UNIVERSITY OF CALIFORNIA, SAN DIEGO

Operator and Radio Resource Sharing in Multi-Carrier Environments

A thesis submitted in partial satisfaction of the requirements
for the degree Master of Science

in

Computer Science

by

Pongsakorn Teeraparpwong

Committee in charge:

Professor Amin Vahdat, Chair
Professor Alex Snoeren
Professor Geoffrey M. Voelker

2009

The thesis of Pongsakorn Teeraparpwong is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____
Chair

University of California, San Diego

2009

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to each and every person who has taken part in the completion of this thesis. First, I would like to thank my advisor Amin Vahdat for his excellent and consistent guidance and support. This project would not have been completed without his wisdom, vision as well as financial and moral support.

Also, I would like to acknowledge Martin Johnsson for his invaluable knowledge contributing to this thesis. I also really appreciate his effort to join us in every teleconference and give us visits regardless of the distance between him and UCSD.

In addition, I would like to thank Per Johansson for his insight and valuable comments and support as well as his assistance as a coordinator.

Moreover, I would like to thank Alex Snoeren and Geoffrey Voelker for their support and for serving in my thesis committee.

Furthermore, I also would like to thank my brilliant colleague Harsha Madhyastha, who has helped me overcome many challenges and given me a lot of decent advice.

Finally, in this work, we have extended and used Swing developed by Kashi Vishwanath and resource reservation framework developed by Mohammad Al-Fares. I would like to acknowledge both of them for creating these wonderful programs and giving me assistance in setting up and using these programs.

ABSTRACT OF THE THESIS

Operator and Radio Resource Sharing in Multi-Carrier Environments

by

Pongsakorn Teeraparpwong

Master of Science in Computer Science

University of California, San Diego, 2009

Professor Amin Vahdat, Chair

A key challenge that the mobile networking world is facing is seamless network composition. In spite of a wide range of available access technologies, the operator agreements prevent users to freely access these networks. With seamless network composition, users can gain better quality of service, and operators can provision less bandwidth by sharing their resources. Al-Fares et al. [4] previously proposed a resource reservation framework that implements this network composition concept. We extended their work by analyzing the benefits of operator cooperation in a realistic environment using their framework. In our scenario, we tried to leverage the difference in burstiness in small timescales to shed the peak usage of one operator onto another. We utilize Swing [26] to reproduce traffic based on the real 3G trace data. The results show that even when the operator's capacities are limited, the cooperation can help maintain quality of service for most sessions.

Nevertheless, even if all access technologies can be accessed seamlessly, more challenges appear to be which technology users should use and which combination of access technologies is appropriate to deploy. To provide more insight to these

challenges, we investigated the performance delivered by various kinds of wireless technologies. We utilized Swing's abilities to tune the topology parameters to reflect different access technologies. Our results show that WiFi can provide significantly better performance compared to the 3G. On the other hand, we expect the performance of LTE to be comparable to or even better than WiFi.

Chapter 1

Introduction

In recent years, the mobile and wireless networking world has undergone a rapid evolution. Driven by a rapid increase in the demand for superior services and a highly competitive market environment, both business and technology environments have become more heterogeneous and dynamic. Many novel wireless access technologies have been under development or deployment globally including 3G/UMTS, WiFi, WiMAX or LTE. New business entities such as virtual operators and aggregators/brokers have also been emerging.

With the current trends, a key challenge that the cellular world is facing is seamless network composition. In particular, although the coverage of mobile networks is ubiquitous, the access of users is limited to only their home operator's networks. The users cannot freely and seamlessly access any operator or any network at any time due to the relations and agreements between service providers.

In order to allow users to roam across different operators and networks seamlessly, a concept of network composition has been developed. With this concept, users can gain a better service coverage and quality of service, especially when their home operator has limited resources. Additionally, operators do not need to provision the peak bandwidth usage of every network and every access technology. The peak usage can be offloaded to other networks or other access technologies which still have more available resources at that time. Therefore, the provisioning cost can be reduced. Also, all the resources can be more effectively and fully utilized. Nevertheless, the trade-off is that the quality of service can be

impacted if the provisioning bandwidth is not enough.

Several efforts have been trying to implement the network composition concept such as the European Ambient Networks (AN) [20]. Also, Al-Fares et al. [4] previously proposed a resource reservation framework to enable seamless network composition across multiple wireless access technologies and operators. Nevertheless, the concept and the framework have yet to be evaluated in a realistic environment with real usage behavior of users.

In this work, we extended the framework proposed by Al-Fares et al. to analyze the benefits of operator cooperation based on realistic 3G data traffic. In our scenario, we tried to leverage the difference in burstiness in small timescales to shed the peak data bandwidth usage of one operator onto another even when the average load of all operators are almost equal.

We used the Swing [26] traffic generator tool to extract the traffic characteristics and generate the traffic based on traces captured from an operational 3G network. We extended Swing to create a multi-operator scenario and integrated the resource reservation framework to the system. The ModelNet [25] emulation tool was used along with Swing to emulate the wide-area network characteristics and the sharing of resources between operators.

The results show that even when the capacities of all operators are limited to less than the peak usage, the operator cooperation can help maintain quality of service upto the non-constrained case for most of the user's sessions. As a result, all operators can benefit from the cooperation by provisioning lesser bandwidth. Also, we found that the delays due to the control signals required for operator cooperation mostly result from the high wireless link latency and are expected to be able to hide within the existing client's initial setup time.

Nevertheless, even all radio access technologies (RATs) can be shared and accessed seamlessly, another challenge appears to be which technology users should use in order to gain the best performance or minimize the cost/performance ratio. From an operator's point of view, the challenges include which combination of access technologies is more appropriate to deploy in order to support users' demand at present and in the future with minimum cost.

To provide more insight to these challenges, we also investigated the performance delivered by various kinds of wireless technologies including 3G, WiFi 802.11b/g and LTE for various application classes based on the real user usage. We utilized Swing's abilities to generate realistic traffic as well as to tune the topology parameters to reflect different kinds of access technologies. Our results show that 802.11b/g can provide significantly better performance compared to 3G in every application class. Therefore, WiFi hotspots can be a good choice to augment 3G services in a small highly-populated area such as a coffee shop or an airport. On the other hand, we expect the performance of LTE – the next evolution of 3G – to be comparable to or even better than WiFi, especially when the number of users increases.

The rest of the thesis is organized as follows. Chapter 2 presents prior work related to our work. Chapter 3 gives the background knowledge about different wireless access technologies. In Chapter 4, the resource allocation framework proposed by Al-Fares et al. [4] is reviewed. Then we summarize the mechanism of the Swing traffic generator tool in Chapter 5. In Chapter 6, we analyze the characteristics of our trace data. Then, the experimental setup and results for the evaluation of the operator cooperation are presented in Chapter 7 and 8. In Chapter 9, we investigate the performance of alternate and future radio access technologies. Finally, we outline future work and conclude in Chapter 10.

Chapter 2

Related Work

Mobile and wireless network composition was previously proposed in the European *Ambient Networks* (AN) project [20]. The vision of AN is to allow instant composition of networks on demand without any prior configuration between operators. This can be done by establishing so-called a *Composition Agreement* (CA) dynamically. By establishing this agreement, services and resources can be shared across networks, and the compensation of usage can be regulated. User devices are also considered as an individual network, therefore, allowing users to connect to any network anywhere at any time. The AN project delivered a system framework design [17] as well as validation and evaluation of the framework [27, 15].

As a part of AN project evaluation, Pöyhönen et al. [21] investigated the benefits of operator cooperation by using performance metrics for both operators and end-users. The analysis shows that cooperation can provide better service availability and quality, leading to more happiness in both users and operators.

Johansson et al. [16] analyzed the cost and performance of deploying heterogeneous access networks. The results show that the deployment cost can be significantly reduced and the radio resources can be utilized more effectively. The results from AN project [15] also show that all operators can save their cost by using multiple radio access technologies and supporting cooperation with other operators.

There have already been numerous studies on the performance and usage of

WiFi [3, 7] and 3G [23, 19, 22, 14] in real environments for the past several years. Nevertheless, the performance evaluations of LTE are still limited. For example, Dahlman et al. [10] evaluated the performance of LTE compared to WCDMA and HSPA in a simulation environment.

Chapter 3

Background

3.1 3G/UMTS Network Architecture

The 3G/UMTS network architecture is presented in Figure 3.1. The network can be divided into two main parts: *Radio Access Network* (RAN) and *Core Network* (CN).

An operator can choose to maintain both *UMTS/HSDPA RAN* (UTRAN) and *GPRS/EDGE RAN* (GERAN) for backward compatibility of GSM devices, called *Mobile Stations* (MS). In GERAN, the user terminal connects to one of the *Base Transceiver Stations* (BTS) which are controlled by a *Base Station Controller* (BSC) forming a *Base Station Subsystem* (BSS). On the other hand, in UTRAN, the terminal *User Equipment* (UE) connects to one of the base stations (*Node B*'s) via WCDMA (Wideband Code Division Multiple Access) wireless interface which supports downlink and uplink capacities upto 384 and 128 Kbps respectively. Some modern UEs and networks also support *High Speed Packet Access* (HSPA) enhancement to WCDMA, allowing UE to achieve up to 14.4 Mbps in downlink speed.

The geographical area served by a Node B is considered as a *cell*. In practice, the coverage of a single Node B can divided into *sectors* to fit the area, providing more efficient overage. One Node B typically can be configured to serve up to six sectors. A group of Node B's are controlled by a *Radio Network Controller* (RNC) forming a *Radio Network System* (RNS). RNSs are interconnected to form

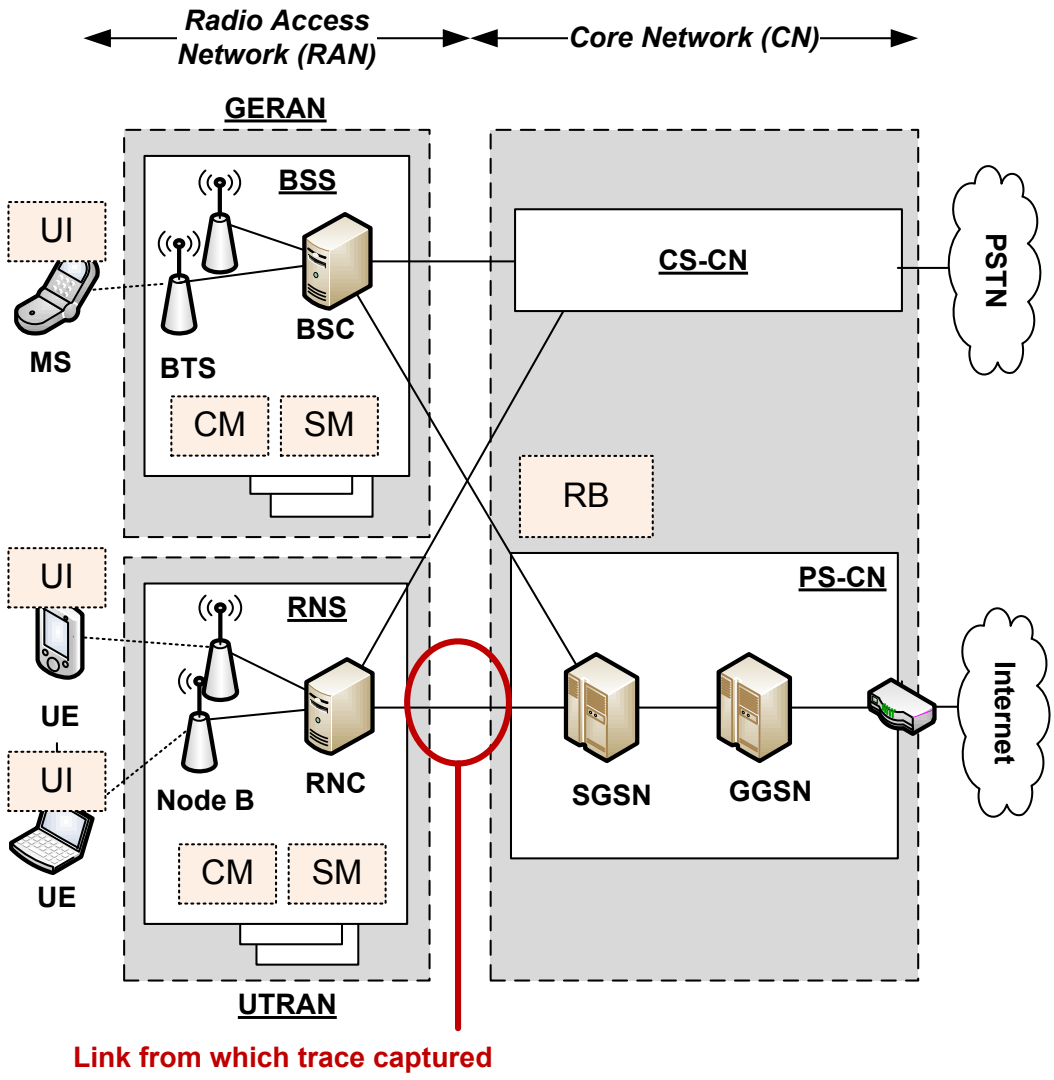


Figure 3.1: 3G/UMTS network architecture and location of resource reservation components

the UTRAN. An RNC is responsible for radio resource management including UE connection and handover procedures which might require interaction with other RNCs. RNCs also maintain links to the core network and multiplex data to circuit-switched (voice) and packet-switched (IP) domains in the CN.

The *circuit-switched domain* (CS-CN), which is not the focus of our study, routes voice data to other mobile or landline networks (PSTN). On the other hand, the *packet-switched domain* (PS-CN) routes user IP packets to the external PS networks (e.g., the Internet). PS-CN contains the *serving GPRS Support Nodes* (SGSNs) and *Gateway GPRS Support Nodes* (GGSN). Each SGSN maintains links to RNCs and BSCs in the RAN and performs UE mobility management and access control. When the user wants to transfer data to/from external PS networks, a connection between an SGSN and GGSN will be established. The GGSN then serves as a gateway to external PS networks such as the Internet.

More details regarding the architecture of 3G/UMTS network can be found in [18] and [5].

3.2 Long-Term Evolution (LTE)

LTE is a new radio access standard developed by 3GPP aiming towards mobile broadband 4G. The LTE design aims to deliver the quality level comparable to those of wired networks by providing very high performance and low latency. As a part of LTE requirements, the system should support peak data rates of 100 Mbps and 50 Mbps for the downlink and uplink respectively. Note that this is a requirement for the configuration of one downlink and one uplink antenna in the user equipment. The system allows for speeds upto more than 300 Mbps for a configuration with more antennas. Also, the round-trip time to RAN should be less than 10 ms. LTE also comes with a flat IP-based system architecture which is easy to maintain and be extended. At present, many carriers globally have already started or planed to deploy LTE in the next couple years. Please refer to [11] for more details regarding LTE.

3.3 WiFi

The 802.11-wireless LAN (WLAN), publicly known as WiFi, is one of the most widely-used wireless access technologies nowadays in homes, enterprises and many public areas. The standards supported by most wireless routers are 802.11b and 802.11g which have a maximum range of around 45 meters for indoor and 90 meters for outdoor. The peak physical data rates can be up to 54 Mbps for 802.11g. Also with the new 802.11n standard, the coverage can be increased by roughly a factor of two, and the maximum bit rate can be increased upto hundreds Mbps. The coverage of WiFi or the so-called hotspot can also span many square miles by using several access points with overlapped coverage. Currently, many cities around the world have set up a city-wide mesh network based on WiFi technology. WiFi is supported by a broad range of end-user devices including laptops, PCs, video game consoles, mobile phones, portable media players, etc. Due to the standardization and popularity, WiFi is easy to deploy and the cost of deployment is becoming lower. Nevertheless, the main issues of WiFi include the range and mobility support.

Chapter 4

Resource Reservation Framework

Al-Fares et al. [4] previously proposed a resource reservation framework to enable network composition across multiple wireless access technologies and operators. This framework was used in our experiments to allow users to switch across multiple access networks dynamically. The proposed framework is based on the GENI resource reservation system documented in [24]. GENI [13] is a network platform designed to support large-scale networking experimentation for the future Internet. The proposed GENI resource reservation system uses a capability-based mechanism to access or gain control of resources. In order to access the resource, the user must first have credits which are represented by cryptographically signed *Tokens*. Tokens can be exchanged for a signed *Ticket* which is a promise or a lease to access concrete resources. The value of the token is abstracted to provide flexibility of interpretation. Additionally, any resource is also abstracted and described by an XML-style document called *RSpec*. This allows the framework to be adaptable to various kinds of policies.

4.1 Components

This section summarizes main components of the framework and their interactions as shown in Figure 4.1.

- **Resource Broker (RB)**: The resource broker receives resources donated by

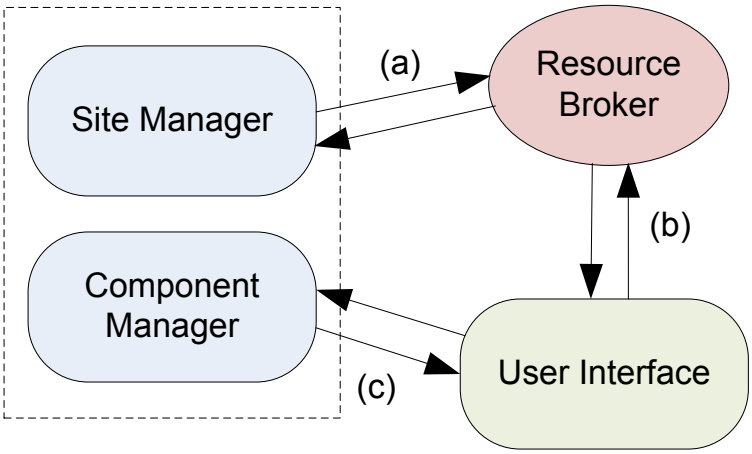


Figure 4.1: GENI resource reservation components which consist of following interactions: (a) resource donation, (b) ticket request and (c) ticket redemption

site managers and grant them tokens in exchange as shown in Figure 4.1(a). The value and the number of tokens granted are based on the policies. The resource broker also matches the request from the user with an appropriate resource and issues the ticket to the user based on tokens.

- **Component Manager (CM)**: The component manager receives the ticket issued by resource broker from the user. It will initiate the resource and grant user an access to the resource specified in the ticket.

- **Site Manager (SM)**: The site manager continuously monitors its local resources and donates available resources to the resource broker in exchange for tokens. The site manager then distributes tokens to users upon demand. Note that a component manager is also co-located with each site manager to keep track of the usage of donated resources.

- **User Interface (UI)**: The user interface provides an interface for users to access the resource. It consists of a *resource discover* mechanism to find wanted resources. When the user wants to acquire the resource, the user interface component first exchanges its tokens for a ticket through the resource broker (Figure 4.1(b)). Then it can redeem that ticket at the component manager to access the resource (Figure 4.1(c)).

4.2 Resource Allocation Framework Mapping

In this section, we show how Al-Fares et al. integrate GENI resource reservation components to the operational 3G/UMTS and WiFi networks.

4.2.1 3G/UMTS Network Mapping

As shown in Figure 3.1, the user interface component can be embedded in the mobile station's USIM/SIM. It will be responsible for resource discovery and managing token balance and tickets. The site manager and component manager would be co-located at each RNC and BSC. The site manager will keep monitoring resources such as available bandwidth at each RNS/BSS and donating available resources to the resource broker periodically. The resource broker is situated in the core network to handle donations from RNS's/BSS's and issue tickets to users connected to those RNS's/BSS's.

The control channel for these components can be separated from normal data channel and can be network dependent. The resource broker must also be able to contact the home operator of the visited users via the Internet to handle billing and invalidate tokens.

4.2.2 Wi-Fi Hotspot Mapping

Figure 4.2 shows how resource reservation components can be integrated into the Wi-Fi hotspot scenario. The user interface is still implemented on the user's devices. The site manager and component manager will be embedded in the access point router. The access control of users can be dynamically configured by manipulating firewall rules in the access point. Finally, the resource broker resides at the core network of the service provider and is responsible for one or more access points. The location of the resource broker has been changed from the original design proposed by Al-Fares et al. [4] to make it similar to the 3G/UMTS topology.

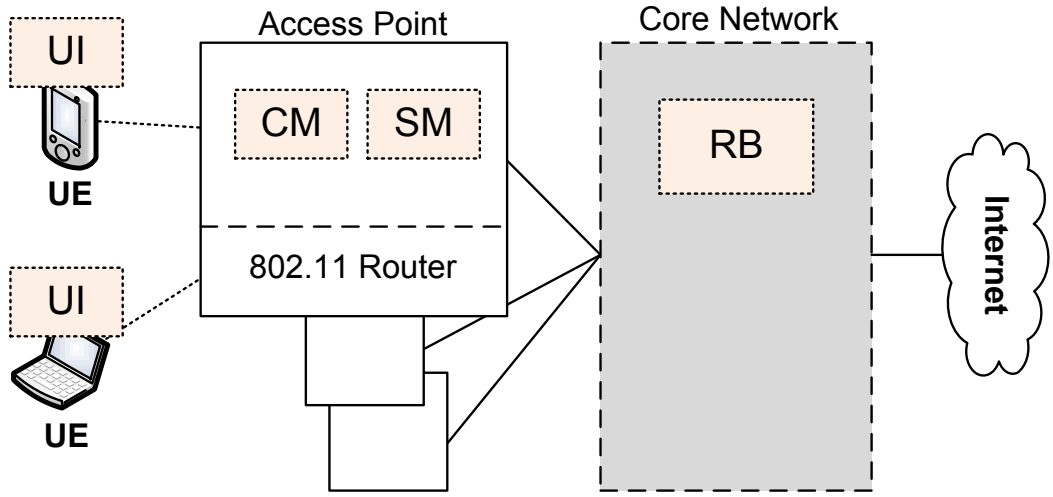


Figure 4.2: Wi-Fi hotspot architecture and location of resource reservation components

4.3 Policy Issues

With regard to token and billing policies, each operator can generate its own tokens which are given to users when they pay their bill. When the users need to visit another service provider's network, they can use their home operator's tokens to pay for the access. Then that service provider will contact and bill the user's home operator who issued the tokens. To insure about the security, all donations, tickets and tokens must be cryptographically signed by their issuer's private key.

4.4 Implementation

Al-Fares et al. implemented a prototype of all components in Java. The Apache XML-RPC library is used as a communication medium between components. Additionally, SHA1-DSA 1024-bit keys are used for cryptographic signatures.

Chapter 5

Swing Traffic Generator Tool

In our experiments, we used the *Swing* traffic generator tool [26] to generate the network usage based on original traces. Swing was proved to be able to accurately capture and reproduce traffic burstiness across a wide range of timescales. It also provides an ability to tune traffic characteristics and network properties to emulate user's custom scenarios. The traffic is generated by using actual TCP connections, so it is responsive to the user's custom network conditions.

In this chapter, we summarize how Swing generates the traffic model based on the original trace, how Swing captures the original network characteristics and how Swing reproduces the traffic in the emulated environment. We refer readers to [26] for more details.

5.1 Traffic Model

Swing uses a structural model to capture traffic characteristics across multiple layers including user, application and network layers. From traffic traces, Swing analyzes the traffic separately for each application class. Swing relies on port-based classification to assign packets into application classes. For each application class, Swing groups packets into flows based on TCP connection semantics. Given flow information, Swing can create distributions of packet sizes, number of request/response pairs and server think time for each request/response.

From the list of flows per application class, Swing now can extract session-

related parameters. Swing defines a session as a set of one or more *Request/Response Exchanges* (RREs). An RRE consists of one or more flows or connections. A new connection which starts within 30 seconds after the last connection ends will be grouped into the same RRE, otherwise that connection will start a new distinct RRE. If the gap or idle period between connections is more than 5 minutes then the new connection will be categorized as a new session. From the session information, Swing can create distributions of number of connections per RRE, time between start of connections, number of RREs per session, time between RREs in session, number of sessions and time between sessions.

Finally, Swing extracts network characteristics per host which consists of three following parameters:

- Link delays: Swing measures time difference between a TCP packet sending to the client and its corresponding ACK. Assuming that the link is symmetric, the link delay is half of this time difference value. Swing uses the median delay of each host to represent the host's link delay.

- Link capacities: Swing utilizes a variation of packet dispersion or packet-pair technique [12] to estimate link capacities. The time difference between packets can be used to calculate the bottleneck capacity by using the formula:

$$BottleneckLinkCapacity \times TimeDifference = PacketSize$$

  Because of the delayed ACK behavior of TCP, half of the packets are believed to be sent back to back. Therefore, Swing ignores the lowest half of the time difference values and uses the median of the rest to estimate the link capacity.

- Link loss rates: Swing employs a simple algorithm based on [6] to estimate the link loss rates by detecting packet retransmissions.

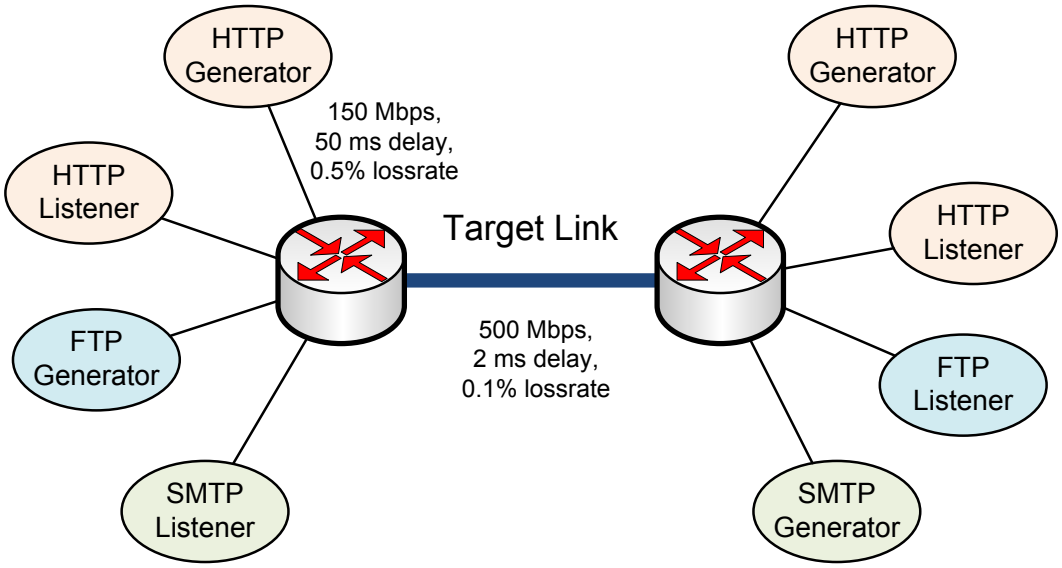


Figure 5.1: Swing dumb-bell topology

5.2 Traffic Generation

5.2.1 Emulated topology

Swing generates a simple dumb-bell topology for emulation as shown in Figure 5.1. The topology consists of a *target link* which corresponds to the link from which the original trace were captured. Nodes or endpoints are attached to both sides of the target link, representing hosts in the original trace. In the ideal case, Swing should create a node per each IP address in the trace. Nevertheless, due to the capacity of emulation environment, Swing collapses multiple hosts in the original trace into around 1,000 nodes in the emulated topology. Also, each node will be responsible for generating traffic for a certain application class only. Finally, Swing assigns link characteristics including bandwidth, latency and loss rate to links connecting the nodes to the target link. These characteristics are based on distributions extracted from the original trace. Note that Swing estimates and assigns only the uplink bandwidth for each host. This would represent the bottleneck link between each pair of hosts.

Swing then executes custom *generators* and *listeners* on each node of the topology. Generators are responsible for initiating a connection to the listeners.

Now Swing will create a list of sessions for those generators/listeners to produce. For each session, Swing randomly selects session's parameters from the distributions. First is to pick the number of RREs and gap between RREs. For each RRE, Swing selects how many parallel connections to which server along with time between the start of each connection. For each connection, the number of requests/responses, packet sizes and server think time would be picked. The sessions along with their parameters will be randomly assigned to generators to generate.

5.2.2 Emulation using ModelNet

Swing utilizes ModelNet [25], a network emulator, to emulate the traffic across end nodes in the emulated dumb-bell topology. ModelNet consists of physical edge machines running multiple end-node applications, which are the generators and listeners in our case. The benefit of ModelNet is that it allows unmodified user applications to run on the end nodes. The packets from every node are forced to route through a ModelNet core machine which emulates network characteristics as in the target topology. ModelNet core operates in real-time by moving packets from queue to queue to emulate per-hop link bandwidth, latency and loss rate. The packet is then forwarded to the edge machine which runs the destination node. Previous work [25] shows that ModelNet can emulate the traffic accurately up to 1 Gbps using a single core machine.

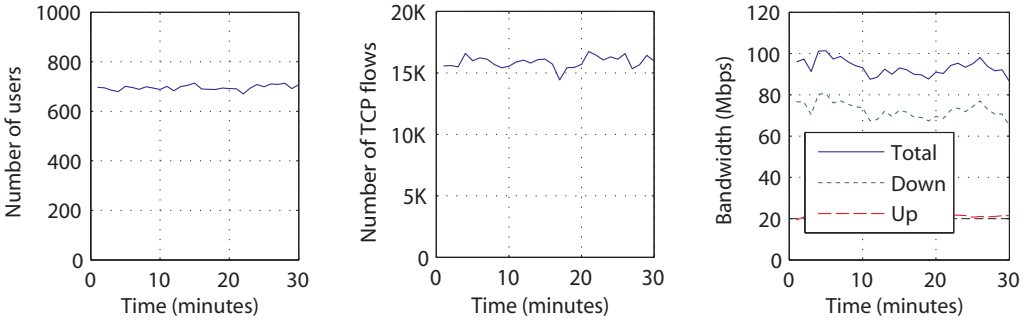
Chapter 6

Trace Data

6.1 Overview

The traces we used in our experiments were captured live from an interface between an RNC and an SGSN in a commercial 3G/UMTS network during Winter 2008. This RNC is connected to approximately 150 Node B's covering the order of 350 sectors. The traces comprise only user-plane data, and all control-plane data is excluded. In addition, the traces include traffic only from 3G PC card devices. To protect the privacy of users whose traffic was captured as part of the trace, the IP addresses were consistently anonymized. More specifically, the same IP in the original trace would always be mapped to the same IP in our trace.

In our experiments, we use 30 minutes of the peak period as original trace data for Swing traffic generation. This consists of approximately 20 GB of traffic in total. 78% of the traffic is the downlink data to mobile clients. Figure 6.1 shows the number of users or unique IP addresses, the number of active TCP flows and aggregate bandwidth at the granularity of 1 minute bins. On average, 696 users sent or received data in each minute, producing around 90 Kbps of aggregate bandwidth usage. Also, approximately 16,000 TCP flows are active in each minute. Note that approximately 5% of traffic is non-TCP traffic and is excluded from Swing traffic generation.



(a) Number of users (c) Number of TCP flows (c) Aggregate bandwidth

Figure 6.1: (a) Number of users, (b) number of TCP flows and (c) aggregate bandwidth usage. All data points are computed at the granularity of 1-minute bins.

6.2 Swing and Trace Data

In order to evaluate the accuracy of Swing traffic generation for our trace data. We ran Swing using the default configuration on the system described in Section 7.4. Note that in this evaluation, we did not classify the traffic into application classes. Instead, we used aggregate trace characteristics in order to maximize the accuracy as mentioned in [26]. Then we compared wavelet scaling plots or energy plots of generated traffic with the original traffic. Energy plots represent byte and packet rates at distinct timescales, showing variance or burstiness in traffic at each timescale. The detailed analysis of energy plots can be found in [2].

As shown in Figure 6.2, Swing managed to preserve the traffic characteristics very well for downlink traffic. Nevertheless, the burstiness of uplink at large timescales was increased from the original trace. This is most likely due to a limitation of Swing and the characteristics of our traces. Since our study focuses only on downlink traffic, it should not have significant impact on the accuracy of our experiments.

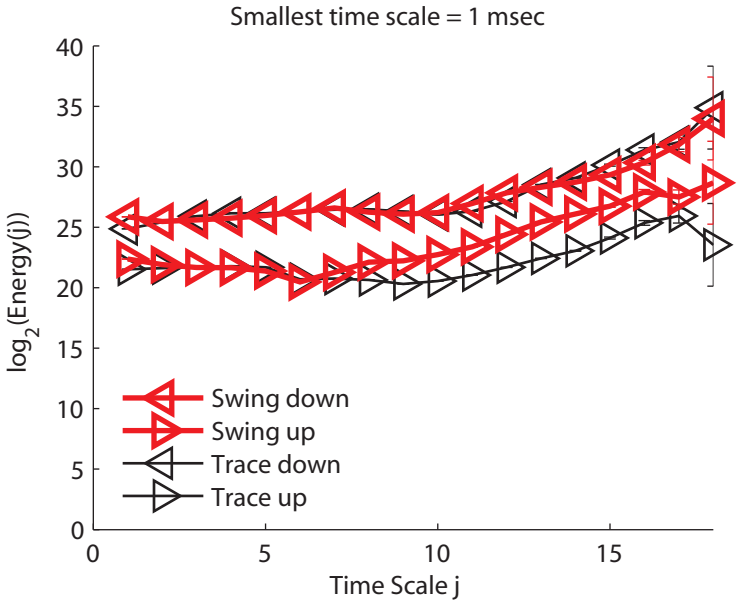


Figure 6.2: Energy plots (bytes) comparing generated and original traffics

6.3 Network Characteristics from the Trace

In this section, we present the mobile client's network characteristics computed from the trace by Swing using the algorithms described in Section 5.1.

Figure 6.3 shows distribution of link delays from client devices to target link. The median of link delays is 96 ms. We attribute the high latency to the 3G wireless link between the client and base station.

Figure 6.4 shows the distribution of estimated uplink and downlink capacities by Swing. We observe that a considerable portion of packet pairs are highly bursty especially downlink traffic, peraphs, due to the queuing scheme in certain links. Therefore, most of the estimated downlink capacities reach Swing's default maximum of 500 Mbps. Nevertheless, as shown in the previous section, Swing can still preserve the traffic pattern very well for the downlink traffic. On the other hand, most of the estimated uplink capacities are close to their median of 400 Kbps which is more similar to the real-world 3G network.

Finally, figure 6.5 shows distribution of link loss rates from clients to the target link. Regardless of the amount of traffic, only around 2% of users show a

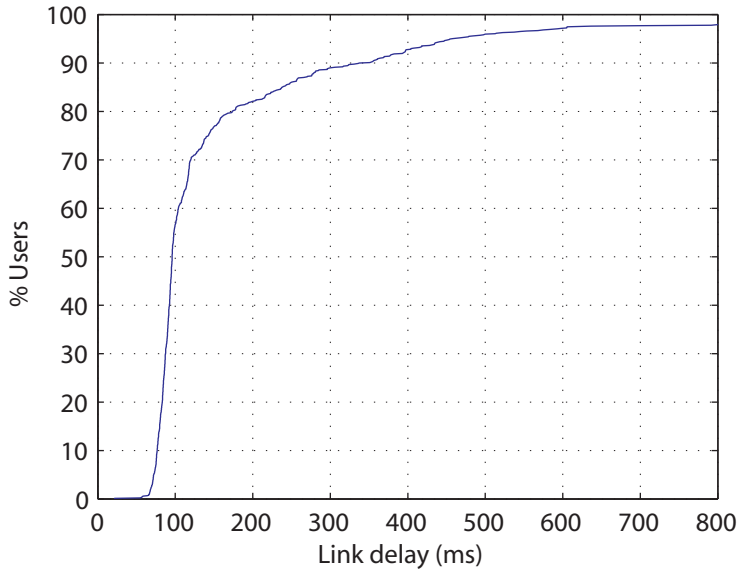


Figure 6.3: CDF of client's estimated link delays

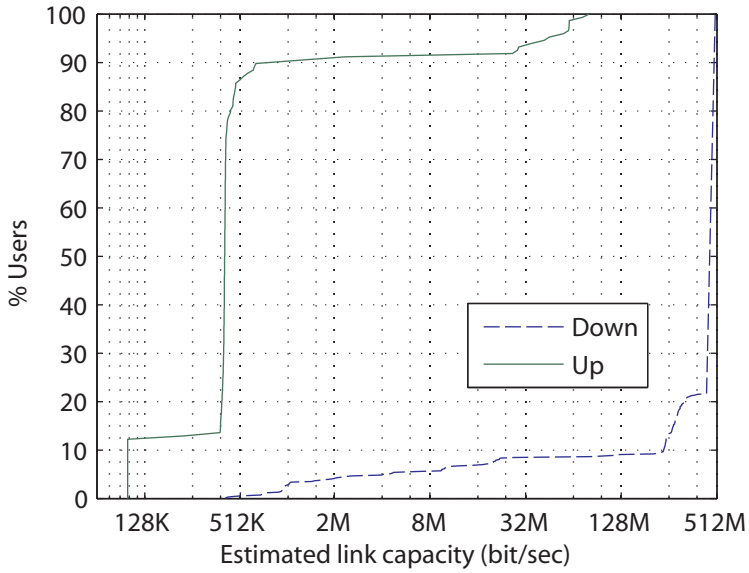


Figure 6.4: CDF of client's estimated link capacities

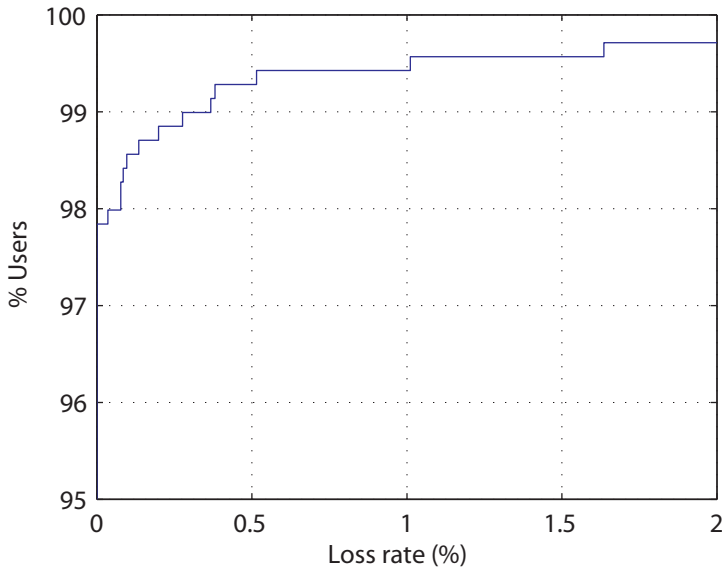


Figure 6.5: CDF of client's estimated link loss rate above the 95^{th} percentile

slight of loss rate percentage, while the others do not experience loss at all. We attribute very low loss rate to the effective loss recovery protocol, namely RLC [1], in 3G wireless link layer and the stationary nature of laptop client.

Chapter 7

Multi-operator Experimental Setup

7.1 Overview

The goal of the following set of experiments is to evaluate various aspects of resource sharing between multiple operators by leveraging a resource peering framework. For the scenario when one operator experiences congestion or receives higher load compared to the others, the benefit of operator cooperation can be obviously perceived by clients. Nevertheless, in a real-world situation, every operator usually experiences the same level of load and traffic pattern in a particular area. We would like to explore the scenario when every operator can still benefit from the resource sharing even if their networks are receiving the same average load and might be experiencing congestion as well.

Due to burstiness in the traffic, the average bandwidth usage may actually be significantly lower than the peak bandwidth usage. With operator cooperation to freely share resources with one another, the operator might be able to offload some of the peak traffic onto other operators which, in turn, have available bandwidth at that time due to the traffic burstiness or vice versa. Consequently, we expect that operators can provision less bandwidth while the level of quality of service can still be maintained. The resources will also be more effectively utilized.

Additionally, all operators would expectedly benefit from sharing resources with one another.

7.2 Experimental Scenario

In this set of experiments, we use Swing to reproduce 30 minutes of traffic during the peak period. The information about trace data is described in Chapter 6. The environment consists of two operators providing 3G/UMTS service. Each client which is generated by Swing is randomly assigned to one of the operators which acts as their home operator. We assume both operators have the same number of subscribers. Also, in this scenario, we are interested in reproducing aggregate characteristics of the trace, so we do not separate the traffic into application classes. The experiment can be broken down into following cases.

1. The first case is to let Swing's clients generate traffic normally through their home operator using traffic model and characteristics captured by Swing. At this point, there is no bandwidth constraint on operator's networks. The resource sharing or operator cooperation is not employed. The quality of service of users such as the average transfer rate will be measured as a baseline.

2. We then try to limit the downlink capacity of operators to certain values to mimic the situation when both operators are experiencing congestion or when the capacities of operators are limited. Since our traces comprise the aggregate traffic from many base stations at the RNC, our limited bandwidth can represent the limited capacity either at the RAN level or within the operator's core network or both. The quality of service will be measured again to see the impact of constrained capacity. No resource sharing is allowed at this point.

3. The final case is to enable resource sharing between operators while the operator's capacities are limited. In this case, the user can access to any operator based on the cooperation policy. Now we can see how much the

quality of service has been improved or worsened from the first two cases. Also, the amount of traffic being offloaded from the home operator onto another service providers can be evaluated. Finally, we can also analyze the impact of control traffic including the increase in user's latencies, the amount of traffic required and the scalability of resource sharing framework.

The policy of resource sharing for our scenario is that users will connect to their home operator's network until its available bandwidth is below a certain threshold. This threshold is set to 20% of total capacity at this point. Then the user will access the operator who has more available bandwidth. Also, we are interested in only available downlink bandwidth which tends to be the bottleneck in the real network.

Operator selection is done every time a client begins a new Swing session (please refer to Chapter 5 for the definition of a session) and the operator will be fixed for that entire session. Particularly, before a client can initiate a new session, it must contact the resource broker to exchange its token for a ticket. We assume that at startup clients already have enough tokens. These tokens can be purchased in advance in a real situation. We also assume that clients know the location of resource broker from the beginning. Practically, this can be known by resource discovery mechanism implemented in the user interface component. Also, in our scenario, we assume that the resource broker is shared between two operators. The resource broker will decide the operator for the client to access based on the implemented sharing policy. It then issues the ticket specifying the operator and its component manager's location to the client. The client then redeems the ticket with the component manager and begins sending the traffic. Additionally, each operator must have a site manager which keeps monitoring the resource (available bandwidth), and donates its resources to the resource broker every half a second. We then can exploit the difference in burstiness down to half-second timescale. All control traffic is separated into dedicated control channels, so it will not interfere the normal data traffic.

Currently, we do not focus on the value of tokens and assume that a ticket allows the client to access the network for the entire session length with no quality

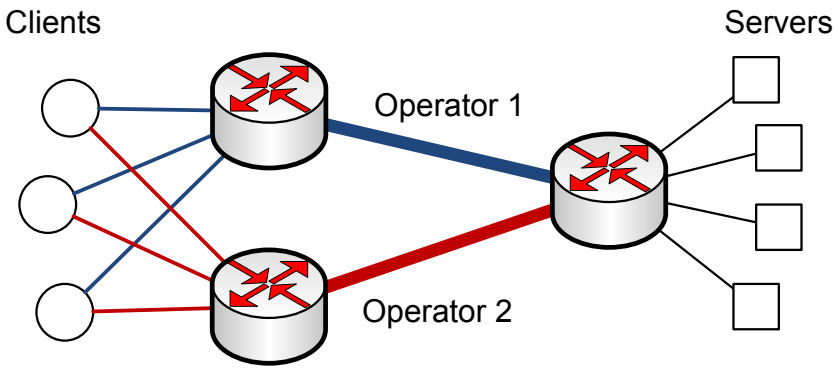


Figure 7.1: Two-operator topology

of service guaranteed (best-effort QoS). The bandwidth will be shared with other clients fairly.

7.3 Implementation

This section describes the modification of Swing and ModelNet as well as any additional implementation to perform the experiments described in the previous section.

7.3.1 Multi-operator Topology

From the Swing dumb-bell topology in Figure 5.1, the links connecting end nodes on one side to the target link represent the path from mobile clients to RNC. On the other hand, the links connecting end nodes on the other side represent the path from SGSN to destination hosts connected to the Internet. The target link itself represents a link between RNC and SGSN of the operator. In our scenario, we assume the congestion happens at the RNC level or the core network level. Therefore, according to the topology, we could think of the target link as a bottleneck link. Consequently, for the two-operator scenario, we can add another target link to represent a second operator as shown in Figure 7.1. One end of the new target link represents the RAN of the new operator. The other end can be

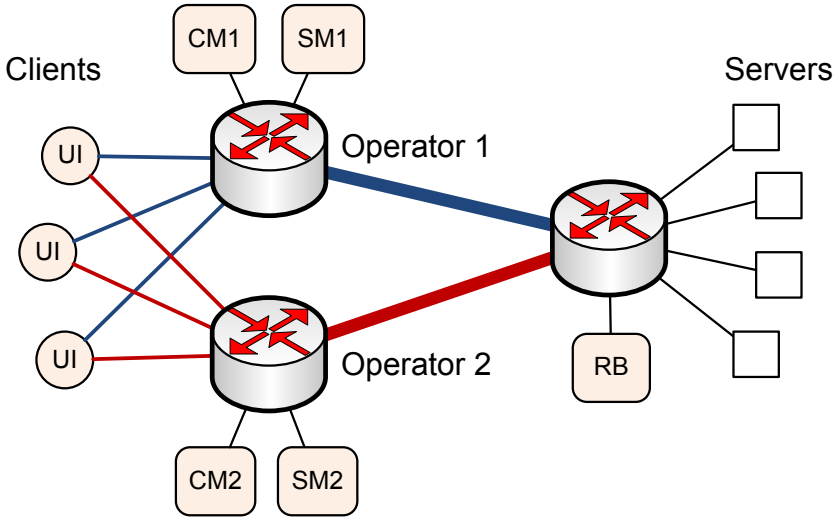


Figure 7.2: Two-operator topology with resource reservation components

shared with the first operator, assuming both operators have the same latency from the core network to the Internet. Every client node will have an additional link connected to the new operator, creating another possible path to all destination servers. By assuming the distance from clients to base stations and the load on base stations of both operators are same, the characteristics of links connecting to the new operator will be the same as those of the first operator. Additionally, since Swing tends to overestimate the link capacity, we cap the maximum downlink and uplink capacities of clients to 3.6 Mbps and 512 Kbps respectively in order to make the client's transfer rate more similar to the real-world scenario.

7.3.2 Resource Reservation Framework Integration

In order to support cooperation between operators, the resource reservation framework components described in Chapter 4 are integrated into the topology as shown in Figure 7.2. User interface (UI) components are located at each client's node. For each operator, a site manager (SM) and a component manager (CM) are connected to the RAN side of the target link. The resource broker is attached to another side of target link which represents the core network. Additional links are added to connect these components together along with the normal data links

to carry control traffic separately.

Generally Swing assigns both generators and listeners to both sides of the target link. So, generators and listeners on the client's side are modified to imitate mobile clients in the 3G scenario. Each of them will have a user interface component running along with it. Before the generator initiates a normal session to the server on another side, it will make an XML-RPC call to its user interface component via loopback interface. Then the user interface will handle the resource reservation procedure and pass control back to the generator to start the session. For those listeners on the client's side, they do not initiate the session by themselves. Therefore, when the listener receives a connection of a new session, it will block that session and carry out resource reservation procedure like the generator does. Once the reservation process is done, it will continue the session.

Each site manager is programmed to monitor the available bandwidth of its operator's target link and donate it to the resource broker every half a second. In order to monitor the bandwidth of the link in the emulated topology, a special process must be running on the ModelNet core machine. This process will periodically monitor the available bandwidth through a ModelNet interface and write the results to a file for the site manager components to read. Finally, the component manager is responsible for executing a command to switch the operator associated with the client who redeems the ticket.

7.3.3 ModelNet Modification

Originally, ModelNet uses a set of fixed routing paths between every pair of nodes. These paths are computed and loaded when the topology is deployed at startup. In our scenario, however, each client has more than one path to each server. Each path corresponds to each operator. So, we need an ability to change the path or operator during runtime. We modified ModelNet to load multiple sets of paths for every pair of nodes. Then a new ModelNet interface is implemented for selecting the path or operator for each pair of nodes to use. At startup, each client will be set to use the path of its own home operator. Then during runtime, the component manager can use the interface to change the path or operator of

each pair of client/server dynamically.

7.4 System environments

We run Swing and ModelNet on a cluster of eight machines. One machine is setup as a ModelNet core, while the rest are edge machines running applications on end nodes of the emulated topology. All machines are powered by Intel Xeon 2.8 Ghz with 2 MB of memory and a Gigabit NIC. The operating system that we use is CentOS 5.2 (Linux 2.6.18). Each edge machine is responsible for running approximately 150 to 250 end nodes.

Chapter 8

Multi-operator Experimental Results and Analysis

This chapter presents the results from the experimental setup in the previous chapter. First we explore the traffic and bandwidth usage for each operator in all experimental cases. Then we analyze the quality of service when operators can cooperate compared to when there is no cooperation. Also, the overheads of resource reservation system are investigated. Finally, we discuss about alternate potential scenarios, configurations and policies for operator cooperation.

8.1 Traffic

Figure 8.1 shows the aggregate downlink bandwidth at the target link for both operators at the granularity of 1-second bins. The average bandwidth for both operators are 33.8 and 33.5 Mbps. Although the average usage is roughly the same, we can still see the difference in burstiness in smaller timescales. From the time series of bandwidth usage, we create the distribution of bandwidth usage values at each second. Then we cap the downlink capacities of both operators to 80th, 90th and 95th percentiles which are approximately 37, 39 and 41 Mbps respectively.

Figure 8.2 shows the aggregate downlink bandwidth when the capacities are capped to 37, 39 and 41 Mbps. For all cases, we can still see some certain

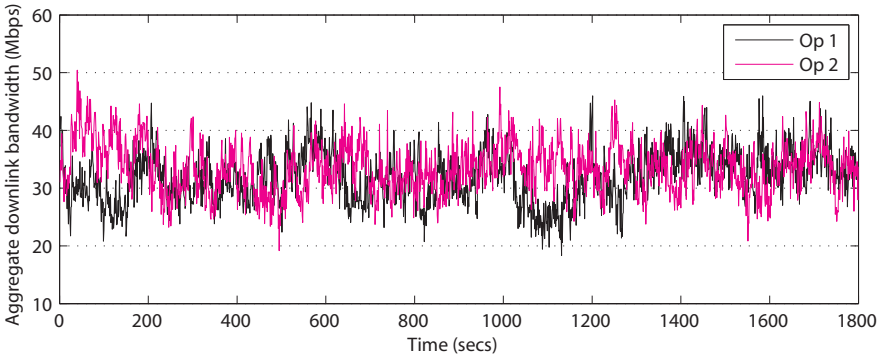


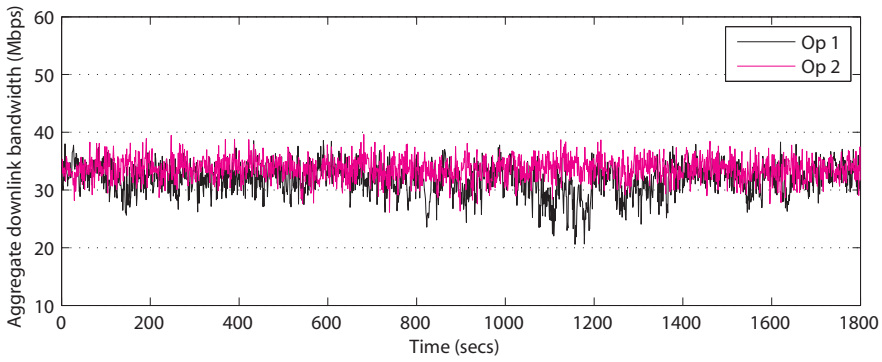
Figure 8.1: Aggregate downlink bandwidth for each operator every 1 second when there is no capacity constraint

periods when the usage of one operator reaches the limit while the other does not.

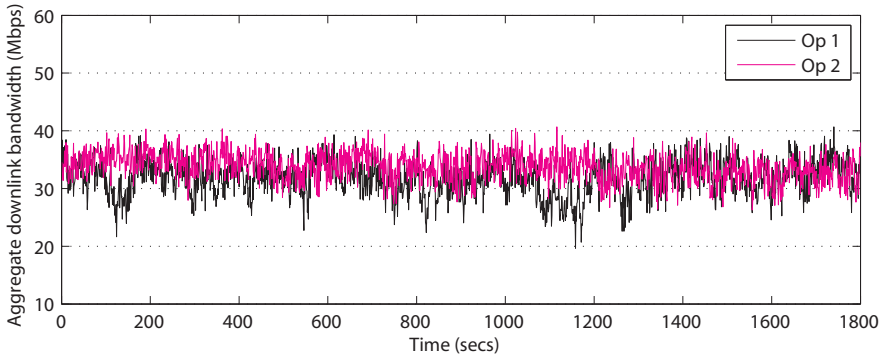
When the resource sharing is utilized, the usage is distributed fairly equally to both operators as shown in Figure 8.3. As expected from the sharing policy, when the traffic of one operator is greater than the other one, it will be offloaded to another operator. For the 37-Mbps cap, the usage of both operators becomes close to the cap almost all the time which means the cap is relatively too strict. There might be many periods when the combined capacity is not enough to support all the usage during the peak burstiness in the aggregate traffic.

The summary of the amount of traffic passing through each operator when the operator sharing is supported is shown in Table 8.1. Each entry represents the amount of traffic belongs to the original or home operator (rows) and the real access operator (columns). For all cases, the amount of traffic belongs to each home operator is roughly the same. The amount of traffic which is being switched to another operator is highest for the case of 37-Mbps cap because the network is congested the most in this case. Then the amount of switched traffic becomes less when there is more available bandwidth like in the other two cases.

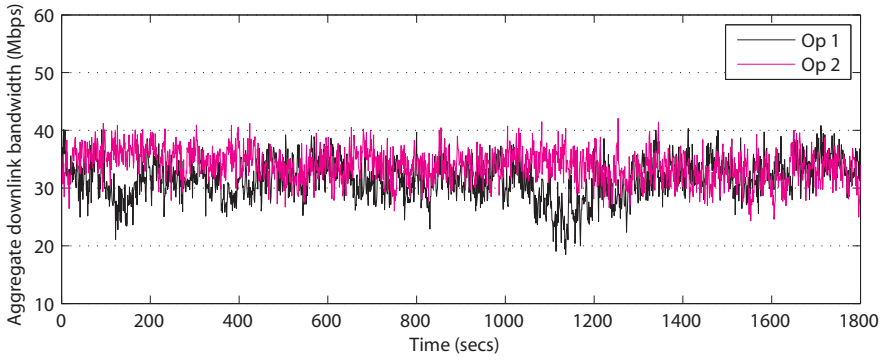
In a real scenario, when the user uses a service of another provider, the home operator needs to pay that provider for the network usage. Nevertheless, even though our policy can cause traffic to unnecessarily switch to another operator, both operators still receive the same amount of traffic in the end. Therefore, no operator is at disadvantage at this point.



(a)

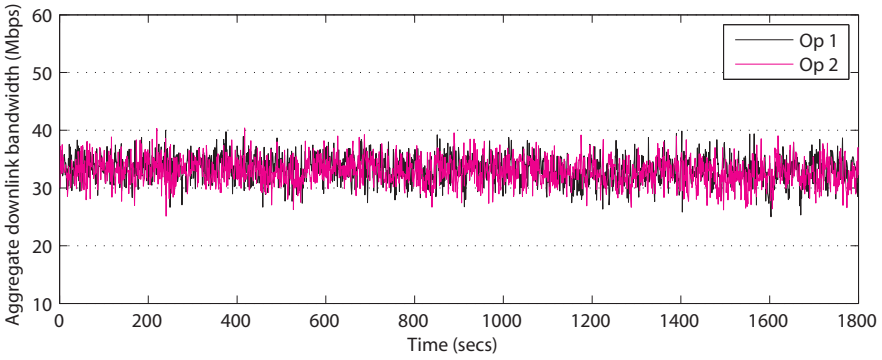


(b)

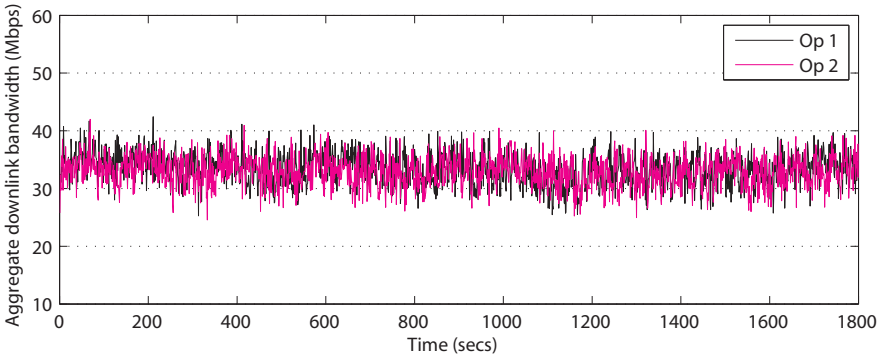


(c)

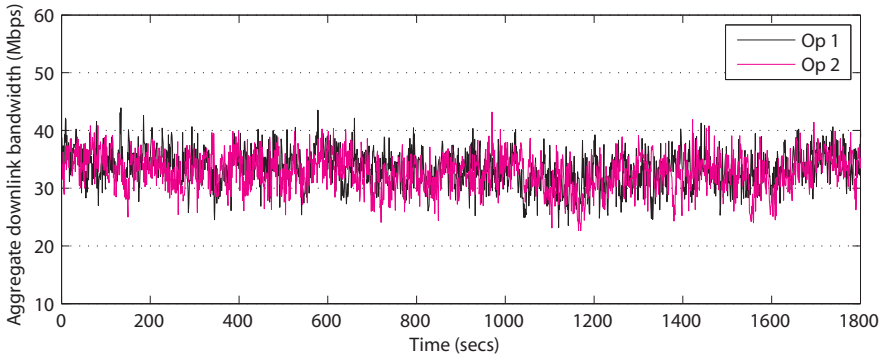
Figure 8.2: Aggregate downlink bandwidth for each operator every 1 second when the capacities are limited to approximately (a) 37 Mbps, (b) 39 Mbps and (c) 41 Mbps



(a)



(b)



(c)

Figure 8.3: Aggregate downlink bandwidth for each operator every 1 second when operator cooperation is supported and the capacities are limited to approximately (a) 37 Mbps, (b) 39 Mbps and (c) 41 Mbps.

Table 8.1: Amount of traffic shared between two operators when capacities are capped to (a) 37 Mbps, (b) 39 Mbps and (c) 41 Mbps.

(a)

	Access Op. 1	**Access Op. 2**	**Total**
Home Op. 1	23.9%	25.8%	**49.7%**
Home Op. 2	26.2%	24.2%	**50.3%**
Total	**50.1%**	**49.9%**	**100.0%**

(b)

	Access Op. 1	**Access Op. 2**	**Total**
Home Op. 1	26.0%	23.9%	**49.9%**
Home Op. 2	23.4%	26.7%	**50.1%**
Total	**49.3%**	**50.7%**	**100.0%**

(c)

	Access Op. 1	**Access Op. 2**	**Total**
Home Op. 1	31.2%	18.9%	**50.1%**
Home Op. 2	18.8%	31.2%	**49.9%**
Total	**49.9%**	**50.1%**	**100.0%**

8.2 Quality of Service

The metric we use to measure the quality of service is the session's average downlink transfer rates. The average transfer rate is computed by dividing total downloading bytes by total active time. The active time of the session is defined as a summation of time between the beginning and ending of all Request/Response Exchanges (RREs) in that session (see Chapter 5). Although there might be some idle period within the RRE such as server think time or time between TCP connections, we believe this is an appropriate metric to represent the actual user's perceived quality of service. Note that for the operator cooperation cases, we do not include latency of resource reservation procedure at the beginning of session to the transfer rate calculation.

For each bandwidth cap level, we compare distributions of transfer rates for the case when operators can cooperate and when they cannot cooperate. The case when operator's capacities are not constrained is considered as a baseline. Figure 8.4 compares the CDFs of average transfer rates for the case when operator's capacities are restricted to 37 Mbps. According to the figure, the operator

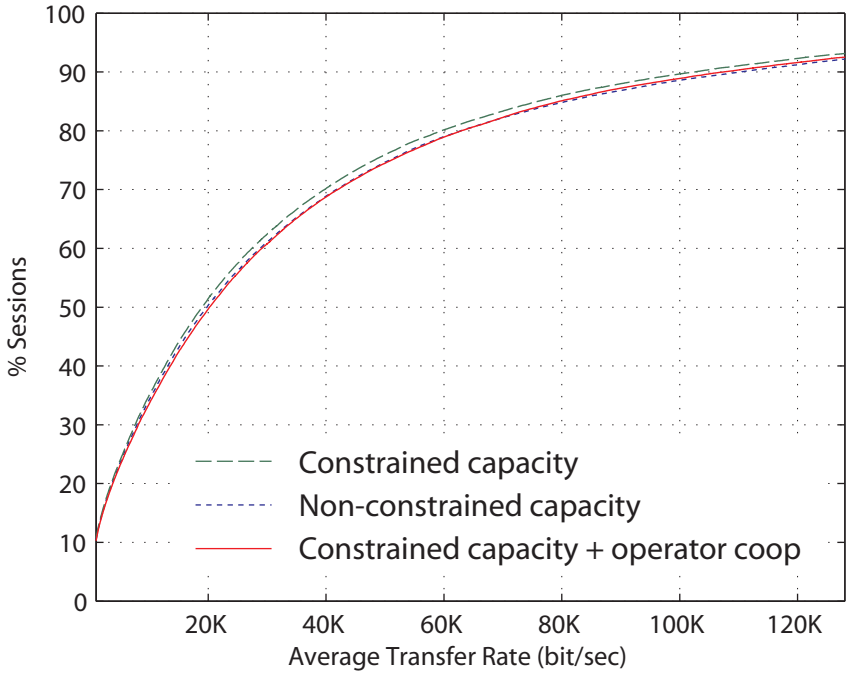


Figure 8.4: CDF comparing session's average transfer rates (Kbps) when the capacities are limited to approximately 37 Mbps

cooperation can make the quality of service comparable to the baseline even the operator's capacity is limited.

In order to analyze the results in more detail, we summarize the transfer rate distributions for all experiments in Table 8.2. Additionally, the percentages of the transfer rates compared to the non-constrained case are shown in Table 8.3. The constraints do not have much impact on the performance for percentiles less than 90^{th}, and the operator cooperation is able to improve the performance up to or even better than the baseline. The reason why it can outperform the baseline is because of that fact that Swing collapsed many clients in to the same node. When some flows on the node are switched to another operator, the congestion on that client's link can be relieved. Nevertheless, we would like to focus on higher percentiles which tend to be impacted by the capacity constraint the most. When the bandwidth cap is strict, the operator cooperation can improve the performance quite significantly such as from 72% to 81% at 99.9^{th} percentiles for the 37-Mbps constraint case. Since there obviously exists the period when the total usage

Table 8.2: Session's average transfer rates (Kbps) at different percentiles

	w/o constraint	**37-Mbps constraint**		**39-Mbps constraint**		**41-Mbps constraint**	
		w/o coop	**w/ coop**	**w/o coop**	**w/ coop**	**w/o coop**	**w/ coop**
25^{th}	5.66	5.51	5.88	5.56	5.99	5.60	6.02
50^{th}	19.81	18.98	20.25	19.23	20.54	19.54	20.62
75^{th}	50.76	48.21	51.11	48.90	51.94	49.72	52.58
90^{th}	110.35	102.57	107.42	104.72	110.25	106.43	112.68
95^{th}	163.21	150.70	156.63	154.51	162.18	157.47	164.57
99^{th}	313.85	278.34	290.26	294.28	307.07	300.66	312.59
99.5^{th}	420.85	353.48	373.73	373.29	390.84	392.75	403.43
99.9^{th}	898.48	646.04	726.40	755.12	787.74	789.26	850.80

Table 8.3: Percentages of session's average transfer rates compared to non-constrained case at different percentiles

	w/o constraint	**37-Mbps constraint**		**39-Mbps constraint**		**41-Mbps constraint**	
		w/o coop	**w/ coop**	**w/o coop**	**w/ coop**	**w/o coop**	**w/ coop**
25^{th}	100%	97%	104%	98%	106%	99%	106%
50^{th}	100%	96%	102%	97%	104%	99%	104%
75^{th}	100%	95%	101%	96%	102%	98%	104%
90^{th}	100%	93%	97%	95%	100%	96%	102%
95^{th}	100%	92%	96%	95%	99%	96%	101%
99^{th}	100%	89%	92%	94%	98%	96%	100%
99.5^{th}	100%	84%	89%	89%	93%	93%	96%
99.9^{th}	100%	72%	81%	84%	88%	88%	95%

Table 8.4: Session's average transfer rates (Kbps) and their percentages compared to the non-constrained case at different percentiles for the first 200 seconds of 37-Mbps constraint case

	w/o constraint	**37-Mbps constraint**	
		w/o coop	**w/ coop**
25^{th}	4.69 (100%)	4.62 (98%)	4.84 (103%)
50^{th}	19.82 (100%)	18.91 (95%)	20.27 (102%)
75^{th}	52.63 (100%)	50.13 (95%)	53.57 (102%)
90^{th}	113.52 (100%)	107.05 (94%)	111.19 (98%)
95^{th}	170.37 (100%)	157.36 (92%)	161.14 (95%)
99^{th}	292.91 (100%)	276.49 (94%)	287.82 (98%)
99.5^{th}	363.07 (100%)	315.65 (87%)	345.7 (95%)
99.9^{th}	902.08 (100%)	618.64 (69%)	753.03 (83%)

exceeds the total capacity combined from two operators, the performance still cannot be improved to the same level as the non-constrained case. Nevertheless, for the case with less strict cap such as 41-Mpbs cap, there is more available bandwidth, so the peak usage can be distributed among two operators better and less congested. As a result, the performance can become very close to baseline case when operators can cooperate.

The benefits of operator cooperation might not be groundbreaking in general because the difference in burstiness might still be not enough in our scenario. However, if we consider the period when the usage of both operators is highly unbalanced such as the first 200 seconds, the performance gain from the operator cooperation can be more obvious. Table 8.4 shows the distribution of average transfer rates of sessions within the first 200 seconds of the 37-Mbps constraint case. At the 99.9^{th} percentile, the performance can be improved considerably from 68% to 83%.

In summary, when the provisioning capacities are limited or when the congestion occurs in the network, the operator cooperation can help improve the performance up to when there is no capacity constraint for most of the user's session. Nevertheless, there is still a trade-off. When available bandwidth is lower, the quality of service will be more impacted even with operator cooperation.

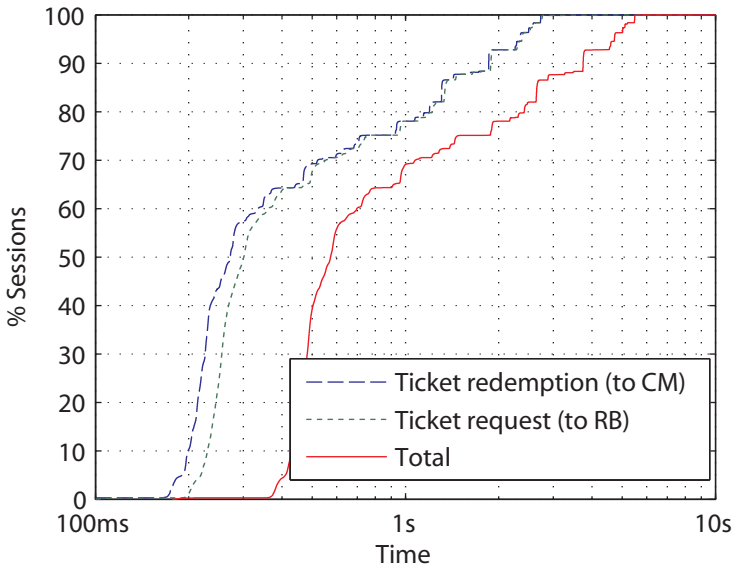


Figure 8.5: CDF of session's resource reservation overheads in log scale

8.3 Resource Reservation Overheads

Figure 8.5 shows CDF of session overheads in milliseconds due to the resource reservation procedure at the beginning of each session for the 37-Mbps constraint case. Note that the overhead distributions are roughly the same for every experiment regardless of the level of constraint because the control traffic has its own dedicated channel. This overhead includes two RPC calls from the client. First, the client must make an RPC call to the resource broker to request for a ticket. Then it will make another RPC call to the component manager to redeem the ticket. The overhead distribution is heavy-tailed with a relatively high median of 573 ms for the total latency.

The average processing time at the resource broker and component manager for each session is only 43 ms in total. This processing time is mostly for signature signing and verification. The remaining factors for the delay include the RPC call overhead and the network latency. Particularly, the 3G mobile client wireless link latency, as shown in section 6.3, seems to dominate the overall delay. Also, in the real system, we expect more latency for the operator to contact the user's home operator to verify and invalidate the token in case of roaming. Nevertheless, the

operator might choose not to wait for this process by assuming users are honest. Also, typically, standard 3G clients already experience a significant connection setup delay. Therefore, our reservation system delay might be able to merge and hide into those existing setup routine to some extent.

Note that the high setup delay can affect the performance of resource sharing since it introduces a gap between when an operator is chosen for the user and when the user can actually access the network. Therefore, the network condition might have changed already. However, in our experiments, we still could not a see significant gain after eliminating the client's link latency for control traffic.

With regard to the resource donation, the average latency for the site manager to make a donation to the resource broker is 32 ms with relatively low deviation since these components are connected by wired links.

The total amount of client's resource reservation control traffic is only about 0.8% of data traffic on average. The amount of traffic required for resource donations is negligible in our scenario. In the real scenario, the network scale can be significantly expanded. The component manager and site manager are distributed and located at each local site or RAN, so the scalability should not be an issue. The resource broker, which both processes donations and matches requests from users with the resource, can be a bottleneck. Nevertheless, it is possible to deploy multiple resource brokers, each of which is responsible for a specific set of sites, and let them exchange information when necessary.

8.4 Limitations

Our emulation topologies are fixed and simplified from the real 3G network. In particular, we cannot emulate real wireless medium characteristics or user mobility. Instead, the characteristics of wireless link will be abstracted by a normal wired link with latency, bandwidth and loss rate estimated from the original trace by Swing. Since our original traces were captured from the usage in the real world, we expect our traffic and topology models to still be very realistic and be comparable to the real wireless characteristics and user mobility.

8.5 Alternate Scenarios

8.5.1 Alternate Cooperation Policies

We also explored an alternate policy which let users switch the operator when its home operator reaches maximum capacity, and another operator still has available bandwidth. Though, in practice, the responsiveness of this policy is not good enough. In particular, the switching occurs when the congestion already happened. Also, sometimes congestion can be hard to detect when the traffic gets bursty in a timescale smaller than the monitoring interval. As a result, this policy is not as effective as we hoped.

8.5.2 Alternate Resource Sharing

In our scenario, we emulate the operator's bottleneck at the RAN or core network level of aggregate traffic. This traffic, therefore, includes the usage from many base stations. Nevertheless, another scarce and very important resource in the cellular world is the radio spectrum at the base station.

Consequently, another potential scenario is to leverage burstiness in traffic to reduce the contention at the base station by shedding the load onto the base station of other service providers within a cell or sector area. We expect to see more potential benefits of operator cooperation from this scenario, especially in the real world, since users tend to experience quality drop due to congestion at base station. Also, we might be able to see more variety in burstiness that we can exploit.

Unfortunately, with our current trace data and emulation tool, we still cannot realistically simulate this alternate scenario. More information about the base station associated with users, mobility model and 3G wireless link model is still required to do so. We leave this as a future work when we can obtain more pieces of information.

Chapter 9

Alternate and Future Access Technologies

In this chapter, we exploit Swing to generate realistic traffic through many types of wireless access networks including 3G, WiFi, and LTE. Then we compare the performance delivered by each kind of access technology for various application classes.

9.1 Experimental Setup

9.1.1 Application Classes

For the experiments in this chapter, we still use the trace data described in Chapter 6 for Swing to generate the traffic. First, we classify the traffic in our traces into one of the following application classes.

- **HTTP**: HTTP and HTTPS protocols. Note that HTTP traffic can also include Flash video streaming under HTTP protocol such as Flash videos on YouTube.
- **Streaming**: RTSP (Real Time Streaming Protocol), RTMP (Real Time Messaging Protocol), RealPlayer, Quicktime, and Shoutcast streaming. Nevertheless, some TCP streaming protocols send the streaming data separately

Table 9.1: Share of different application classes in traffic volume

Application	Fraction of bytes
Streaming	5%
HTTP	34%
P2P file sharing	40%
Other TCP	21%

via UDP which will not be included in the generated traffic.

- **P2P file sharing**: BitTorrent and other P2P file sharing applications such as eDonkey and Gnutella.

- **Other TCP**: any other TCP applications including Email, file transfer protocols, instant messaging protocols, etc. However, these known protocols constitute less than one percent of total traffic. Majority of this class actually comprises traffic in non-standard ports which we failed to classify.

Table 9.1 shows the fraction of the total traffic volume by different applications. Traffic from P2P file sharing constitutes the highest amount of traffic followed by HTTP and Other TCP.

9.1.2 Network Topologies

We used the Swing dumb-bell topology as shown in Figure 5.1 as a base topology. Then we modified the characteristics of links connecting the nodes on client's side to the target link based on the radio access technology. Note that originally Swing only set the uplink bandwidth of both clients and servers to the estimated capacity. By assuming the bottleneck links of network are on client's side, we modified the Swing to estimate and constrain both downlink and uplink capacities of clients. And the link capacities of all server nodes are not constrained anymore.

For the 3G access technology, we used the estimated characteristics from the original trace data as shown in Figure 6.3, 6.4 and 6.5. Similar to the experiment setup in Chapter 7, we capped the maximum bandwidth to 3.6 Mbps and 512 Kbps for downlink and uplink respectively.

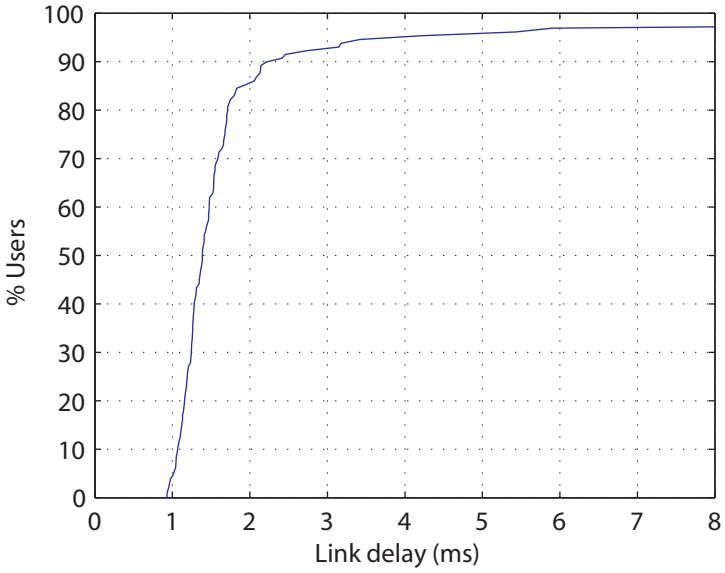


Figure 9.1: CDF of WiFi client's estimated link delays

The WiFi and LTE link characteristics are described below.

WiFi Link Characteristics

We used Swing to estimate WiFi link characteristics from the WiFi tcpdump traces which are a part of Jigsaw project [8, 9]. The traces were captured at the gateway router that interfaces the UCSD campus giga-ether network and the CSE wireless VLAN on Jan 11, 2007 from 1 pm to 4 pm. The traces consist of aggregate traffic from 40 802.11b/g access points covering four floors and the basement in UCSD CSE building. There are only about 46 users using the network on average in every minute, producing an aggregate bandwidth of 2.8 Mbps on average. Figure 9.1, 9.2 and 9.3 show distributions of link delays, capacities and loss rate extracted from the traces respectively. We estimated only uplink capacities and used it for both uplink and downlink because Swing tends to highly overestimate the downlink capacity in this case. The median for link capacity is 18.6 Mbps, and the median for link delay is 1.39 ms. We can see that these characteristics are significantly better than those of 3G wireless access. Nevertheless, approximately 16% of users show a slight of loss rate percentage which is relatively higher than

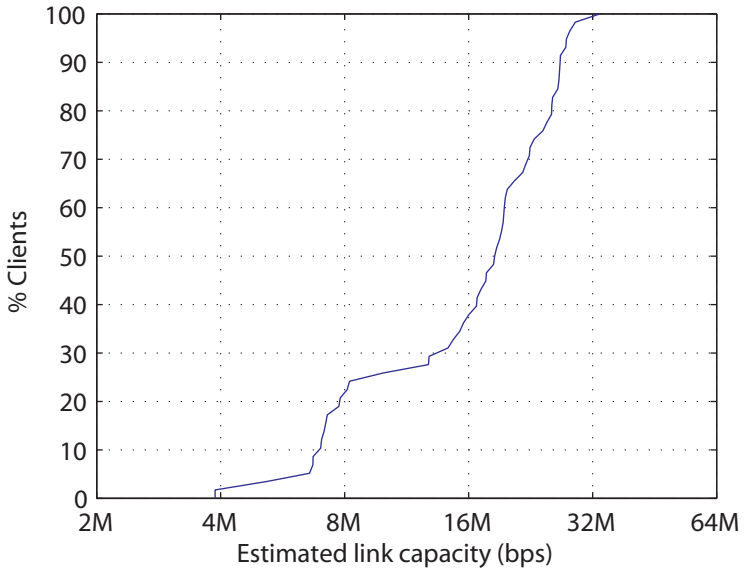


Figure 9.2: CDF of WiFi client's estimated link capacities

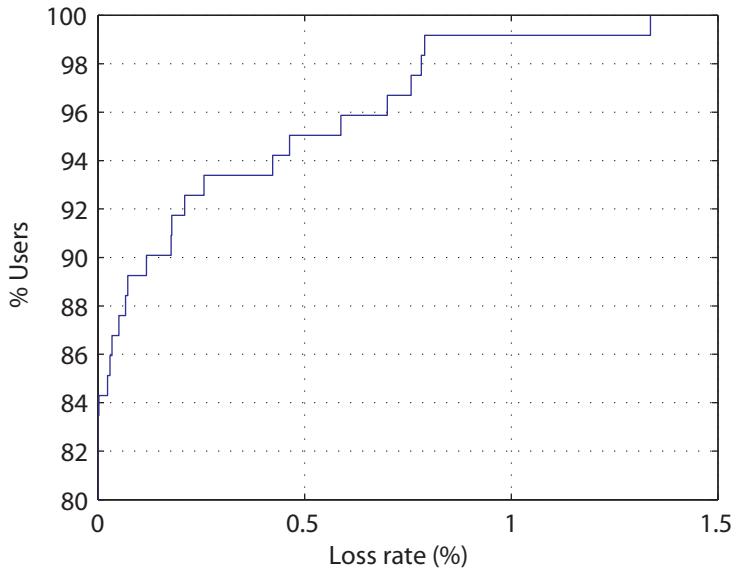


Figure 9.3: CDF of WiFi client's estimated link loss rate above the 80^{th} percentile

3G. We randomly assigned the link characteristic values to clients based on these distributions, replacing those 3G link characteristics from the previous topology. Then we use Swing to generate the same traffic as the previous 3G scenario.

LTE Link Characteristics

Since the LTE technology has not been widely adopted and deployed yet, it is very difficult to obtain any trace data of the real user usage. Additionally, the results of performance evaluations published so far are mostly based on simulations and are very varied depending on the simulation configurations and parameters. Consequently, we chose to estimate LTE link characteristics by scaling the estimated 3G link characteristic distributions based on the 3G and LTE specifications. Regarding the link bandwidth, the maximum downlink bandwidth that mobile clients in our trace can achieve should be no more than 2-3 Mbps due to the early version of HSPA. On the other hand, LTE can support up to 100-300 Mbps in an ideal condition. Consequently, we decided to increase the bandwidth of all 3G clients by a factor of 20 to represent an LTE link bandwidth distribution. For the link latency, 3G WCDMA/HSPA has RAN round-trip time approximately 100-150 ms according to the specifications, while the round-trip time of LTE can be as low as 10 ms. Therefore, we scaled down all 3G link latencies by a factor of 10 to represent the LTE link latency distribution.

9.2 Experimental Results and Analysis

Figure 9.4, 9.5 and 9.6 show distributions of session's average transfer rates by applications for 3G/UMTS, WiFi 802.11b/g and LTE access technologies respectively. The sample values of all distributions at certain percentiles are shown in Table 9.2, 9.3 and 9.4. The results show that the average transfer rates of 802.11b/g are roughly double of those of 3G. The differences are even greater for higher percentiles. This is due to a big difference in both latencies and capacities of clients between two access technologies. In particular, the median link latency of WiFi is lower than that of 3G by almost two orders of magnitude. Also, the

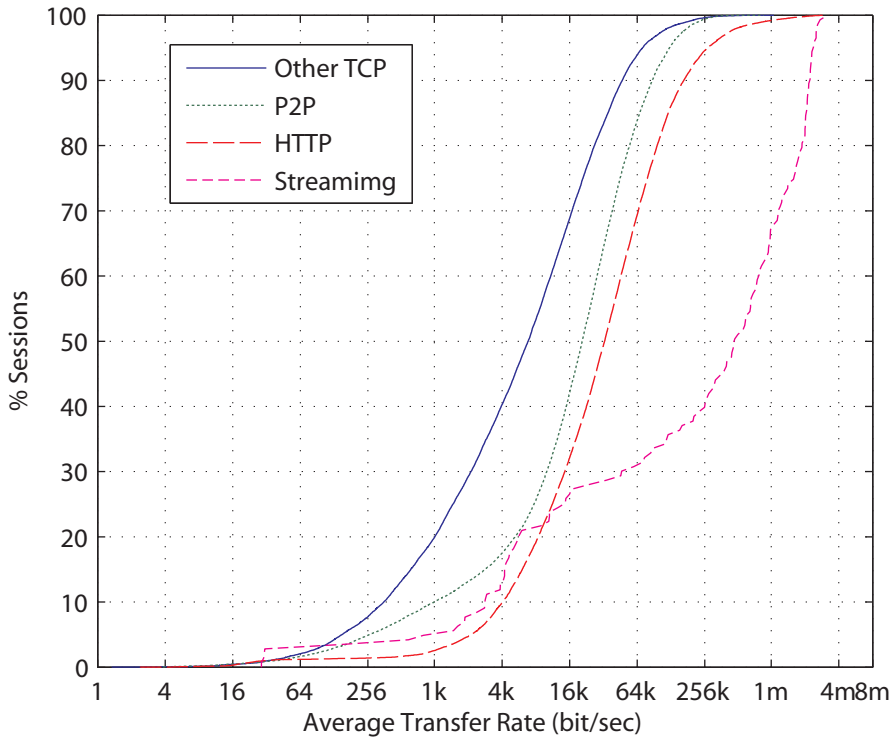


Figure 9.4: CDF of session's average transfer rates by applications for the 3G/UMTS access technology

Table 9.2: Distributions of session's average transfer rates (Kbps) by applications for the 3G/UMTS access technology

	HTTP	**P2P**	**Streaming**	**Other TCP**
25^{th}	11.2	7.7	14.7	1.5
50^{th}	32.8	20.7	473.1	6.8
75^{th}	78.9	45.1	1604.9	21.0
90^{th}	166.3	87.1	2219.0	47.1
95^{th}	272.1	123.8	2381.6	71.4
99^{th}	851.7	225.3	2756.1	178.6

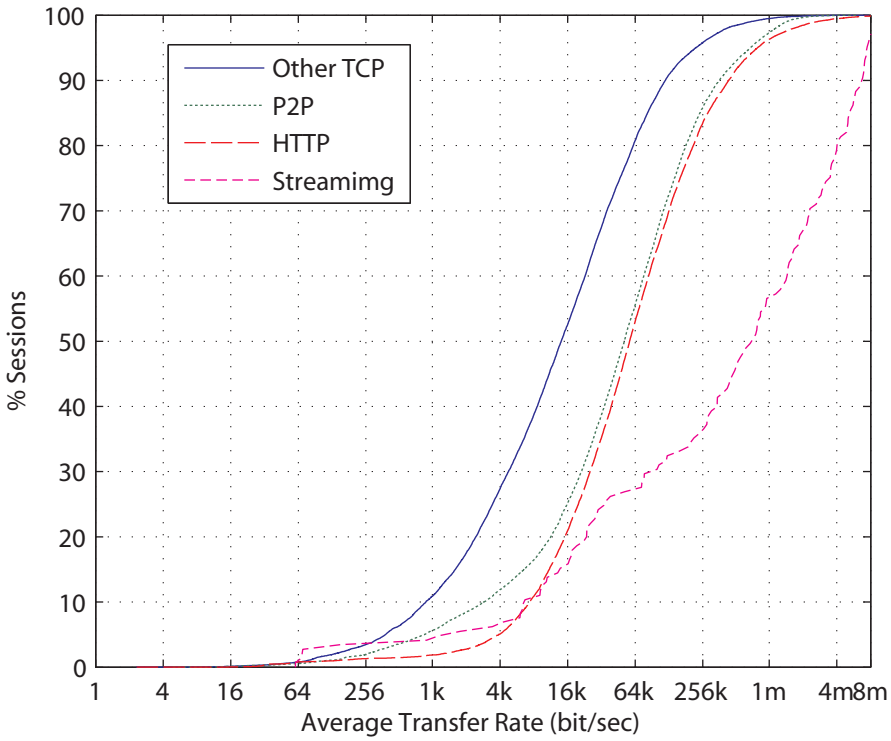


Figure 9.5: CDF of session's average transfer rates by applications for the WiFi 802.11b/g access technologies

Table 9.3: Distributions of session's average transfer rates (Kbps) by applications for the WiFi access technology

	HTTP	**P2P**	**Streaming**	**Other TCP**
25^{th}	20.0	15.9	34.8	3.4
50^{th}	57.0	51.3	714.9	14.0
75^{th}	160.4	143.4	3502.8	46.8
90^{th}	434.6	358.8	6783.0	118.6
95^{th}	785.5	678.5	7510.9	221.9
99^{th}	2661.5	1422.0	10267.1	684.0

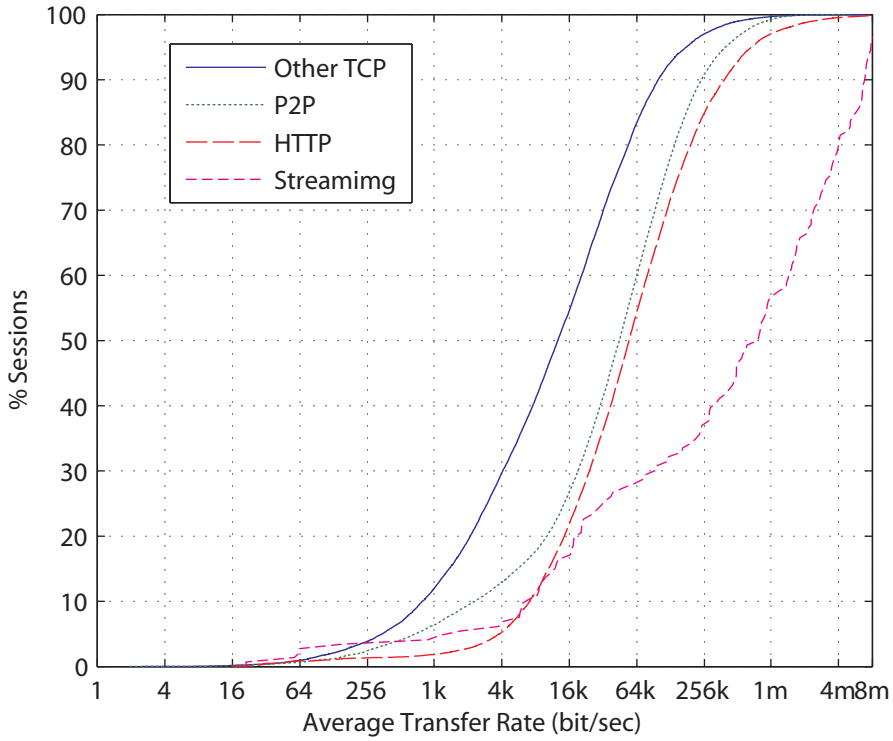


Figure 9.6: CDF of session's average transfer rates by applications for the LTE access technology

Table 9.4: Distributions of session's average transfer rates (Kbps) by applications for the LTE access technology

	HTTP	**P2P**	**Streaming**	**Other TCP**
25^{th}	18.9	14.4	32.9	3.0
50^{th}	54.5	44.7	774.1	12.6
75^{th}	148.5	112.1	3391.9	41.2
90^{th}	383.7	240.5	6876.7	99.2
95^{th}	671.7	392.3	7,774.6	177.3
99^{th}	2,369.9	906.3	9,896.3	526.0

median bandwidth of WiFi is greater than that of 3G by about a factor of 5.

On the other hand, LTE can deliver performance almost up to WiFi in most cases. The streaming application class can even perform better using LTE in many percentiles. Although the link capacities of LTE clients are better than those of WiFi clients, the link latencies are not. Consequently, the streaming applications which involve large volume transfers can gain more benefits from a higher bandwidth of LTE. However, WiFi's better latency can still have more impact on the performance for many types of applications which comprise many short and small flows.

In the real scenario, there are many factors which can make the performance results varied. In our scenario, WiFi link characteristics are extracted from traces captured from a lightly-loaded network. When the number of users is increased, we expect to see a significant drop in performance for WiFi. On the other hand, LTE has been developed to scale much better than WiFi. Therefore, it is possible for LTE to outperform WiFi in heavy load. Additionally, the LTE capacities might be improved from those in our scenario when user equipments are configured with more antennas.

From the results, we can still conclude that WiFi is a good candidate to augment 3G services in a small highly-populated area such as a coffee shop or an airport, so operators can save the cost of investing in more expensive 3G infrastructure. On the other hand, we expect the performance of LTE to be comparable to or even better than WiFi, especially when the number users increases. Additionally, LTE provides better mobility management and quality of service control. Therefore, LTE can be a good candidate to replace 3G as well as the current WiFi in the future. Though, we have yet to evaluate the 802.11n standard which has just started to become prevailing. We believe both LTE and 802.11n are potential access technologies that can dominate the mobile communication world in the future.

9.2.1 Limitations

We still have limitations of wireless characteristics and user mobility as mentioned in section 8.4. Also our model cannot capture jitter or quality of service control in the real network. Therefore, we cannot realistically measure the user's perceived quality related to these issues such as the jitter delay, coverage and availability of service.

Additionally, we only estimate the link characteristics of LTE by scaling them from those of 3G based on the specifications, so the performance in a real world can be different from our results.

Chapter 10

Conclusion and Future Work

We extended a resource reservation framework proposed by Al-Fares et al. to analyze the benefits of operator cooperation based on the realistic 3G data traffic. In our scenario, we tried to leverage the difference in burstiness in small timescales to shed the peak data bandwidth usage of one operator onto another even when the average load of all operators are almost equal. We integrated the resource reservation framework into Swing and exploited Swing to generate the traffic based on the real 3G trace data. The results show that even when the capacities of all operators are limited to less than the peak usage, the operator cooperation can help maintain quality of service upto the non-constrained case for most of the user's sessions. As a result, all operators can be benefited from the cooperation by provisioning lesser bandwidth.

We also investigated performance delivered by various kinds of wireless technology networks including 3G, WiFi and LTE for various application classes. We utilized Swing's abilities to generate realistic traffic as well as to tune the topology parameters to reflect different kinds of access technologies. Our results show that WiFi can provide significantly better performance compared to 3G in every application class. On the other hand, we expect the performance of LTE to be comparable to or even better than WiFi, especially when the number of users increases.

There are still many potential and interesting aspects to explore further such as the business model. With operator cooperation, the notion of home operator

becomes less significant. In the ideal case, users can choose to access any network they feel like. Then, the pricing model and business competition will come into a play. It is interesting to see how both operators and users can benefit from this scenario.

Additionally, user behavior has never stopped evolving. In the future, the benefits of operator cooperation may be more significant. Therefore, it is certainly valuable to evaluate the system again when we have more trace data.

Bibliography

[1] 3GPP. Radio link control (RLC) protocol specification (Release 5) TS 25.322. 2002.

[2] P. Abry and D. Veitch. Wavelet analysis of long-range-dependent traffic. *IEEE transactions on information theory*, 44(1):2–15, 1998.

[3] M. Afanasyev, T. Chen, G. Voelker, and A. Snoeren. Analysis of a mixed-use urban wifi network: When metropolitan becomes neapolitan. In *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement conference*, pages 85–98. ACM New York, NY, USA, 2008.

[4] M. Al-Fares, M. Johnsson, P. Johansson, and A. Vahdat. Flexible resource allocation and composition across GSM/3G networks and WLANs. In *Proceedings of the 3rd international workshop on Mobility in the evolving internet architecture*, pages 73–78. ACM New York, NY, USA, 2008.

[5] J. Bannister, P. Mather, and S. Coope. Convergence Technologies for 3G Networks: IP, UMTS, EGPRS and ATM. 2004.

[6] P. Benko and A. Veres. A passive method for estimating end-to-end TCP packet loss. In *IEEE Global Telecommunications Conference, 2002. GLOBECOM'02*, volume 3, 2002.

[7] V. Brik, S. Rayanchu, S. Saha, S. Sen, V. Shrivastava, and S. Banerjee. A measurement study of a commercial-grade urban WiFi mesh. In *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement conference*, pages 111–124. ACM New York, NY, USA, 2008.

[8] Y. Cheng, P. Verkaik, P. Benkö, J. Chiang, A. Snoeren, S. Savage, and G. Voelker. Automating cross-layer diagnosis of enterprise wireless networks. 2007.

[9] Y. Cheng, P. Verkaik, P. Benkö, J. Chiang, A. Snoeren, S. Savage, and G. Voelker. UCSD CSE wireless traces, 2007. http://sysnet.ucsd.edu/wireless/traces/sigcomm2007/.

[10] E. Dahlman, H. Ekstrom, A. Furuskar, Y. Jading, J. Karlsson, M. Lundevall, and S. Parkvall. The 3G long-term evolution-radio interface concepts and performance evaluation. In *IEEE 63rd Vehicular Technology Conference, 2006. VTC 2006-Spring*, volume 1, 2006.

[11] E. Dahlman, S. Parkvall, J. Skold, and P. Beming. *3G evolution: HSPA and LTE for mobile broadband*. Academic Press, 2007.

[12] C. Dovrolis, P. Ramanathan, and D. Moore. Packet-dispersion techniques and a capacity-estimation methodology. *IEEE/ACM Transactions On Networking*, 12(6):963–977, 2004.

[13] N. S. Foundation. Geni: The global environment for network innovations, 2009. http://www.geni.net/.

[14] T. Hoßfeld, A. Binzenhöfer, M. Fiedler, and K. Tutschku. Measurement and analysis of Skype VoIP traffic in 3G UMTS systems. *Proc. of IPS-MoMe*, pages 52–61, 2006.

[15] R. Jennen et al. System evaluation results (d27-h.5), December 2007. http://www.ambient-networks.org/deliverables.html.

[16] K. Johansson, A. Furuskar, and C. Bergljung. A Methodology for Estimating cost and Performance of Heterogeneous Wireless Access Networks. In *Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on*, 2007.

[17] M. Johnsson et al. An system description (d18-a.4), February 2008. http://www.ambient-networks.org/deliverables.html.

[18] H. Kaaranen, S. Naghian, L. Laitinen, A. Ahtiainen, and V. Niemi. UMTS Networks: Architecture, Mobility and Services. 2005.

[19] M. Kohlwes, J. Riihijarvi, P. Mahonen, F. GmbH, and G. Dusseldorf. Measurements of TCP performance over UMTS networks in near-ideal conditions. In *2005 IEEE 61st Vehicular Technology Conference, 2005. VTC 2005-Spring*, volume 4, 2005.

[20] A. N. Partners. Ambient networks. http://www.ambient-networks.org.

[21] P. Pöyhönen, J. Markendahl, and O. Strandberg. Analysis of benefits of operator cooperation using end-user and operator performance metrics. In *The International Telecommunications Society 17th Biennial Conference, Montreal*, 2008.

[22] P. Svoboda, F. Ricciato, W. Keim, and M. Rupp. Measured WEB Performance in GPRS, EDGE, UMTS and HSDPA with and without Caching. In *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, 2007. WoWMoM 2007*, pages 1–6, 2007.

[23] W. Tan, F. Lam, and W. Lau. An Empirical Study on 3G Network Capacity and Performance. *Proc. EEE INFOCOM*, pages 1514–1522, 2007.

[24] A. V. Thomas Anderson. Geni distributed services, November 2006. http://groups.geni.net/geni/attachment/wiki/OldGPGDesignDocuments/GDD-06-24.pdf.

[25] A. Vahdat, K. Yocum, K. Walsh, P. Mahadevan, D. Kostić, J. Chase, and D. Becker. Scalability and accuracy in a large-scale network emulator. *ACM SIGOPS Operating Systems Review*, 36:271–284, 2002.

[26] K. Vishwanath and A. Vahdat. Swing: Realistic and Responsive Network Traffic Generation. In *Proceedings of IEEE/ACM Transactions on Networking*, 2009.

[27] D. Zhou et al. Validated composition and compensation architecture (d26-g.2), December 2007. http://www.ambient-networks.org/deliverables.html.