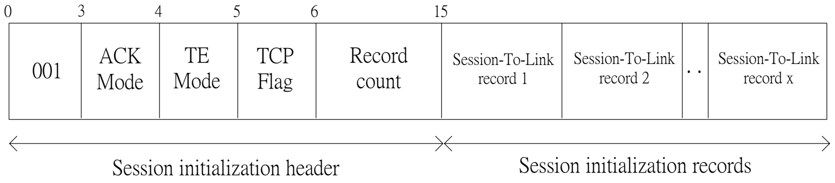

Figure 7.6: MH-DCP session initialization message format

Session initialization

In the session initialization phase, there are two different ways in which the sessions can be established. For the flow-based traffic engineering scheme, the MR will trigger its flow assignment algorithm once the MR detects the arrival of a new flow. In contrast, the user-based traffic engineering scheme assigns users to the appropriate networks when the users board the PTV and indicate their service preference. In both scenarios, the MR will need to decide which downlink access link the new session will be mapped to. Once a decision is made, the MR will need to send a MH-DCP session initialization message to the MRHA, which contains the session-to-link records (i.e. Figure 7.2 and 7.3) that needs to be created in the MRHAs downlink session switching table. The message format for the MH-DCP session initialization message is shown in Figure 7.6.

The MR has the option to send a session initialization message for either a single session or multiple sessions. The number of session-to-link records can be indicated in the *record count* field in the MH-DCP header. Upon receiving the MH-DCP session initialization message, the MRHA will first read the TE-mode field in the MH-DCP header to determine the correct session-to-link record format. It will then iterate through the message payload according to the record count field and extract the

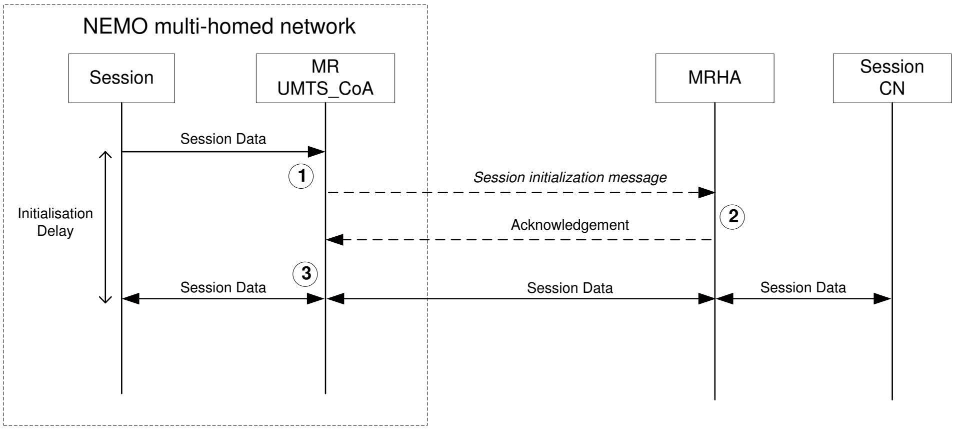

Figure 7.7: MH-DCP session initialization process with acknowledgement mode enabled

session-to-link mapping accordingly. For each record, the MRHA will create the appropriate entries in its session switching table so that subsequent data packets from these sessions will be tunneled to the corresponding IP address and port numbers of the access link.

If the acknowledgement mode is enabled in the session initialization phase, the MR can utilize the acknowledgement message to probe the wireless access link before forwarding the session's data traffic over the target access link. This allows the MR to ensure that the target access link is available before mapping a new session to it. A more advanced approach can even allow the MR to make use of the acknowledgement message(s) to measure the performance characteristics of the target access link, which can be used to assist the MR in its traffic engineering decisions. A limitation with this approach is that it creates an initialization delay for the data session. This process is illustrated in Figure 7.7.

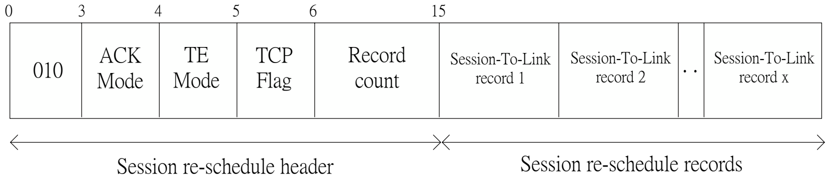

Figure 7.8: MH-DCP session re-schedule message format

Session re-schedule

When the system is in transit, the MR may decide to re-schedule an existing session from one access link to another. If the MR decides to change the switch the session's uplink traffic from one access link to another, the MR will need to modify its own session switching table accordingly. To perform a session re-schedule on the downlink data traffic, the MR needs to inform the MRHA by sending a *session re-schedule* message to the MRHA. The format of the session re-schedule message is similar to the session initialization message, and it is shown in Figure 7.8. When the MRHA receives the session re-schedule message, it will iterate through the session-to-link records and update the corresponding session records in its session switching table accordingly.

The MH-DCP protocol provides a highly flexible framework for the MR to exhibit downlink traffic control for the on-board users. Hence, there are various ways in which the system can make use of the MH-DCP protocol to perform session re-schedule. To illustrate how this can be done, we look at two types of session re-schedules scenarios. Firstly, a *soft* session re-schedule occurs when both access links are still available for data transfer during the session re-schedule, so that the data packets will continue to be sent over the original link until the MR and MRHA re-directs the data packets to

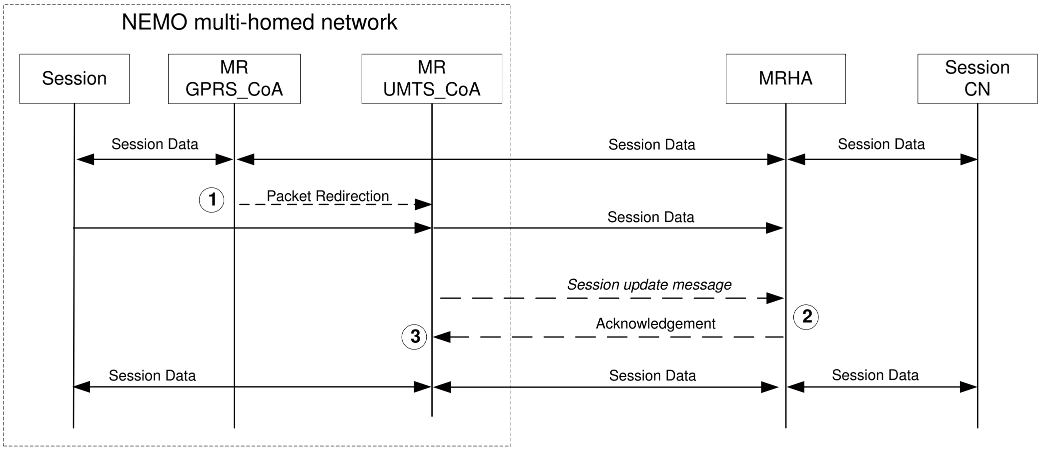

Figure 7.9: MH-DCP soft session re-schedule signaling procedure

the new interface. This usually occurs when the mobile hotspot decides to make a session re-schedule for the sake of improving the system performance; i.e. the system utility index. The second type of session re-schedule is the *hard* session re-schedule, which occurs when the original link cannot be used during the session re-schedule. This usually occurs in a more critical session re-schedule, where the mobile hotspot must switch the passenger's traffic over to a new interface since the original interface has gone down, due to congestion or poor signal strength etc. To illustrate a complete example of the signaling procedures for these two types of session re-schedules, we assume that the MH-DCP acknowledgement mode is enabled.

Figure 7.9 illustrates the signaling involved with a soft session re-schedule. Basically, there are three major steps in this procedure:

1. The mobile hotspot system triggers a session re-schedule for session A. The MR

sends a session re-schedule message to the MRHA, specifying the new session-to-link record for the session and the details of the Universal Mobile Telecommunications System (UMTS) link. The uplink and downlink data continues to be sent over the original GPRS interface.

2. The MRHA receives the session re-schedule message from MR, and modifies its session switching table by updating the access link details for record of session A. The MRHA sends an acknowledgment back to the MR, and starts re-directing the data packets destined for session A to the current CoA for the UMTS link. The downlink traffic is now being sent over the UMTS network.

3. Once the MR receives the acknowledgment message, it performs a session re-schedule for session A so that all subsequent outgoing data packets from session A are forwarded to the UMTS interface.

4. All the data transfer between session A and the CN are now being sent over the UMTS link.

The signaling involved with a hard session re-schedule is a bit more complicated, and is shown in Figure 7.10. Basically, there are four major steps in this procedure:

1. The MR detects a failure in the GPRS link, and triggers a session re-schedule for session A. The MR sends a session re-schedule message to the MRHA over the UMTS interface, containing the session-to-link record for session A and the UMTS link. The data transfer for session A is temporarily disrupted since the traffic cannot be routed over the original GPRS interface.

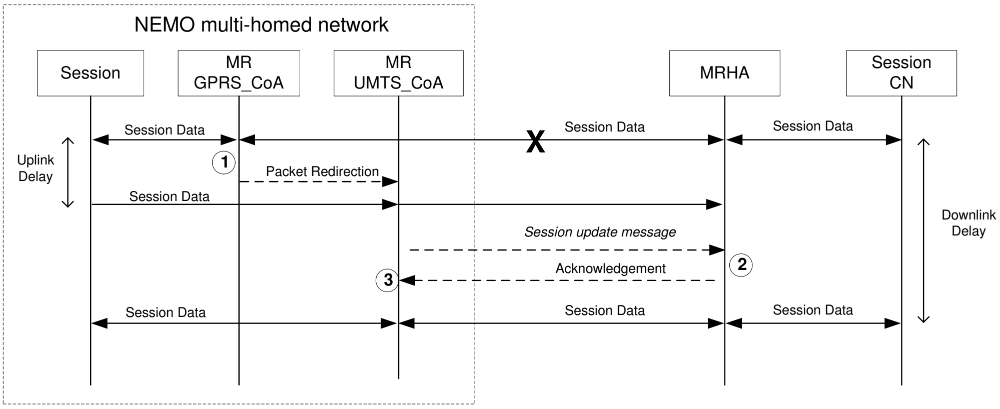

Figure 7.10: MH-DCP hard session re-schedule signaling procedure

2. The MRHA receives this signal, and modifies its session switching table accordingly. The MRHA sends an acknowledgment message back to the MR, and starts re-directing the data packets destined for session A to the UMTS link.

3. Once the MR receives the acknowledgment message, it modifies its session switching table for session A so that all subsequent data packets from session A are now forwarded over the UMTS interface.

4. All the data transfer between User A and their corresponding nodes are now sent over the UMTS link.

Session termination

At the end of a session's lifetime, the MR and MRHA needs to remove the corresponding entry from their session switching table. This is particularly important if the system is using the flow-based traffic engineering scheme, since the system can potentially be switching from hundreds to thousands of flows concurrently, and removing the unnecessary records from the flow switching tables will reduce the session

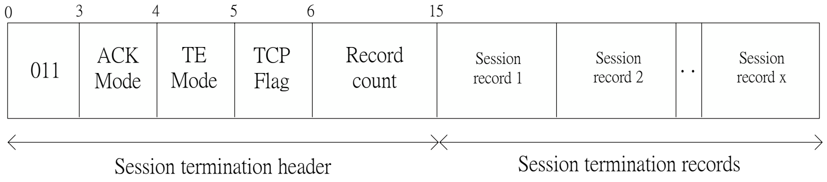

Figure 7.11: MH-DCP session termination message format

switching lookup time for each data packet.

Once a session has finished data transfer, the MR will need to inform the MRHA about the session termination so that it can remove the corresponding entry from its traffic distribution table. To do this, the MR needs to send a *session termination* message to the MRHA. In contrast to the session initialization and session re-schedule messages, the session termination message only needs to contain the terminated session ID, which will be used by the MRHA to remove the corresponding entry in its session switching table. Depending on the TE-mode value in the message header, the session ID can either contain 128 bits for the user ID in the user-based traffic engineering scheme, or 288 bits for the $flowID$ in the flow-based traffic engineering scheme. Again, the MR can terminate multiple sessions by writing the number of termination records in the record count field of the message header. The session termination message format is illustrated in Figure 7.11.

7.3.3 Aggregate downlink control

There may be cases where the MR may wish to perform downlink traffic engineering control on an aggregate level. The MH-DCP protocol provides aggregate downlink control support by introducing two new MH-DCP procedures called the *downlink* *update* and the *session table update* procedure. The downlink update procedure allows the MR to control the aggregate downlink traffic on a per-link basis, while the table update procedure allows the MR to replace the entire session switching table in the MRHA. We will now discuss these two processes in detail.

Downlink update process

There may be situations where the MR may wish to modify the details of a particular access link in the MRHA session switching table. This is called the *downlink update* process, which requires the MRHA to change all the entries that are mapped to that link in its session switching table. For example, the system may wish to perform a downlink update when the system detects that a particular access link has gone down. In this case, the MR may wish to inform the MRHA to re-schedule all the existing flows that were mapped to the inactive link, to another access link attached to the system. Another example scenario which requires a downlink update occurs when an access network attached to the MR changes its CoA. Even though the standard NEMO protocol takes care of the underlying mobility management for the MR, the MR will still need to inform the MRHA so that it can update the link details for all the session switching table entries that were mapped to the old CoA. This procedure is illustrated in Figure 7.12.

Instead of sending a session re-schedule message for all the affected sessions; where all the session-records will be pointing to the new access link details; a more elegant solution will be for the MR to send a dedicated *downlink update* message to the MRHA instead. The downlink update message can reduce the message payload size considerably by specifying a *downlink update* record that only stores the original link details and the new link details. The format of the downlink update record is shown

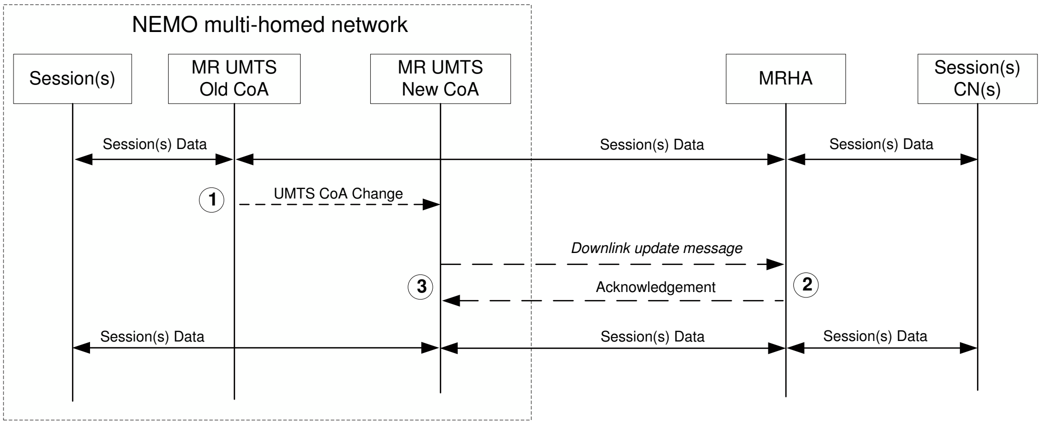

Figure 7.12: MH-DCP downlink update procedure for CoA change in MR

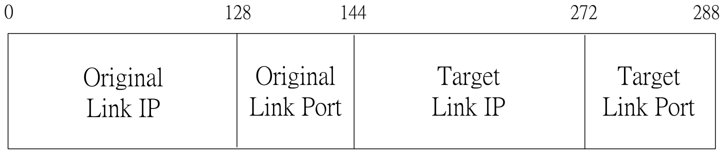

Figure 7.13: MH-DCP downlink update record format

in Figure 7.13.

The MR can also perform multiple downlink updates. When the MR sends the downlink update message to the MR, it can write the number of downlink update records in the record count field of the MH-DCP message header. The format of the downlink update message is shown in Figure 7.14. When the MRHA receives a downlink update message, it will be able to determine the number of downlink updates records in the record count field. The MRHA will then iterate through the downlink update records, and lookup all the entries that were mapped to the old CoA and replace these entries with the new link details.

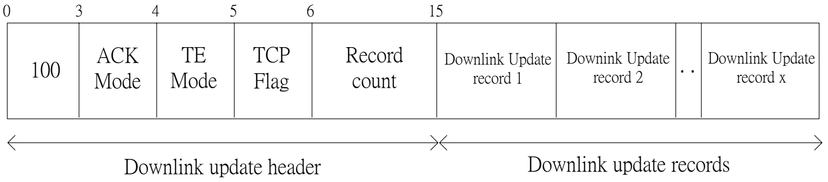

Figure 7.14: MH-DCP downlink update message format

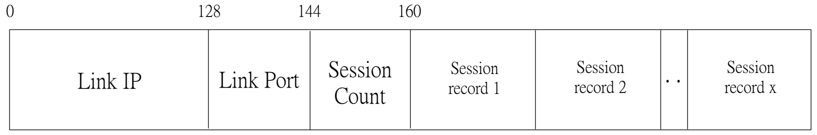

Figure 7.15: MH-DCP link-to-sessions record format

Session table update process

Finally, *session table update* allows the MR to send the complete session switching table to the MRHA in order to replace the entire downlink session switching table in the MRHA. Session table update is particularly useful in the implementation of the *in-transit* and *pre-transit* user-based traffic engineering schemes (as proposed in Chapter 6), since both schemes computes traffic engineering solutions for the entire group of users. The session table update allows the MR to send the entire downlink session switching table to the MRHA. Also, session table update may also be useful in cases where the MR detects (from the incoming data packets) that the MRHA is not forwarding the user downlink data packets over the intended links, due to the lost of MH-DCP messages. In this case, the MR may use session table update to fully synchronize the MRHA session switching table with the its expected downlink session switching table.

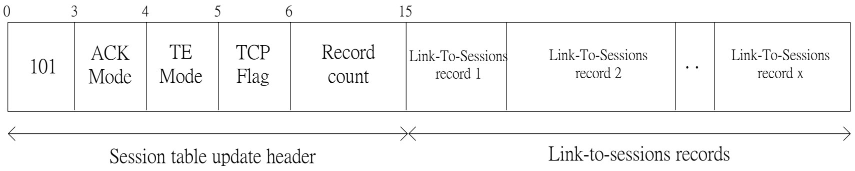
0
3
4
5
6
15
101
ACK Mode
TE Mode
TCP Flag
Record count
Link-To-Sessions record 1
Link-To-Sessions record 2
. .
Link-To-Sessions record x
Session table update header
Link-to-sessions records

Figure 7.16: MH-DCP session table update message format

To perform session table update, the MR needs to send a *session table update message* to the MRHA, which contains a list of *link-to-sessions records* to represent the entire session switching table. Each link-to-sessions record contains a mapping between a particular access link details (i.e. IP and port number), and a list of all the sessions mapped to that particular link. An entire session switching table can be represented by storing the link-to-sessions records for all the access links attached to the MR. The format of the link-to-sessions record and the session table update message are shown in Figure 7.15 and Figure 7.16 respectively. To ensure that session table update is applied correctly, it is recommended that the MR uses TCP to transfer the session table update messages to the MRHA.

7.4 Summary

In this chapter, we presented the design of the newly proposed Multi-Homed Data Control Protocol (MH-DCP), which allows the mobile hotspot to exhibit traffic engineering control over the on-board users' downlink data traffic in the NEMO multihomed architecture. This allows the MR to perform independent uplink and downlink traffic engineering that takes into consideration the asymmetric link and user data traffic characteristics in the uplink and downlink direction. The proposed MH-DCP protocol takes into consideration the session initialization, session re-scheduling, and session termination phases for the user-based and flow-based traffic engineering schemes. By implementing the MH-DCP protocol in the NEMO multi-homed architecture, the MR can use any of the user-based and flow-based traffic engineering solutions proposed in this thesis to control the on-board users' downlink data traffic.

Chapter 8

Conclusion and future work

8.1 Conclusion

In this dissertation, it has been demonstrated that traffic engineering is a critical research issue for the realization of multi-homed mobile networks. This dissertation has provided extensive studies on designing traffic engineering solutions for the NEMO multi-homed network under three different levels of traffic switching granularity. In Chapter 4, it was clearly evident that the fine grain packet-based traffic engineering scheme is not suitable for mobile multi-homed networks, due to the vast number of packet re-transmissions that are caused by the performance disparities and dynamic variations in the wireless access links. In Chapter 5, we proposed a *MaxUtility* flow scheduler that was able to provide better throughput and flow fairness for the on-board users than previously proposed flow schedulers. Extending features such as flow re-scheduling and utilizing link predictions in the *MaxUtility* flow scheduler further enhanced the throughput performance of the system. In Chapter 6, we proposed the *pre-transit* and *in-transit* user-based traffic engineering schemes that were able to maximize the profit for the system while maintaining bandwidth guarantees for the on-board users. Finally, in Chapter 7, we designed a new Multi-Homed Data Control

Protocol (MH-DCP) which allows the MR to exhibit downlink traffic engineering control under the flow-based and user-based traffic engineering schemes.

8.2 Future work

We conclude this dissertation with an enumeration of several remaining future challenges in the design of traffic engineering solutions for the mobile multi-homed networks.

8.2.1 Web traffic analysis

The traffic engineering solutions proposed in this dissertation were not tailor-designed towards any specific user data traffic patterns for the mobile multi-homed network model. As the NEMO multi-homed network aims to provide Internet connectivity to the on-board users, a potential future work may be to extend our proposed traffic engineering solutions to address the specific characteristics of web data traffic models [69–71], which are typically characterized by a mixture of short and long data flows; i.e., the mice and elephant flows as discussed in [46, 47]. For the *MaxUtility* flow scheduler, it may not be necessary to provide fairness for short data flows. Therefore, the *MaxUtility* flow scheduler may implement a flow-size classifier that identifies short and long data flows, and provide differential treatment to the flows according to the flow-size classification [46, 47]. For example, the MR may prioritize the scheduling of short flows, by forwarding these flows to the fastest link in order to improve the overall response time. Whereas for the long flows, the MR may use the proposed flow rescheduling algorithms in order to maximize the fairness and throughput of these flows. Nonetheless, the consideration of user data patterns opens up another interesting traffic engineering research issue for the NEMO multi-homed network platform.

8.2.2 Link prediction algorithms

The results on the utilization of link predictions in the flow-based and user-based traffic engineering solutions presented in this dissertation were obtained with the simplifying assumption that the predicted bandwidth patterns were readily available to the system prior in each trip. Since we did not consider any particular method on how the actual link predictions can be obtained by the mobile network, an important future research issue will be to design link prediction algorithms for the mobile networks. The predictions can be computed by analyzing the historical data for each trip, where patterns are likely to exist since PTVs traverses pre-defined routes repeatedly. In Appendix C, we started some preliminary work on data measurements which investigated the correlation between the GPRS signal strength and various environmental factors such as location, vehicular speed, and humility etc. From the measurements collected in a test-bed that is connected to a live GPRS network, we showed that GPRS signal strength is strongly related to the vehicle's location in each trip. Consequently, this finding may serve as a useful starting point for the research on the design of link prediction algorithms.

As with any form of predictions, we can never guarantee that the predicted results will be 100% accurate. Therefore, another important research issue is to investigate the effects of link prediction inaccuracies on the performance of the various traffic engineering solutions proposed in this dissertation. For example, it will be interesting to see whether we can extend our proposed *pre-transit* user-based traffic engineering scheme proposed in Chapter 6, by considering link prediction inaccuracy probability models in its user admission control scheme. This will allow the mobile hotspot operator to decide on whether to over-provision or under-provision admitted users accordingly.

8.2.3 Prototype implementation

Simulation studies performed in this dissertation only provided an abstract representation of the real-world traffic engineering problem for the mobile multi-homed network model. The purpose of the simulations is to allow us to perform preliminary realization, verification and performance analysis on the proposed traffic engineering solutions. Therefore, a possible future work will be to build a prototype test-bed that serves as a live experimental platform for researchers in the NEMO research community. Since there are no special hardware requirements in building a MR, the MR prototype can be easily implemented using a Personal Computer (PC) or laptop, which is connected to a group of PCs or laptops via wired or wireless technologies to simulate an on-board LAN. The MR prototype can provide Internet connectivity to the on-board LAN via a wireless access network connection that is subscribed to either a single or multiple readily available wireless access networks; e.g. GPRS, 3G, WLAN etc.

The On-board Communication, Entertainment, And Information (OCEAN) research group [72] has started preliminary efforts on this by building a test-bed that deploys their own implementation of the NEMO protocol [73]. This NEMO prototype can be easily extended to include multi-homed support by equipping the MR with multiple network interfaces subscribed to different wireless access networks, which will provide us with a test-bed that allows us to perform live trials on the various traffic engineering solutions proposed in this dissertation. The simulation codes used in our research were specifically written with this in mind, where its strict object-oriented design makes it highly portable to other software modules. By writing a simple customized interface between our simulation code and the system kernel in the MR, the prototype can re-use our simulation code to exhibit traffic engineering control over real user data traffic.

Apart from using it solely as a test-bed, the MR prototype will also provide researchers with the opportunity to collect extensive measurements on various important data such as link performance traces, user data traces, and MR performance metrics for future research and analysis. By deploying a fully functional test-bed on a real PTV, we can conduct extensive trials to collect traces of the users' data traffic patterns and see whether it differs to the data traffic traces collected in other types of network environments; e.g. static hotspots, corporate networks, home networks etc. We expect that the collected data traffic trace may contain notable differences from the other traces due to the unique user dynamics and usage behavior introduced in the new mobile network connection paradigm. Furthermore, the MR prototype allows us to collect traces of the link performance behavior over the course of each trip. If the measurements can be collected over a relatively long period of time, researchers may be able to use data mining techniques to derive patterns among the collected data, which will be useful for the future design of link prediction algorithms.

Part I

Appendix

Appendix A

NS2 simulation details

A.1 Introduction

In this appendix, we will provide the implementation details of the NS2 simulations used in this thesis. The NS2 simulator is a discrete event simulator which is widely used by the research community to evaluate network performances. The NS2 simulations used in the thesis were all performed in version 2.27, which was the latest release at the time of our implementation.

This appendix is not designed to be an introductory tutorial on the NS2, but instead the aim is to highlight the implementation issues for readers who are interested in performing similar simulations for the NEMO multi-homed network model. For a complete description of the NS2 simulator software, please refer to the official NS2 manual [74].

To simulate the performance of the data distribution algorithms for the NEMO multi-homed network model, we created a NS2 simulation topology as shown in Figure A.1. In this simulation architecture, the on-board users and the MR represent the mobile hotspot. The MR is connected to the Internet via the multiple wireless access links, which are represented by the dotted lines between the MR and the various BSs.

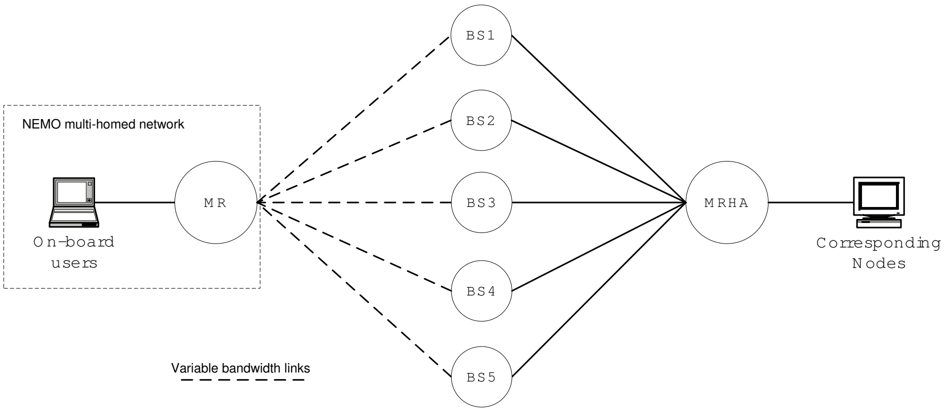

Figure A.1: The general NS2 simulation architecture for the mobile hotspot model

The link dynamics are simulated by varying the link bandwidths and propagation delays during the simulation, and hence we did not implement any wireless physical layer models in the access links. The base-stations, MRHA, and the CNs are all assumed to be located in the Internet and hence they are all connected together via fixed high speed and low delay links. For simplicity, we did not introduce any other external data traffic in any of the links in the simulation model.

We will now discuss the implementation details of the data distribution module, which can be implemented in the MR and MRHA to control the on-board users uplink and downlink data traffic respectively.

A.1.1 Classifier class

To exhibit data routing control in the MR or MRHA, we first need to understand how the packet forwarding works in NS2. Figure A.2 illustrates the data forwarding model in a standard NS2 node. The NS2 node is basically made up of several *classifiers*, where the role of the classifier is to make a decision on how it should handle each

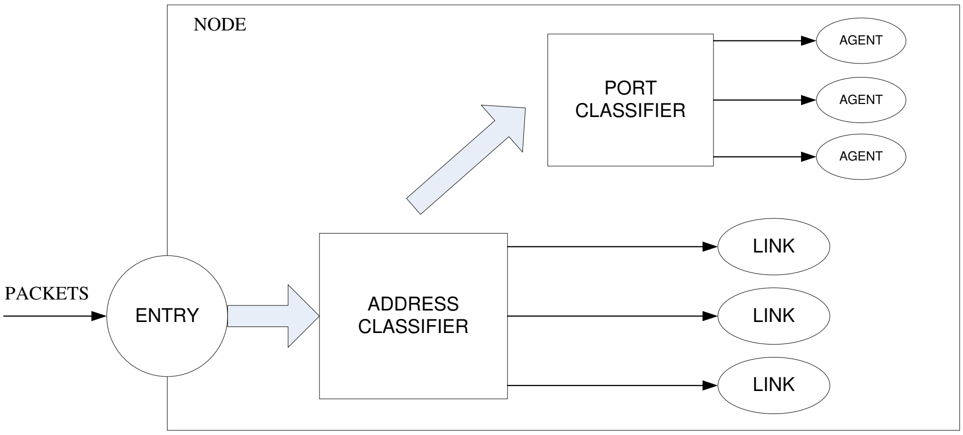

Figure A.2: NS2 node classifier model

packet. When a packet arrives at the node, the packets are first passed to the *address classifier* class, which determines where to forward this packet to, based on the destination address of the packet. If the destination address of the packet is directed at the node itself, it will forward the packet to the *port classifier*, which determines which *agent* to pass the packet to. In NS2, the agents are the communication endpoints where network layer packets are constructed or consumed (e.g. TCP sender and receiver). The agents are identified by the port numbers in a flow. If the packets are destined for other nodes in the networ, it will need to determine which link to forward the packets to. The address classifier does this by looking at its routing table, which is pre-computed at the start of the simulation. The routing table of each node will contain routing entries for every other node in the network, since there is no notion of subnet addressing in the NS2 standard addressing scheme. We did not implement any routing protocols at the nodes, and hence the routing table remained static throughout the simulation.

In the simulation model, we can see that there are multiple paths for the MR and

Slot ID	Object
Slot 0	Port Classifier
Slot 1	Connector (link 1)
Slot 2	Connector (link 2)
Slot 3	Multi-path Classifier 1
Slot 4	Multi-path Classifier 2

Table A.1: Sample slot table mappings for a NS2 node

MRHA to reach each other. For the data packets that are destined for the end-to-end users, the MR or MRHA can forward the packets to either one of the base-stations. The standard NS2 implementation provides limited support for this kind of *multi-path* routing, by computing and storing the multiple paths that can be used to reach the eligible nodes in the routing table. For nodes that can be reached by multiple paths, the routing table entry for these nodes will point to their own copy of a *multi-path classifier* instead of a link object. The multi-path classifier is responsible for storing the multiple paths (link objects) that can be used reach to that particular destination. Figure A.2 illustrates the standard NS2 multi-path classifier model.

In the NEMO multi-homed network model, the MR and MRHA are assumed to have the flexibility to switch the users' data traffic over any of the access links without breaking end-to-end connection semantics. To exhibit control over the selection of the multiple links that are attached to the MR and MRHA, we initially wanted to replace the multi-path classifier with our own multi-path classifier. But there were two reasons why we couldn't do this:

- *Multiple copies of the multi-path classifier objects* - for *each* destination node where multiple paths can be used, there is an independent multi-path classifier associated in each corresponding slot. For the simulations where there are multiple end-users in the model, there will be multiple copies of the multi-path classifier in the MR or MRHA, which means that the data distribution control cannot be centralized. The main drawback with a non-centralized approach is that common information cannot be easily shared between the multiple multi-path classifiers.

- *No direct reference to the multi-path classifier objects* - if we were to exhibit a centralized control over the multiple multi-path classifier objects, it will be logical to do this in the address classifier. But the problem with this approach is that there is no standard function to get hold of each of the multi-path classifier in the address classifier.

Consequently, we had to do ignore the standard NS2 multi-path routing support and we replaced the standard address classifier with our own version of the address classifier; which we call the *mobile router classifier* (MR-classifier). The MR-classifier can be used in both the MR and MRHA nodes for be used respectively for uplink and downlink traffic control.

A.1.2 MR-classifier

The MR-classifier replaces the standard address classifier in the MR and MRHA node. As there are 3 possible data switching schemes for the NEMO multi-homed network model, we implemented abstract classes for the host-based; flow-based; and packet-based MR-classifiers which allow users to implement their own data distribution algorithms in any of these switching schemes. The design of the MR-classifier is shown in Figure A.3.

Every time a packet comes into the MR-classifier, the classify() function is called

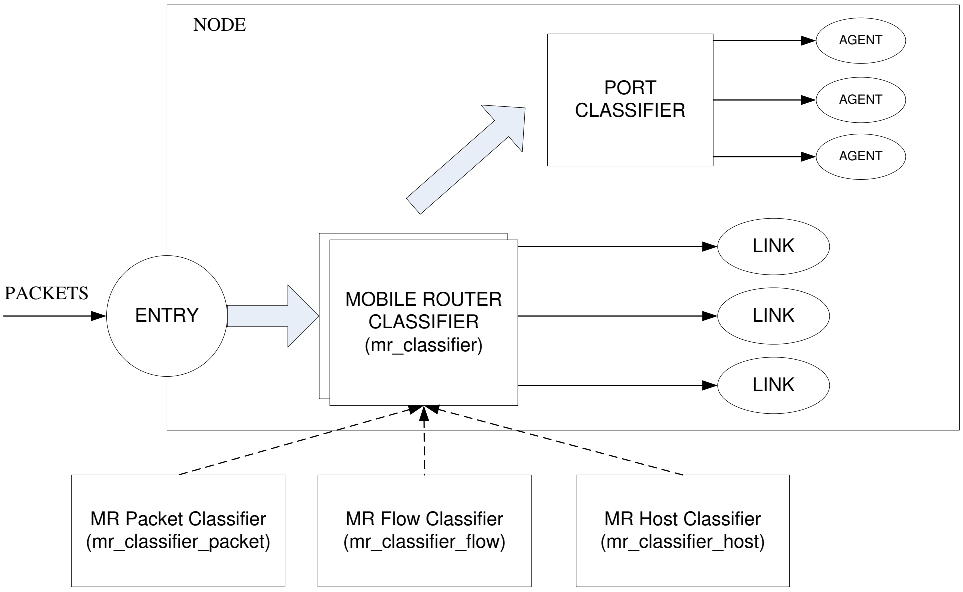
NODE
PORT CLASSIFIER
AGENT
AGENT
AGENT
LINK
LINK
LINK
PACKETS
ENTRY
MOBILE ROUTER CLASSIFIER (mr_classifier)
MR Packet Classifier (mr_classifier_packet)
MR Flow Classifier (mr_classifier_flow)
MR Host Classifier (mr_classifier_host)

Figure A.3: NS2 mobile router classifier model

to determine which *slot number* to forward the packet over to. The slot numbers are basically numerical identifiers to the various objects (i.e. link, port classifier) that are mapped to each destination in the routing table. The main problem with the new address classifier implementation is that it does not have any idea on the simulation topology, which includes the network addresses of the other nodes in the model. As we disabled the built-in multi-path routing support, the routing table of the MR-classifier only contains a single routing entry (slot number) for each destination address and hence there is no way for the MR-classifier to know if there are multiple paths to a particular destination node when the classify() function is called. To get over this limitation, we had to design the classifier in such a way that allows the Tool Command Language (TCL) simulation scripts to explicitly specify the following information before the start of the simulation:

- *the addresses of the end nodes* - Since multi-path support is disabled in the simulation, we need to explicitly tell the MR-classifier the address of the corresponding end nodes which needs explicit traffic engineering control (i.e. determining which access link to forward the packet to). For the MR-classifier in the MR, this means the simulation scripts needs to specify the address of the CNs while for the MR-classifier in the MRHA, the simulation scripts needs to specify the addresses of the on-board users. This allows the MR-classifiers to trigger its data distribution control for packets destined to these special addresses. For the routing of packets to the other nodes, the MR-classifier will simply perform a standard routing lookup and return the slot numbers in the pre-computed static routing table.
- *the address of the base-stations* - this allows the MR-classifier to determine the slot numbers of the link objects for each of its multiple access links. This information allows the MR-classifier to know which link options it has when a packet is destined for the corresponding end users.

With the above information specified at the start of the simulation, the MR-classifier will first look at the destination address of all incoming packets to see if it matches the address of the (specified) end nodes. If the packet belongs to one of the specified end nodes, the MR-classifier will make a call to MR-classify(), which determines which access link the packet should be forwarded to. Once a packet is forwarded to any one of the BS, the BS will be able to forward the packet to the end hosts according to the routing paths that are pre-computed by the standard NS2 implementation.

A.2 Conclusion

This section highlighted some of the important implementation issues in our modified version of the NS2 simulator. It allows users to gain a better understanding on the implementation challenges we have experienced, which hopefully makes it easier for readers who wished to implement their own data distribution algorithms for the NEMO multi-homed architecture.

Appendix B

CPLEX simulation details

B.1 Introduction

In this appendix, we provide the implementation details of the CPLEX simulations that were used in the evaluation of the *in-transit* and *pre-transit* user-based traffic engineering algorithms proposed in Chapter 6 of this thesis.

The CPLEX software package is a commercial mathematical optimizer developed by ILOG. It is designed to solve linear programming, quadratic programming, quadratically constrained programming and mixed integer programming problems. The solutions are computed using proprietary algorithms in the CPLEX library, which are highly optimized for computation speed. Also, the CPLEX algorithms have the ability to handle very large and complex optimization models. The commercial package contains C++ and Java libraries that allow users to call the optimization functions in their own programming code.

In the *in-transit* and *pre-transit* user-based traffic engineering solutions (Chapter 6), the algorithms require the extensive use of LIP computations. To simulate and compare the performance of the proposed algorithms, we designed a C++ software module which models the system state of the mobile hotspot in a way that allows

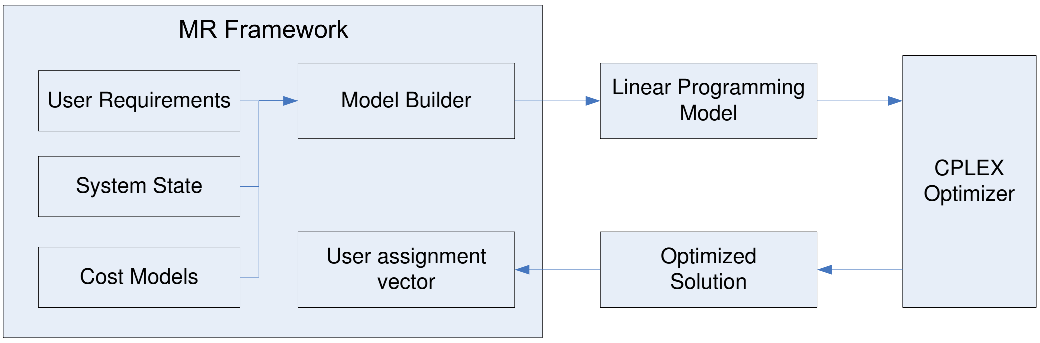

Figure B.1: The MR-framework software model

the traffic engineering algorithms to be computed by the CPLEX optimizer. This is called the *MR-framework* module, and the architecture of the MR-framework module is shown in Figure B.1.

The *MR-framework* module is designed to simulate the mobile hotspot model, which contains information that represents the state of the system. Information such as the user requirements, the current network states (e.g. the link bandwidths), and the charging models are stored in the framework, and we create the LIP model using these information. The MR-framework module is dynamically linked to the CPLEX version 9.1 C++ library, where the LIP model is built using the classes and functions provided by the CPLEX C++ library. Once the model is built, the model is passed to the CPLEX solver function for the computation of the model. Once the computation is finished, the CPLEX library returns a solution object to the caller.

To conduct the simulations for the proposed user-based traffic engineering solutions, we developed an external C++ module which is called the *MR-simulator* module. The linkage between the MR-simulation module, the MR-framework module and the CPLEX library is illustrated in Figure B.2.

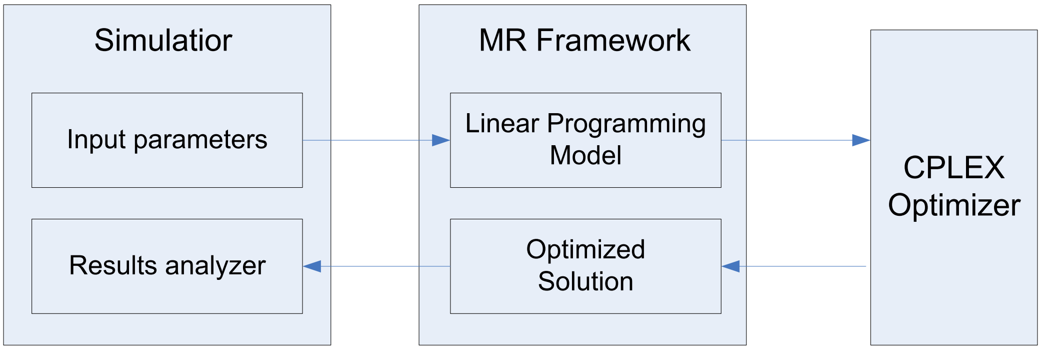

Figure B.2: The CPLEX simulation model

The MR-simulator class is responsible for conducting the various experiments of the *in-transit* and *pre-transit* algorithms. Its main role is to control the input parameters of the various simulations scenarios, and conduct the experiments accordingly. The MR-simulator builds objects of the MR-framework class, and calls the appropriate CPLEX solver solutions in the CPLEX optimizer. The *in-transit* user-based scheme was simulated by making iterative calls to the CPLEX solver, and inputting the system state information for each bandwidth variation point. In contrast, the *pre-transit* user-based scheme was simulated by creating the system state model for the entire trip and feeding it into the CPLEX solver where the computation is only performed once. Once the computations are finished, the solutions are converted to user switching tables for performance analysis. By feeding the computed user switching tables into the *results analyzer*, the results analyzer will compute the various performance metrics for comparison. The results are stored in formatted text-based data files, which can be used to plot graphs of various performance metrics.

B.2 Summary

This appendix provided an overview on the implementation details of the CPLEX simulation model that were used in the simulations of the proposed user-based traffic engineering solutions. We provided a brief introduction on the CPLEX optimizer, and discussed the implementation details of how the simulations were performed. This appendix will be useful for readers who are interested in performing their own user-based traffic engineering simulations for the mobile hotspot model.

Appendix C

GPRS measurements

C.1 Introduction

One critical issue in the mobile network architecture is the possibility of temporary outages due to low quality of wireless links between MR and the base stations. Therefore, the ability to predict link conditions before each trip, which becomes possible when the route of the vehicle is known in advance (e.g. PTVs), can significantly improve the performance of on-board moving networks. In this appendix, we show some results on the impact of different factors on the signal strength of a GPRS link in a metropolitan area. Our study suggests that location is a promising factor in the context of link prediction for on-board mobile networks. We observe that signal strength is strongly correlated with locations across different times of a day. Additionally, we find that, while the signal strength levels measured at low and medium speeds are similar, there are larger variations and more frequent handoffs at a lower speed. These insights may be useful for designing for link prediction algorithms in the future.

One critical issue in such a mobile networking architecture is the possibility of temporary outages due to low quality of wireless links between on-board routers and base stations. An outage can occur in a cellular network when the received signal strength is below a certain threshold. For example, the threshold for a GSM 900 mobile phone is -104 dBm [75]. An outage disrupts existing connections and ongoing services, which is exacerbated even in an mobile multi-homed network because a single link outage may impact a large number of existing connections.

Unlike most wireless end devices (such as mobile phones), whose mobility patterns are in general unpredictable, the routes of PTVs are known in advance and repetitive. One can take advantage of this fact to predict link conditions to a certain extent. For example, the MR may record information such as signal strength and available bandwidth on the wireless link, at different locations and times on its route. Analysis of recorded information over time may reveal that the signal loss occurs with a high probability at certain locations, times of day, or weather conditions. While a link outage may not be preventable, predicting it in advance can nonetheless improve the performance of on-board connections significantly [76]. For example, by sending a signal about an imminent outage to existing connection endpoints, the MR can prevent the chaotic and uncoordinated attempts by the individual connections to resume their data flows after the outage occurs; which may be subject to contention, excessively long timeouts, and other undesirable features.

Motivated by the above observations, we set out to explore the significance of the various factors that may be affecting the quality of the wireless signal. Our ultimate goal is to gain the insight necessary to properly design the recording and mining of wireless signal strength and bandwidth availability in the MR. Note that, while the received signal strength is obviously affected by various *physical* factors such as distance, noise, multi-path, etc, it is non-trivial to measure and hard to utilize these factors in the context of outage prediction in practice. In this work, we focus on understanding which of the potential *environmental* factors have a dominating impact on the received signal strength and can be utilized in the outage prediction process. The environmental factors include location, weather, time of day, people, vehicle velocity, and transportation mode, etc. Compared to the above mentioned *physical* factors, measurements of these *environmental* factors are easier to be obtained and utilized for outage prediction.

To answer this question, we have studied cellular network signal strength quality under a variety of conditions in a metropolitan area. Specifically, we have conducted wide-area measurements by recording the GPRS signal strength in different locations and under a variety of conditions in Sydney, Australia, focusing in particular on several public transport routes. We find that, among the factors studied, *location* is clearly the most dominating factor affecting the signal quality, with the overall impact of all other factors being much less significant. Therefore, even a simple outage prediction approach, taking into account only the location of the vehicle is likely to bring about a performance improvement in practice. We note that, undoubtedly, similar measurements have also been performed by the cellular service providers and some consulting companies. However, to our best knowledge, their results are not publicly available.

The rest of this appendix is organized as follows. In Section C.2 we present some related work. Section C.3 describes our data collection process, and our results are discussed in Section C.4. Finally, conclusions and future work are presented in Section C.5.

C.2 Related work

Our work builds on prior work in on-board communication, outage prediction, signal strength measurements and GPRS network measurements. The idea that the advance knowledge and repetitiveness of public transport routes can be used to predict wireless signal outages was suggested by Baig et al. [76], who proposed to employ the Freeze-TCP [77] extension on the MR and analyzed the TCP performance benefits gained by outage prediction. They assumed outages to be predicted independently of each other with a given probability, and showed that the improvement in TCP throughput is super-linear in the prediction probability, especially when the TCP connections used large window sizes. However, they did not consider the mechanisms required to implement the prediction in practice.

Empirical measurements of the wireless signal strength and of its dependence on various factors have been done by cellular service providers and consulting companies. For example, Omnitele [78] conducted experiments to measure call success rate, signal strength and throughput along roads, train routes and in urban and rural areas. However, instead of characterizing network behaviors, their objective was to utilize the collected measurements to optimize operators' network performance for better customer service and hardware utilization. However, their data is not publicly available. Wagen [79] conducted a series of experiments in small (62 meters by 65 meters) urban areas to measure signal loss in both line-of-sight and non-line-of-sight conditions. He developed an empirical model to characterize the relationship between signal strength and distance. Lastly, Chen and Siew [80] have conducted indoor experiments to measure the performance of a wireless LAN. Their focus was to investigate how the packet and bit level error characteristics are affected by different environmental factors such as humidity, microwave interference, wall obstacle and distance.

Several previous studies [81–83] have investigated TCP performance on GPRS links by conducting stationary experiments over production networks. They found that the long queuing delay of GPRS links has a major negative effect on TCP performance. They showed that GPRS links are currently plagued with several problems such as high and variable round trip time, bursty packet loss and frequent link outages.

Unlike prior work, the focus of our work is to investigate the impact of different environmental factors on the received signal strength in an operational GPRS network and search for the dominant factors to be used for outage prediction in a known route, as is the case, e.g., for a public transport or a regularly commuting private vehicle.

C.3 Data collection

For data collection, an undergraduate honors thesis student named Irene Chan traveled around the Sydney metropolitan area and conducted measurements of GPRS signal strength on public transportation systems. We recorded the signal strength of Vodafone's [1] GPRS network at different locations and under different conditions such as moving speeds, times of a day, levels of humidity, etc. Our measurements record the quality of signal of wireless links between the receiver and base stations, the identity of base stations that our mobile user connected to, and locations and timestamps when measurements were taken. We used a Garmin eTrex Global Positioning System (GPS) receiver to record locations and times when measurements were taken. Due to the line-of-sight limitation of GPS, for certain locations (e.g.

[1] Vodafone is one of the largest national GPRS operators in Australia.

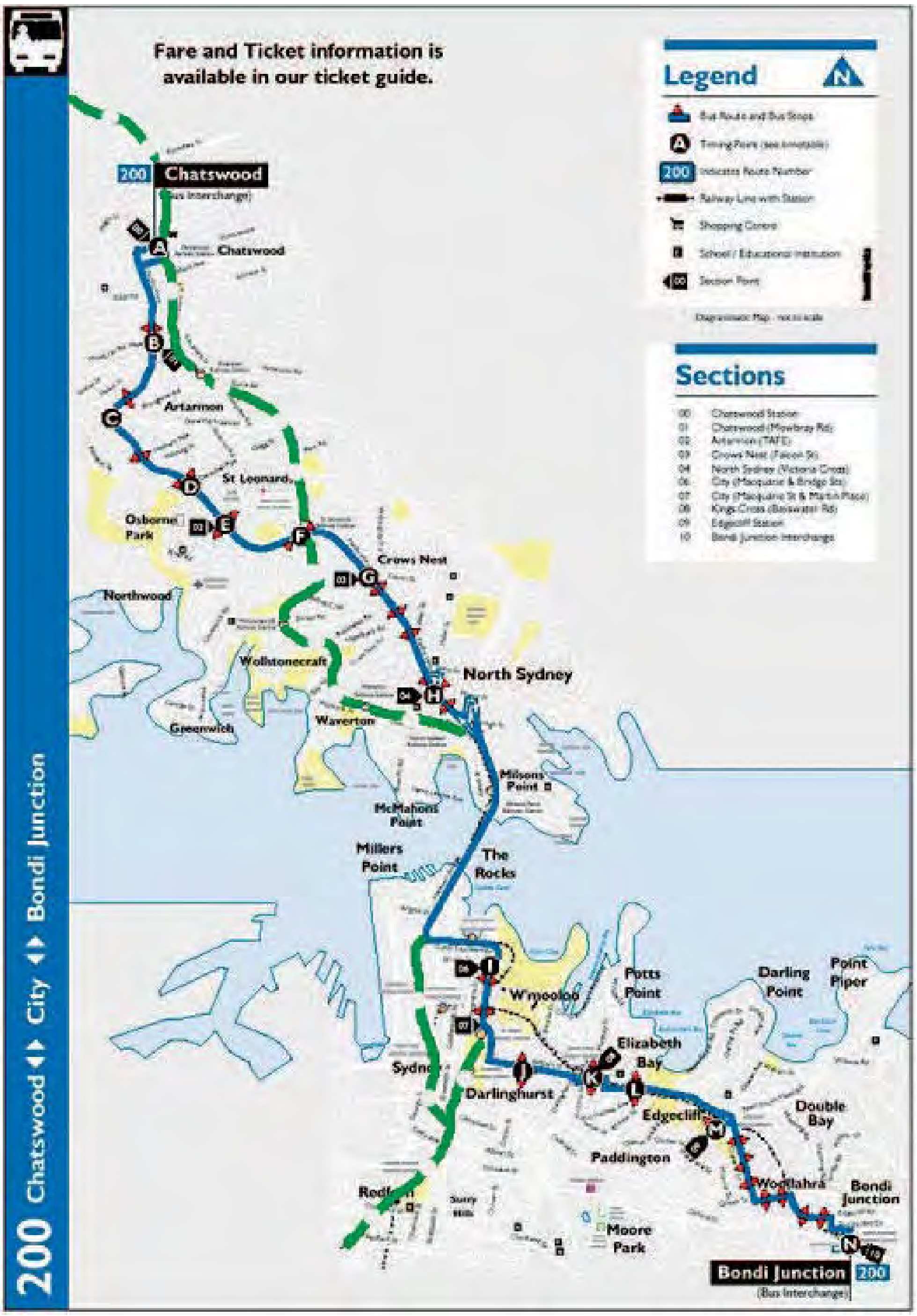

Figure C.1: Bus and train routes where most measurements were taken. The green dotted line (on the left) denotes the Chatswood-Bondi bus route. The blue line (on the right) indicates the St. Leonards-Central train route.

inside a tunnel) we manually recorded the time and estimated the location based on surrounding landmarks.

The data was collected during a six-month period between October 2003 and April 2004. Our traces were collected on different transportation means including train, bus and car. Some experiments were performed in a car driven along a bus route to study the effect of different speeds on signal strength reception. We also performed the experiments at some chosen locations in the Sydney urban area for stationary measurements. The bus and train routes where measurements were taken, run across an area from north to south Sydney. For the train experiments, the route from St. Leonards to Central station is chosen. For the bus experiments, the route 200 traveling between Bondi and Chatswood is chosen. Figure C.3 shows the bus and train routes where measurements were taken. We measured the quality of signal at different locations and under different conditions such as different moving speed and different times of a day, etc. Each experiment was repeated a number of times to ensure that our results are consistent.

C.4 Results

We investigated the effects of a range of different environmental factors on the measured signal strength. In this appendix, we present our preliminary results for the following factors: location, moving speed, obstruction, humidity and people. Among those factors we studied, we find that *location* has the most significant effect on the measured signal strength.

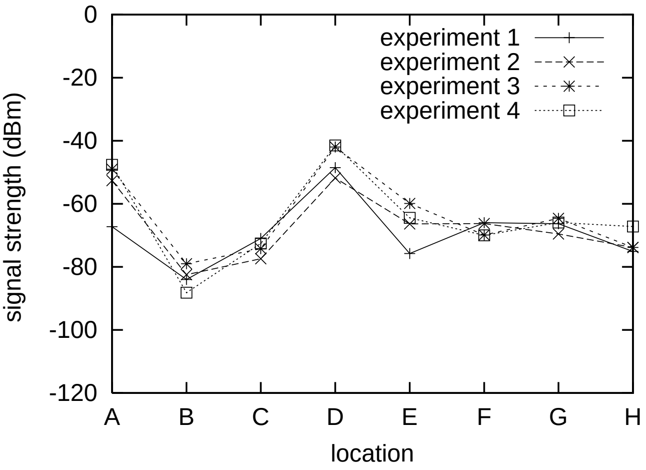

Figure C.2: Signal strength measured at different locations across different times.

C.4.1 Location

We find that signal strength is strongly correlated with *locations* across different times of day. As shown in Figure C.2, each line indicates a different experiment that was performed at different times on the same train route. We find that the quality of signal varies significantly among different locations, but remains similar across different times of day for the same location. The above observation suggests the possibility of using correlations between location and signal strength as a predictor for outage prediction.

Furthermore, we find that there is some correlation between the location and the base station that our mobile user connected to. Figure C.3 shows a sequence of switching between different base stations when the mobile user moved along a train route. Although the switching sequences are not identical between experiments, they indicate a certain degree of predictability. Such an observation seems to suggest the

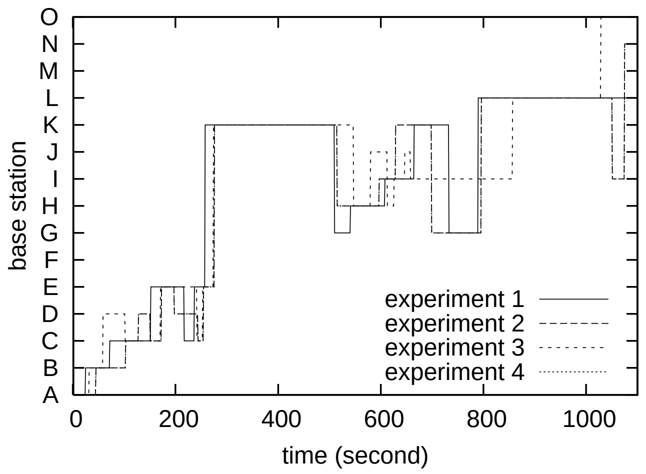

Figure C.3: Switching sequence to different base stations at different times.

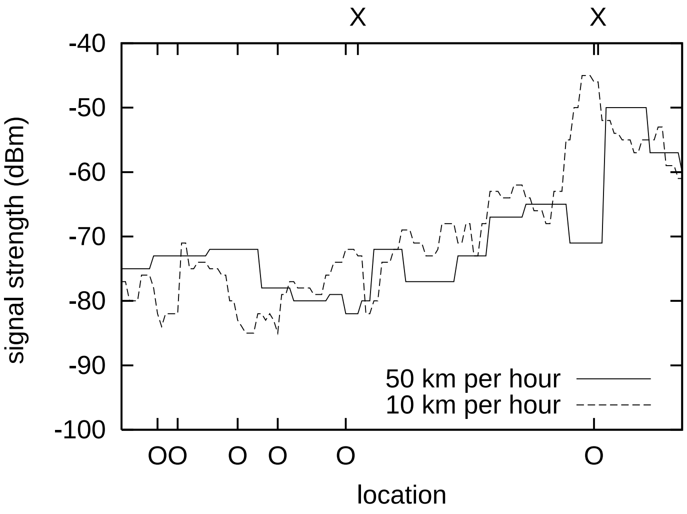

Figure C.4: Signal strength measured at different speeds on the same route. “X” marks indicate the locations where handoff occurs when traveling at 50km/hr. “O” marks indicate the locations where handoff occurs when traveling at 10km/hr.

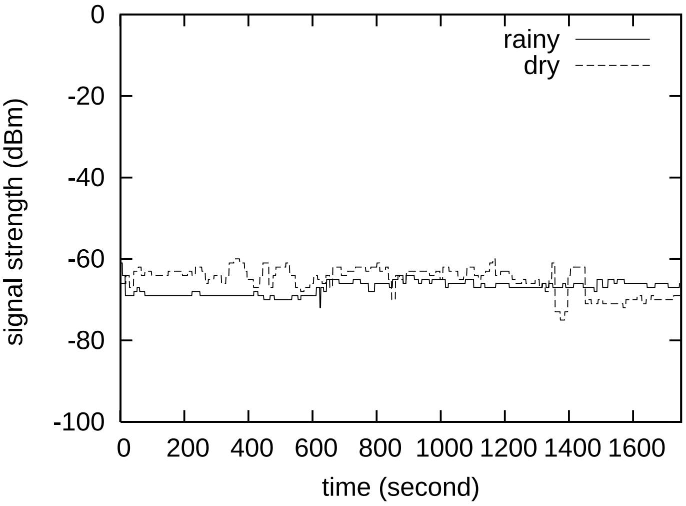

Figure C.5: The effect of humidity on the measured signal strength.

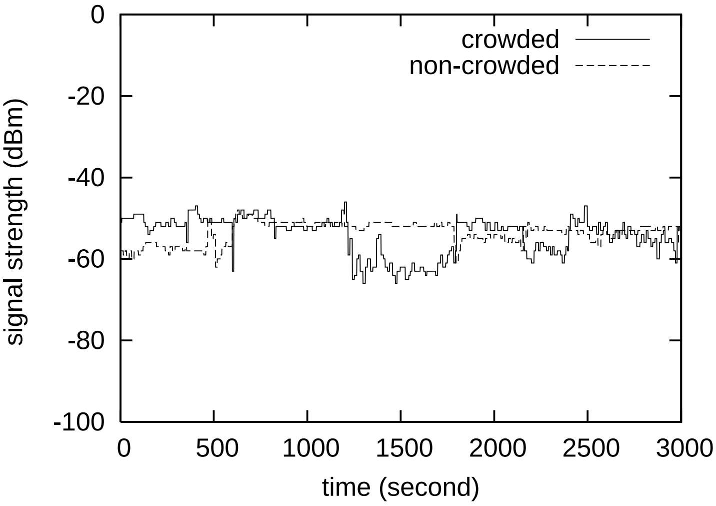

Figure C.6: The effect of people on the measured signal strength.

feasibility of deploying some resource reservation schemes to improve the performance of on-board networks. For example, one could employ some pre-association techniques to reduce the latency of handoff for base stations that are in different IP subnets.

C.4.2 Other factors

In addition to *location*, we also investigate a number of other environmental factors (such as moving speed, humidity, people, transportation mode, etc.) but find that they do not have significant effects on the received signal strength. Consequently, we only present some preliminary results for the following factors: speed, humidity, people and tunnel.

Speed

To investigate effects of speed on signal strength reception, we measured signal strength when driving at different speeds, namely 10km/hr and 50km/hr, on the same route. We find that there is no significant difference in the received signal strength level, as shown in Figure C.4. However, there are more variations when traveling at a lower speed. Our hypothesis is that, when moving at a lower speed, the receiver would bypass each location more slowly so that the *location* factor has a less dominant effect on the signal strength compared to when driving at a higher speed. In other words, when a user travels faster, other environmental factors might have less time to change and less impact on the received signal. As a result, the user experiences a smaller variation in the received signal strength because only the location affects the signal and other factors remains more or less constant during the travel time.

In addition, we observed that there were more handoffs when measurements were taken at a lower speed. The handoffs are shown as “X” and “O” for different speeds in Figure C.4 (which might be explained by the larger variations in signal strength when driving at a lower speed). The above observations might be useful for improving the realism of wireless simulations. For example, a business merger might result in some of the base stations being in different IP subnets even though they are owned by the same operator [84].In that case, one might want to take the frequency of handoffs into consideration when simulating the traveling between different base stations for some delay-sensitive applications, such as VoIP or streaming traffic.

Humidity

Precipitation in the atmosphere can cause signals to be absorbed and weakened, also known as rain attenuation. We performed six experiments in both dry and rainy conditions. The rainfall rate for experiments performed in rainy days is 2.0 millimeters per hour. As shown in Figure C.5, we find that the average signal strengths in both cases are similar when measured at the same locations. Rain attenuation typically has a stronger effect on wireless networks that operate in higher frequency bands (such as 10GHz). However, GPRS operates in a lower frequency band (900MHz and 1.8GHz in Australia) and is not affected much by rain attenuation. Note that some base stations utilize microwave links to connect to the mobile service switching center (MSC). Microwave links are more susceptible to rain attenuation since they operate in a higher frequency, although our experiments do not capture the rain effect on such links.

People

To understand the effect of people on the received signal strength, we performed experiments in a crowded spot at our university. For comparison, we repeated the same experiments during Easter break when most students have left for vacation. As shown in Figure C.6, we find that the average signal strength in both cases are similar. However, we find that the standard deviation of signal strength measured in a crowded environment is two orders of magnitude higher than the un-crowded case. This is not surprising though since moving people might obstruct (or absorb) signal traveling in between and cause larger variations of the received signal.

Tunnel

We also measured the received signal strength when traveling on a train inside tunnels. We find that the existence of tunnel does not necessarily lead to continuous outages. As shown in Figure C.7 and Figure C.8, outages only occurred in some parts of the tunnel. The numbers on top of the graphs indicate the time when measurements were taken. The marks above the numbers indicate where a cell switching occurred. This observation might be due to the deployment of micro-cells. In practice, cellular network providers typically deploy micro-cells around platforms to provide temporary coverage in the tunnels. As shown in Figure C.7 and Figure C.8, the received signal strength significantly increased when the train was near the platform inside a tunnel.

Due to administrative issues, train tunnels typically do not have coverage from any operator because they are public assets. In addition, in practice it is difficult to deploy repeaters or amplifiers in tunnels to improve the signal coverage.

C.5 Summary

In an on-board communication network, it is important to understand when temporary outages occur due to signal degradation when users travel between different environments. In particular, the ability to predict link outages before they occur; which may be possible when the route is known is advance and was traveled before (e.g. in the case of a public transport or regularly commuting private vehicle); can significantly improve the service quality of on-board network connections.

In this appendix, we show some measurement results on the impact of different factors on the received signal strength in a metropolitan area. Our preliminary measurement results suggest that *location* is a dominant factor in the context of outage prediction for on-board mobile networks. Additionally, we find that, while signal strength measured at low and medium speeds are similar, there are larger variations and more frequent handoffs at a lower speed. These insights provide an important first step on the way to designing a practical outage prediction algorithm in a MR.

Because of the hardware limitation, in our preliminary study we recorded only the received signal strength. Information on Signal-to-Noise Ratio (SNR) might be useful for evaluating the actual perceived link quality to applications, and we plan to collect SNR data as a future work. While we are not aware of any existing GPRS card that can measure SNR, some cards provide a reading of *reception quality* which uses SNR as one of its inputs.

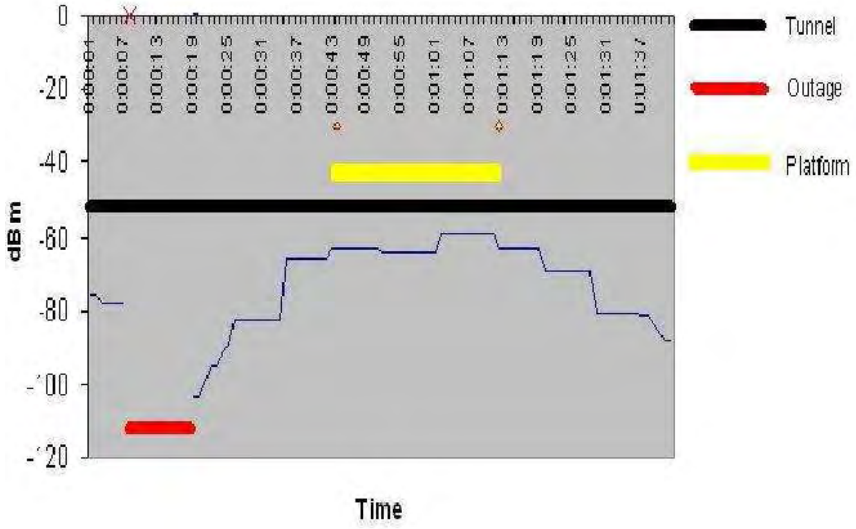

Figure C.7: Effect of tunnels on the measured signal strength: outage occurs only in some parts of the tunnel.

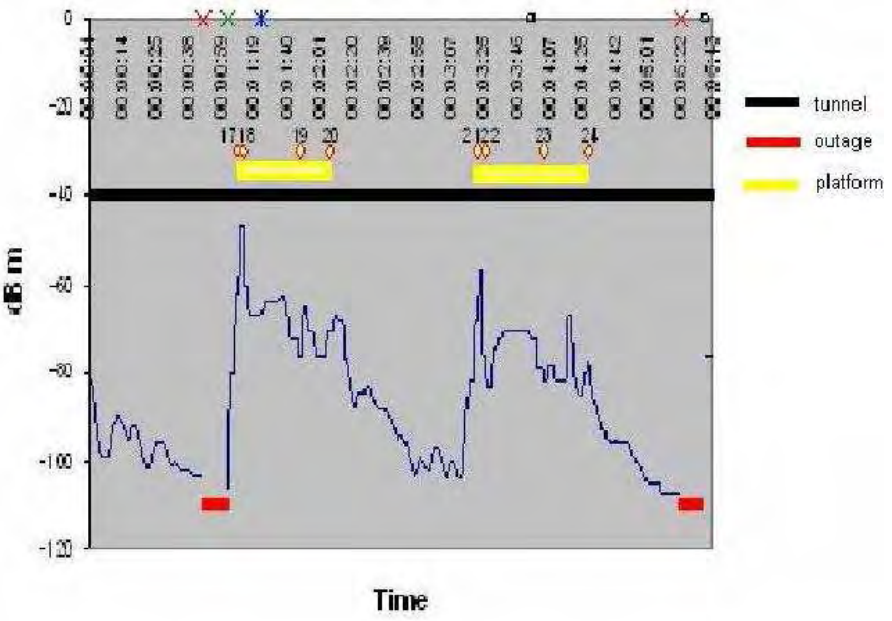

Figure C.8: The effect of people on the measured signal strength.

Bibliography

[1] icomera, "http://www.icomera.com/."

[2] PointShot Wireless, "http://www.pointshotwireless.com/."

[3] Inmotion Technology, "http://www.inmotiontechnology.com/."

[4] Connexion, "http://www.connexionbyboeing.com/."

[5] Cisco Mobile Router, "http://www.cisco.com/en/us/products/hw/routers /ps272/prod_presentation_list.html."

[6] IETF NEMO Working Group, "http://www.ietf.org/html.charters/nemo-charter.html."

[7] European Commission IST OverDRiVE, "http://www.ist-overdrive.org."

[8] Nautilus 6 working group, "http://www.nautilus6.org/."

[9] Korean NEMO, "http://mmlab.snu.ac.kr//research/project/2003/nemo."

[10] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert, "Network mobility (NEMO) basic support protocol," Jan. 2005, IETF RFC3963.

[11] P. Smith, "BGP multihoming techniques," *http://www.nanog.org/mtg-0110/smith.html*, 2001.

[12] T. Bates and Y. Rekhter, "Scalable support for multi-homed multi-provider connectivity," Jan. 2004, IETF RFC2260.

[13] V. Cardellini, M. Colajanni, and P. S. Yu, "Dynamic load balancing on webserver systems." *IEEE Internet Computing*, vol. 3, no. 3, pp. 28–39, 1999.

[14] K. Egevang and P. Francis, "The IP Network Address Translator (NAT)," May 1994, IETF RFC1631.

[15] S. Sinha, S. Kandula, and D. Katabi, "Harnessing TCPs Burstiness using Flowlet Switching," in *3rd ACM SIGCOMM Workshop on Hot Topics in Networks (HotNets)*, San Diego, CA, November 2004.

[16] S. Kandula, D. Katabi, B. Davie, and A. Charny, "Walking the tightrope: responsive yet stable traffic engineering," *SIGCOMM Comput. Commun. Rev.*, vol. 35, no. 4, pp. 253–264, 2005.

[17] S. Rost and H. Balakrishnan, "Rate-Aware Splitting of Aggregate Traffic," in *MIT Technical Report*, 2003.

[18] N. Yamai, K. Okayama, H. Shimamoto, and T. Okamoto, "A Dynamic Traffic Sharing with Minimal Administration on Multihomed Networks," in *Proceedings of IEEE International Conference on Communications (ICC)*, vol. 5, 2001, pp. 1506–1510.

[19] D. K. Goldenberg, L. Qiuy, H. Xie, Y. R. Yang, and Y. Zhang, "Optimizing cost and performance for multihoming," in *SIGCOMM '04: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications.* New York, NY, USA: ACM Press, 2004, pp. 79–92.

[20] K.-H. Kim and K. G. Shin, "Improving tcp performance over wireless networks with collaborative multi-homed mobile hosts," in *MobiSys '05: Proceedings of the 3rd international conference on Mobile systems, applications, and services.* New York, NY, USA: ACM Press, 2005, pp. 107–120.

[21] A. Qureshi and J. Guttag, "Horde: separating network striping policy from mechanism," in *MobiSys '05: Proceedings of the 3rd international conference on Mobile systems, applications, and services.* New York, NY, USA: ACM Press, 2005, pp. 121–134.

[22] D. S. Phatak and T. Goff, "A Novel Mechanism for Data Streaming Across Multiple IP Links for Improving Throughput and Reliability in Mobile Environments." in *Proceedings of IEEE Infocom*, vol. 2, June 2002, pp. 773–781.

[23] H.-Y. Hsieh, K.-H. Kim, Y. Zhu, and R. Sivakumar, "A Receiver-Centric Transport Protocol for Mobile Hosts with Heterogeneous Wireless Interfaces," in *Proceedings of ACM MOBICOM*, 2003.

[24] Nokia WiFi Mobile Phones, "http://www.nokia.com/."

[25] J. Ala-Laurila, J. Mikkonen, and J. Rinnemaa, "Wireless lan access network architecture for mobile operator," *IEEE Communications Magazine*, pp. 82–89, November 2001.

[26] H. Haverinen, J. Mikkonen, and T. Takamaki, "Cellular access control and charging for mobile operator wireless local area networks," *IEEE Wireless Communications Magazine*, pp. 52–60, December 2002.

[27] A. Salkintzis, C. Fors, and R. Pazhyannur, “Wlan-gprs integration for next-generation mobile data networks,” *IEEE Wireless Communications Magazine*, pp. 112–124, October 2002.

[28] P. Rodriguez, R. Chakravorty, J. Chesterfield, I. Pratt, and S. Banerjee, “Mar: a commuter router infrastructure for the mobile internet,” in *MobiSys ’04: Proceedings of the 2nd international conference on Mobile systems, applications, and services.* New York, NY, USA: ACM Press, 2004, pp. 217–230.

[29] E. Paik and Y. Choi, “Seamless Mobility Support for Mobile Networks on Vehicles across Heterogeneous Wireless Access Networks,” in *Proceedings of IEEE VTC-Spring*, vol. 4, April 2003, pp. 2437–2441.

[30] J. Chesterfield, R. Chakravorty, I. Pratt, S. Banerjee, and P. Rodriguez, “Exploiting Diversity to Enhance Multimedia Streaming Over Cellular Links,” in *Proceedings of IEEE Infocom*, vol. 3, 2005, pp. 2020 – 2031.

[31] C.-W. Ng and T. Ernst, “Multiple Access Interfaces for Mobile Nodes and Networks,” in *Proceedings of International Conference On Networks (ICON)*, vol. 2, 2004, pp. 774 – 779.

[32] E. Paik, H. Cho, T. Ernst, and Y. Choi, “Load Sharing and Session Preservation with Multiple Routers for Large Scale Network Mobility,” in *Proceedings of International Conference on Advanced Information Networking and Application (AINA)*, vol. 1, 2004, pp. 393–398.

[33] M. Tsukada, T. Ernst, R. Wakikawa, and K. Mitsuya, “Dynamic Management of Multiple Mobile Routers,” in *Proceedings of IEEE International Malaysia*

Conference on Communications and IEEE International Conference in Networks (MICC-ICON), vol. 2, 2005, pp. 1108 – 1113.

[34] L. Suciu, J.-M. Bonnin, K. Guillouard, and T. Ernst, "Multiple Network Interfaces Management for Mobile Routers," in *International Conference on ITS Telecommunications (ITST)*, 2005.

[35] H. Lach, C. Janneteau, and A. Petrescu, "Network Mobility in Beyond-3G Systems," *IEEE Communications Magazine*, pp. 52–57, July 2003.

[36] Y.-B. Lin and J. Y.-B. Lin, *Wireless and Mobile Network Architectures.* New York, NY, USA: John Wiley & Sons, Inc., 2000.

[37] IETF MONAMI6 Working Group, "http://www.ietf.org/html.charters/monami6-charter.html."

[38] L. Qiu, P. Bahl, and A. Adya, "The Effects of First-Hop Wireless Bandwidth Allocation on End-to-End Network Performance," in *Proceedings of ACM NOSSDAV*, 2002.

[39] I. Chan, A. Chung, K. Lan, L. Libman, and M. Hassan, "Understanding the effect of environmental factors on link quality for on-board communications," in *Proceedings of IEEE VTC-Fall*, vol. 3, 2005, pp. 1877 – 1881.

[40] F. Guo, J. Chen, W. Li, and T. Chiueh, "Experiences in Building a Mulithoming Load Balancing System," in *Proceedings of IEEE Infocom*, vol. 2, 2004, pp. 1241 – 1251.

[41] M. Katevenis, S. Sidiropoulos, and C. Courcoubetis, "Weighted round-robin cell multiplexing in a general-purpose atm switch chip." *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 8, pp. 1265–1279, 1991.

[42] A. Francini, F. M. Chiussi, R. T. Clancy, K. D. Drucker, and N. E. Idirene, "Enhanced weighted round robin schedulers for accurate bandwidth distribution in packet networks," *Comput. Networks*, vol. 37, no. 5, pp. 561–578, 2001.

[43] H. M. Chaskar and U. Madhow, "Fair scheduling with tunable latency: a round-robin approach," *IEEE/ACM Trans. Netw.*, vol. 11, no. 4, pp. 592–601, 2003.

[44] D.-C. Li, C. Wu, and F. M. Chang, "Determination of the parameters in the dynamic weighted round-robin method for network load balancing," *Comput. Oper. Res.*, vol. 32, no. 8, pp. 2129–2145, 2005.

[45] K. Thompson, G. J. Miller, and R. Wilder, "Wide-area Internet traffic patterns and characteristics," *IEEE Network*, vol. 11, no. 6, pp. 10–23, Nov./Dec. 1997.

[46] I. Matta and L. Guo, "Differentiated predictive fair service for tcp flows." in *ICNP: Proceedings of the Eighth International Conference on Network Protocols.* Osaka, Japan: IEEE Computer Society, 2000, pp. 49–58.

[47] L. Guo and I. Matta, "The war between mice and elephants," in *ICNP: Proceedings of the Ninth International Conference on Network Protocols.* Washington, DC, USA: IEEE Computer Society, 2001, p. 180.

[48] X. G. Meng, S. H. Y. Wong, Y. Yuan, and S. Lu, "Characterizing flows in large wireless data networks," in *Proceedings of ACM MOBICOM*, 2004, pp. 174–186.

[49] T. Henderson, E. Sahouria, S. McCanne, and R. Katz, "On Improving the Fairness of TCP Congestion Avoidance," in *Proceedings of IEEE Globecom*, 1998.

[50] A. B. Downey, "Using Pathchar to estimate internet link characteristics," in *Proceedings of ACM SIGCOMM*, 1999, pp. 222–223.

[51] K. Lai and M. Baker, "Measuring link bandwidths using a deterministic model of packet delay," in *Proceedings of ACM SIGCOMM*, Aug. 2000, pp. 283–294.

[52] K. Lai and M. Baker, "Nettimer: A tool for measuring bottleneck link bandwidth," in *Proceedings of the USENIX Symposium on Internet Technologies and Systems*, Mar. 2001.

[53] UCB/LBNL/VINT Network Simulator-ns (Version 2), "http://www.isi.edu/nsnam/ns/."

[54] C. Courcoubetis, V. A. Siris, and G. D. Stamoulis, "Network control and usage-based charging: is charging for volume adequate?" in *ICE '98: Proceedings of the first international conference on Information and computation economies*. New York, NY, USA: ACM Press, 1998, pp. 77–82.

[55] N. Yamai, K. Okayama, H. Shimamoto, and T. Okamoto, "TCP performance over GPRS," in *Proceedings of IEEE Wireless Communication and Networking Conference (WCNC)*, vol. 3, 1999, pp. 1248–1252.

[56] F. Eyermann, P. Racz, B. Stiller, C. Schaefer, and T. Walter, "Generic accounting configuration management for heterogeneous mobile networks," in *WMASH*

'05: Proceedings of the 3rd ACM international workshop on Wireless mobile applications and services on WLAN hotspots. New York, NY, USA: ACM Press, 2005, pp. 46–55.

[57] Australian Broadband Choice, "http://bc.whirlpool.net.au/."

[58] H.-Y. Wei and Y.-D. J. Lin, "A survey and measurement-based comparison of bandwidth management techniques." *IEEE Communications Surveys and Tutorials*, vol. 5, no. 2, 2003.

[59] H.-Y. Wei, S.-C. Tsao, and Y.-D. J. Lin, "Assessing and improving tcp rate shaping over edge gateways." *IEEE Trans. Computers*, vol. 53, no. 3, pp. 259–275, 2004.

[60] M. Liebsch, H. Chaskar, D. Funato, and E. Shim, "Candidate Access Router Discovery," *draft-ietf-seamoby-card-protocol-07.txt*, IETF working draft.

[61] A. Talukdar, B. Badrinath, and A. Acharya, "MRSVP: A Resource Reservation Protocol for an Integrated Services Network with Mobile Hosts," *Wireless Networks*, vol. 7(1), pp. 5–19, 2001.

[62] C. Tseng, G. Lee, R. Liu, and T. Wang, "HMRSVP:A Hierarchical Mobile RSVP Protocol," *Wireless Networks*, vol. 9(2), pp. 95–102, March 2003.

[63] W. Chen and L. Huang, "RSVP mobility support: A signaling protocol for integrated services Internet with mobile hosts," in *Proceedings of IEEE INFOCOM*, vol. 3, 2000, pp. 1280–1292.

[64] ILOG CPLEX, "http://www.cplex.com/."

[65] L. Kalampoukas, A. Varma, and K. K. Ramakrishnan, “Two-way tcp traffic over rate controlled channels: effects and analysis,” *IEEE/ACM Trans. Netw.*, vol. 6, no. 6, pp. 729–743, 1998.

[66] H. Balakrishnan, “Challenges to reliable data transport over heterogeneous wireless networks,” Ph.D. dissertation, 1998, chair-Randy H. Katz.

[67] H. Balakrishnan, R. H. Katz, and V. N. Padmanbhan, “The effects of asymmetry on tcp performance,” *Mob. Netw. Appl.*, vol. 4, no. 3, pp. 219–241, 1999.

[68] N. Joshi, S. R. Kadaba, S. Patel, and G. S. Sundaram, “Downlink scheduling in cdma data networks,” in *MobiCom ’00: Proceedings of the 6th annual international conference on Mobile computing and networking.* New York, NY, USA: ACM Press, 2000, pp. 179–190.

[69] B. Mah, “An empirical model of HTTP network traffic,” Kobe, Japan, Apr. 1997, pp. 592–600.

[70] A. Feldmann and J. Rexford, “Efficient policies for carrying web traffic over flow-switched networks,” *IEEE/ACM Trans. Netw.*, vol. 6, no. 6, pp. 673–685, 1998.

[71] S. Floyd and V. Paxson, “Difficulties in simulating the Internet,” vol. 9, no. 4, pp. 392–403, Feb. 2001.

[72] UNSW OCEAN project, “http://ocean.cse.unsw.edu.au.”

[73] K. Lan, H. Petander, E. Perrera, L. Libman, C. Dwertman, and M. Hassan, “MOBNET: The Design and Implementation of a Network Mobility Testbed for

NEMO protocol," in *Proceedings of IEEE Workshop on Local and Metropolitan Area Networks*, 2005, pp. 1–6.

[74] The ns Manual, "http://www.isi.edu/nsnam/ns/doc/."

[75] 3rd Generation Partnership Project, "Technical Specification Group GSM/EDGE: Radio transmission and reception," *3GPP TS 45.005*, 2004.

[76] A. Baig, M. Hassan, and L. Libman, "Prediction-based recovery from link outages in on-board mobile communication networks," in *Proceedings of IEEE Globecom*, Dallas, TX, Nov. 2004.

[77] T. Goff, J. Moronski, D. Phatak, and V. Gupta, "Freeze-TCP: A true end-to-end enhancement mechanism for mobile environments," in *Proceedings of IEEE Infocom*, Tel-Aviv, Israel, Mar. 2000.

[78] Omnitele, "http://www.omnitele.fi/."

[79] J.-F. Wagen, "Signal strength measurements at 881MHz for urban microcells in downtown Tampa," in *Proceedings of IEEE Globecom*, Phoenix, AZ, Dec. 1991.

[80] V. Chen and K. Siew, "Characterising errors in wireless LAN," in *Undergraduate thesis, University of New South Wales*, 2003.

[81] R. Chakravorty, S. Banerjee, P. Rodriguez, J. Chesterfield, and I. Pratt, "Performance optimizations for wireless wide-area networks: Comparative study and experimental evaluation," in *Proceedings of ACM Conference on Mobile Computing and Networking (MobiCom)*, Philadelphia, PA, Sept. 2004.

[82] R. Chakravorty, J. Cartwright, and I. Pratt, “GPRSWeb: Optimizing the Web for GPRS links,” in *Proceedings of ACM Conference on Mobile Systems, Applications and Services (MobiSys)*, San Francisco, CA, May 2003.

[83] R. Chakravorty, J. Cartwright, and I. Pratt, “Practical experience with TCP over GPRS,” in *Proceedings of IEEE Globecom*, Taipei, Taiwan, Nov. 2002.

[84] Z. M. Mao, J. Rexford, J. Wang, and R. Katz, “Towards an accurate AS-level traceroute tool,” in *Proceedings of ACM SIGCOMM*, Karlsruhe, Germany, Aug. 2003.