

TERENA Technical Report

User Level Network Performance Monitoring Program

Hans Blom *J.Blom@phys.uu.nl* and **Cees de Laat** *C.T.A.M.deLaat@phys.uu.nl*

**Workgroup Computational Physics
Utrecht University
The Netherlands**

September 2000

Summary

At Utrecht University, a network monitoring package was developed, named RTPL (Remote Throughput Ping Load), which is capable of visualising Quality of Service aspects of a network from an end-user's perspective. This is achieved with periodic measurement (once an hour) of distributed throughput and round-trip times and by behavioural observations made from nodes at the edge of the network. The results of these tests are presented in a Web-based format.

This report describes the tool that was further developed in a project supported by TERENA. To illustrate the capabilities of the tool, the TERENA project also included performance measurements on the TEN-155 European research networking backbone. In order to do these measurements, the project had access to several workstations owned by NRNs and universities strategically situated along, or in the vicinity of the central ring: London - Amsterdam - Frankfurt - Paris - London. Due to redundancy in the connections between the workstations, the performance measurements were split into two batches to allow more workstation combinations to be used to test each connection link.

The results of these tests are presented in an extensive appendix to the report which includes graphs of the throughput and round-trip measurements. The main conclusion drawn from these measurements is that during the measurement period the largest performance decreases occurred on workdays at around 15:00, although no dramatic performance decreases were found on any of the tested connections.

The software developed by the project is available from:

<http://www.phys.uu.nl/~wwwfi/rtpl/>

For further information please contact:

TERENA Secretariat	Tel:	+31 20 530 4488
Singel 468 D	Fax:	+31 20 530 4499
1017 AW Amsterdam	E-mail:	secretariat@terena.nl
The Netherlands	WWW:	http://www.terena.nl/

© TERENA 2000, all rights reserved

Parts of this report may be freely copied, unaltered, provided that the original source is acknowledged and the copyright preserved.

Contents

1	Introduction	4
2	Description of RTPL	5
2.1	Overview	5
2.1.1	Measurement procedure	6
2.1.2	Presentation results and data files	7
2.1.3	Test modes	8
2.1.4	Help scripts	8
2.1.5	Platforms	8
3	Test Setup	9
3.1	Sites	9
3.2	Batches	11
3.3	Test parameters	11
3.4	Conclusions	12
4	Bibliography	14
	Appendix A: Results of Measurements	15
A.1	Throughput	15
A.1.1	Batch 1	16
A.1.2	Batch 2	25
A.2	Round-trip	32
A.2.1	Minimum round-trip	32
A.2.2	Typical minimum round-trip plots	34

CONTENTS

1 Introduction

In the second half of 1999 a discrepancy was observed between the perceived performance of the TEN-155 network, as monitored by DANTE, and that observed by some of the network's users. To help identify possible causes, the University of Utrecht developed a distributed measurement tool which periodically (once per hour) observes throughput and delay behaviour from nodes at the edges of the network.

Currently, most network monitoring tools provide information on the volume of data transported by core networks. Other tools exist to measure round-trip times and packet losses; however, most of these tools do not visualise what an end-user is really interested in - the goodput. Goodput is defined as the actual amount of useful data transferred to and from the user application. Packet losses and retransmissions are of no interest for the goodput, although they influence the goodput, because the round-trip time or, more importantly, task completion time in applications may increase. Reports from core networks may show high-volume data transfers but this information is useless to the end-user if it is not known what part of it is retransmission.

The network monitoring package that was developed at Utrecht University, named RTPL, is capable of visualising end-user Quality of Service (QoS) aspects of the (research) network. Measurements were performed for several applications, on the European research network backbone, on a distributed computer cluster system, DAS, scattered over five universities in the Netherlands and on the user sites of the collaborative experiment Dynacore (RE 4005). In the Dynacore project the goal was to see if the current backbone could support the network requirements of a collaborative environment. In the Distributed ASCI Supercomputer project the goal was to compare the QoS obtained from an ATM based MBS service with the "normal" best-effort Internet service. In Dynacore we found that the available network resources were insufficient to exploit any form of collaboration, but the situation has improved a lot recently. In the DAS project we proved that the use of a QoS service is very beneficial to the application of distributed computing.

This report describes the tool that was further developed and documented in a project supported by TERENA. To illustrate the capabilities of the tool, the TERENA project also encompassed performance measurements on the TEN-155 European research networking backbone. To be able to do this we had access to several workstations, owned by National Research Networks (NRNs) or universities, strategically situated along, or in the vicinity of, the central ring of the TEN-155 network: London - Amsterdam - Frankfurt - Paris - London. However, in some cases the distance to the TEN-155 network was relatively large.

The organisation of this report is as follows. Chapter 2 gives a general description of the RTPL package. The setup for the measurements on the TEN-155 network is described in chapter 3 and some conclusions are given there. Appendix A provides detailed results of the measurements that were carried out.

2 Description of RTPL

2.1 Overview

As explained in the introduction, the RTPL (Remote Throughput Ping Load) package is used to perform periodic performance measurement tests between a set of workstations in order to view network performance from a user's perspective. The performance measurements consist of round-trip and throughput measurements between a number of hosts. By default all pairs in the hosts' set are used, but it is also possible to select the host pairs for the tests. The machine load of the participating workstations is also measured so that performance loss can be related to heavy machine load.

Because we are interested in network performance from a user's perspective and not in the maximum possible capacity of a network, the measurement parameters are configured to default values and the duration of each test is limited since long-term statistics will blur short-time fluctuations.

The tests are performed by scripts written in the scripting language Perl, which is available for many operating systems. The scripts are used to parse, sample and organise the results. The real performance measurements are executed by commonly used Internet tools invoked by the Perl scripts.

The presentation of the results is Web-based and dynamic: the network performance data is stored in ZIP compressed plain text files, which are accessible from a Web server. There are various files so that a user can be offered different views of the data, including several time-based averages. The data files are read into the Web browser by a Java Applet. The HTML scripting language JavaScript is used to display the data in various tables. The Applet can also be used to present the data in graph form.

The RTPL package is only available for Unix platforms. The main reason for this is that the so-called remote shell mechanism, used to obtain the results from the participating hosts, although implemented on Windows NT/9X, is not standard and not generally available. There is also a standard form of the package, which only executes round-trip tests from the local host to a specified set of remote hosts. This basic package is available for both Unix and Windows NT/9X. The basic version is not discussed further in this report.

The round-trip tests via the ping service are also used by the HEP Internet Monitoring Project (*PingER*) [1]. This is an initiative to measure the performance of the Internet used by the High Energy Physics Research community. However, we believe that, especially for networks with more capacity, throughput measurements are also required to obtain a better understanding of network performance.

2.1.1 Measurement procedure

In this section a more detailed description of the RTPL package is given. The measurements are performed by a, so-called, control host. This workstation starts the performance measurements at each of the participating workstations with a, so-called, remote shell command¹. The results of the measurements are also send back via the remote shell command. This process is illustrated in figure 2.1.

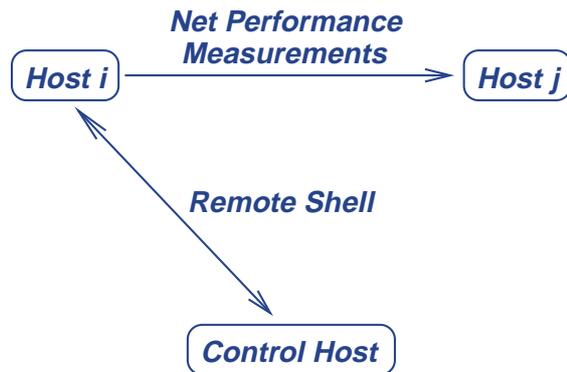


Figure 2.1: The control host starts the measurements at all hosts i by means of remote shells.

The host i performs the measurements to the hosts j . The results are send back by host i to the control host.

At each host i of the set of hosts described above, the following performance measurements are executed:

Throughput: The formal definition from RFC 1224 [2] is: "The maximum rate at which none of the offered frames are dropped by the device". It is a way to quantify the traffic flow that can be handled by a network connection. As default, throughput is measured for the connections from the current host i to all other hosts, but it is also possible to skip connections. Throughput is measured with the public domain command `netperf`.

Round-trip or Ping time: This Internet application is described in RFC 2151 [3] as:

"Ping, reportedly an acronym for the Packet InterNetwork Groper, is one of the most widely available tools bundled with TCP/IP software packages. Ping uses a series of Internet Control Message Protocol (ICMP) [4] Echo messages to determine if a remote host is active or inactive, and to determine the round-trip delay in communicating with it."

The round-trip time quantifies the response offered by a network connection. It will be measured, before the throughput, across the same connections as the throughput. The round-trip time is measured with the system command `ping`.

¹ A remote shell makes it possible to control a remote process in almost the same way as if it was running on the local host. To be able to do this it connects the Standard I/O streams from the remote host to the local host using sockets. The relevant signals are also transported to the remote host. Security aspects are solved in various ways, depending on the remote shell command used. However, in all cases there is no need to specify a username or password to get access to the remote host: this would prevent a non-interactive run of the command at the remote host. A modern implementation of the remote shell is the *Secure Shell* (`ssh` command) which offers an encrypted connection to the remote host.

Load: This is expressed here as the number of fully active processes running on a host. The load of a host is not a quantity related to network load, but it may help to explain unexpected performance decreases. The load is measured at the current host *i*, using the system command `uptime`.

The sampling of the results at the control host and the measurements at all hosts, participating in the tests, are performed by scripts in the scripting language *Perl*.

The Perl script at the control host collects the results of the measurements for each host *i* and stores the results in ZIP compressed data files. ZIP compression is used to reduce disk space and download time. (See section 2.1.2 for a description of the data files.)

The tests are periodically started with the `crontab`² command. A sample `crontab` input file, which is specific for a host's set, will be generated at the installation of that set. If the previous test is not ended when a new one is started by `crontab`, the new test terminates. However, to prevent deadlocks, after a specified number of terminations, the new version kills the old one and continues. Timeouts are used in the test scripts to prevent deadlocks.

2.1.2 Presentation results and data files

The presentation of the results is Web-based: a Java Applet³ is used to load the data from the files into the memory of the Web browser of the user analysing the results. The functionality to read (ZIP compressed) data files from a Web browser is implemented in Java. Therefore, the Applet is required for access to the data.

The HTML scripting language JavaScript is used to dynamically present to the user various HTML tables of the data. That is: the user selects a view of the data and the HTML code is generated on-demand by JavaScript. From the JavaScript code, direct calls to Applet methods are applied to obtain the required data for the HTML tables to display. The direct call of an Applet method is a standard JavaScript feature that is (and should be) supported by most Web browsers. The Applet is also used to make various plots of the data. The plots are displayed in a new Applet window.

The following data files are available for viewing via the Web:

- The data of the last 7 days.
- The data measured in a week. For each week a data file is available. The data is stored for maximum of one year.
- The week mean values from the last year.
- The day mean values from the last year.
- The mean values, calculated at the periodic measurement times (each hour), for the days of the week, averaged during a quarter. The data is stored during the year.
- The mean values, calculated at the periodic measurement times, for the workdays of the week, averaged during a month. The data is stored during the year.

² With the `crontab` command, a user can start programs periodically at specific dates and or times using the `cron` daemon (server). It is available on most Unix systems. A public domain version of the source also exists.

³ A Java Applet is an architecture independent application, written in the Java programming language, which runs in the so-called Java virtual machine of a Web browser.

The user can download the presented data file to the local host. He/she can also download a special version of the RTPL package which can be used to display the downloaded data files off-line. This version only contains the Web files from the package.

2.1.3 Test modes

The network performance tests can be executed in the modes explained below. These modes may be used in all possible combinations. Each selection mode leads to a different presentation of the results. The user can select the mode he/she wishes to use when he/she configures the set of hosts.

- Do the tests across a QoS route between the hosts and via the "usual" Internet route. Alternatively, only one route between each host pair can be selected.
- Do the tests between each host pair uni-directionally or bi-directionally.

2.1.4 Help scripts

At the control host, Perl scripts are available to manipulate the measured data in various ways. These scripts can, among others, be used to:

- Summarise the available data.
- Calculate various histograms from the measured data.
- Calculate various averages within the data set.
- List characteristic events specified by regular expressions in the performance data.

Net load

The Perl help script `net_load` can be used to determine the various routes between sites participating in a test. The network load on these routes, induced by the traffic generated at the participating sites, can also be calculated. To be able to determine the routes between the sites, a simple generic routing scheme is used. However, it is also possible to define (parts of) the routes manually.

In general this script can be used to estimate the effect on network load induced by this package for a particular site setup. The `net_load` script does not contain architecture dependent code; therefore, it should run on every architecture that is supported by Perl.

2.1.5 Platforms

The RTPL package is, in principle, available for all Unix platforms. It has been used with BSD/OS (Free BSD), OSF1 (Digital Unix), IRIX64, HP-UX, Linux (RedHat, Debian) and SunOS.

3 Test Setup

In this chapter, the setup for the network performance measurements on the European research networking backbone is described. The background of these experiments was explained in the introduction to this report.

3.1 Sites

A description of the sites participating in the European research network monitoring experiment is given in table 3.1. The names, displayed in the column with the header "Title", correspond with the host titles used in the HTML tables from the network performance monitor.

The topology map in figure 3.1 shows the part of the TEN-155 network that connects the NRNs and sites involved in the experiment. Note that this map does not give a correct geographical representation of the locations. For more information, also see the topology map⁴ of the TEN-155⁵ network provided by DANTE⁶

From the topology map in figure 3.1 it follows that multiple routes across the TEN-155 network exist for traffic between some sites. The routes that were valid for the test period are listed in table 3.2. Due to changing conditions, these routes may no longer be valid.

Please note that the throughput and round-trip measurements were not executed for all pairs of sites: tests were only performed on connections between sites that were selected such that:

- At each TEN-155 PoP traffic was following all possible routes.
- At each direct TEN-155 connection the total time spent on throughput measurements was limited to 40 or 60 seconds per hour, depending on the connection.

4 <http://www.dante.net/ten-155/ten155net.gif>

5 <http://www.dante.net/ten-155.html>

6 <http://www.dante.net/>

Title	Institute	Location	Hostname (NRN)	System (Bandwidth in Mbit/s)
EUCS	Computing Services, The University of Edinburgh	Edinburgh, United Kingdom	qasbah.ucs.ed.ac.uk (JANET)	Red Hat Linux 6.0 (10)
MCC	Manchester Computing, University of Manchester	Manchester, United Kingdom	nessie.mcc.ac.uk (JANET)	FreeBSD 3.4 (10)
ULB	Université Libre de Bruxelles	Brussels, Belgium	sun7.ihe.ac.be (BELNET)	Sun Solaris 2.6 (10)
SARA	Academic Computing Services Amsterdam	Amsterdam, The Netherlands	192.87.106.120 (SURFnet)	Red Hat Linux 5.1 (100)
UU-36	Institute of Computational Physics, Utrecht University	Utrecht, The Netherlands	hst3736.phys.uu.nl (SURFnet)	Sun Solaris 7 (100)
ZAM	Central Institute for Applied Mathematics, Forschungszentrum Jülich	Jülich, Germany	zam472-b.zam.kfa-juelich.de (DFN)	Sun Solaris 7 (100)
CERN	CERN	Geneva, Switzerland	sunstats.cern.ch (CERN via SWITCH/TEN-155)	Sun Solaris 2.6 (100)
CIC	Computing and Information Centre, Czech Technical University in Prague	Prague, Czech Republic	nms.cvut.cz (CESNET)	Sun Solaris 2.5 (100)

Table 3.1: Description of the sites participating in the European research network monitoring experiment.

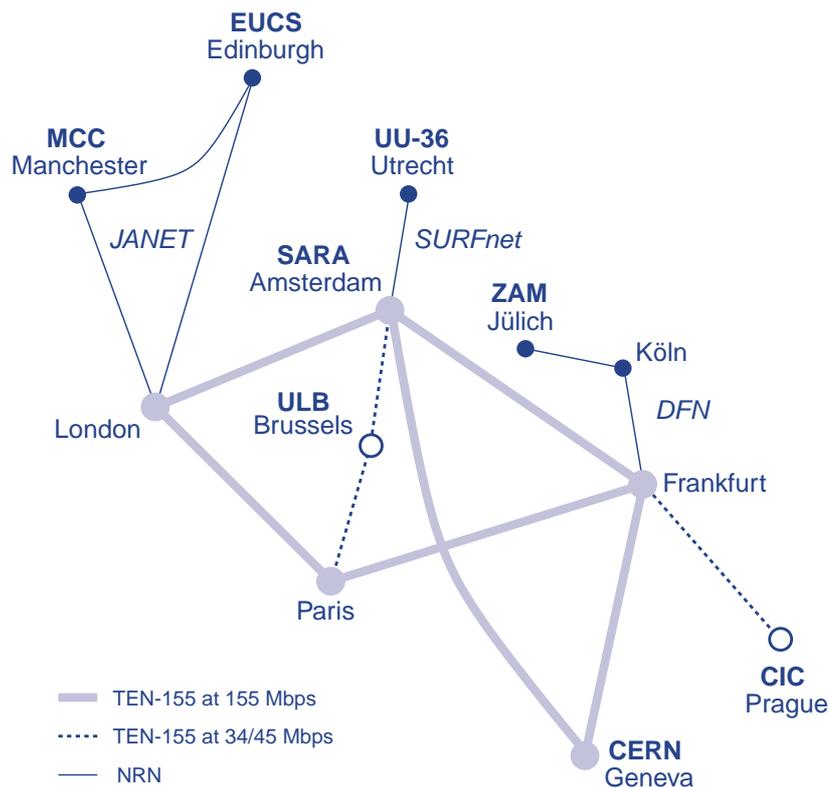


Figure 3.1: Topology map showing the part of the TEN-155 network that connects the NRNs and sites participating in the European research network monitoring experiment.

Connection	Route
London ↔ Frankfurt	London - Paris - Frankfurt
London ↔ Geneva	London - Amsterdam - Geneva
Amsterdam ↔ Geneva	Amsterdam - Geneva
London ↔ Brussels	London - Paris - Brussels
Frankfurt ↔ Brussels	Frankfurt - Paris - Brussels
Geneva ↔ Brussels	Geneva - Amsterdam - Brussels

Table 3.2: TEN-155 routes used when there are multiple possibilities available to make a connection. The reverse path is used for the connection in the other direction.

This redundancy of connections led to the introduction of two test batches, which will be explained in more detail in the next section.

3.2 Batches

Due to the redundancy in the connections between the sites, the performance measurements were split into two *batches* such that more site combinations could be used to test a TEN-155 and/or NRN connection link. The measurement periods for these two batches were:

- **Batch 1** 6 February 2000 - 2 April 2000 (weeks 06 - 13).
- **Batch 2** 4 April 2000 - 7 May 2000 (weeks 14 - 19).

A combination of the participating sites shown in table 3.1 with the topology map in figure 3.1 led to a selection of performance measurements between sites, fulfilling the criteria in section 3.1. The selected tests are presented in table 3.3 in the following way: for each TEN-155 link all site connections are listed where performance measurements were taken using this link. The ULB site became unavailable shortly after batch 2 was started. Therefore, the TEN-155 links to Brussels are excluded from table 3.3 for batch 2.

The most important differences with batch 1 were:

- MCC was only used to monitor the JANET connection with EUCS. The main reason was that the connections with this host showed less performance decreases than with MCC.
- UU-36 was for the largest part replaced by SARA and was only used to monitor the SURFnet connection. SARA is more directly linked to the TEN-155 network, hence the substitution.

3.3 Test parameters

The performance tests were executed with the application parameters given in table 3.4. These parameters were used for both batches, with the exception that for batch 2 larger socket sizes, and related to that, a larger TCP sliding window size, were used to study the long-term influences of these adjustments. Manual tests showed that adjustment of the measurement times did not have much influence.

Batch 1

TEN-155 Link	Site Connections
London - Amsterdam	EUCS ↔ UU-36, EUCS ↔ CERN, MCC ↔ UU-36
London - Paris	EUCS ↔ ULB, EUCS ↔ ZAM, MCC ↔ ULB
Amsterdam - Brussels	UU-36 ↔ ULB, CERN ↔ ULB
Paris - Brussels	EUCS ↔ ULB, MCC ↔ ULB, ZAM ↔ ULB
Amsterdam - Frankfurt	UU-36 ↔ ZAM, UU-36 ↔ CIC
Paris - Frankfurt	EUCS ↔ ZAM, ULB ↔ ZAM
Amsterdam - Geneva	EUCS ↔ CERN, UU-36 ↔ CERN, ULB ↔ CERN
Frankfurt - Geneva	ZAM ↔ CERN, CIC ↔ CERN
Frankfurt - Prague	UU-36 ↔ CIC, ZAM ↔ CIC, CERN ↔ CIC

Batch 2

TEN-155 Link	Site Connections
London - Amsterdam	EUCS ↔ SARA, EUCS ↔ CERN
London - Paris	EUCS ↔ ULB, EUCS ↔ ZAM
Amsterdam - Frankfurt	SARA ↔ ZAM, SARA ↔ CIC
Paris - Frankfurt	EUCS ↔ ZAM
Amsterdam - Geneva	EUCS ↔ CERN, SARA ↔ CERN
Frankfurt - Geneva	ZAM ↔ CERN, CIC ↔ CERN
Frankfurt - Prague	SARA ↔ CIC, ZAM ↔ CIC, CERN ↔ CIC

Table 3.3: List of the site connections that used a specific TEN-155 link in both batches. ULB became unavailable in batch 2. Therefore, the Brussels links are excluded from the batch 2 table.

3.4 Conclusions

The detailed results of the two batches of performance measurements are presented in appendix A. In conclusion:

- No severe performance decreases concerning the throughput values and round-trip values were detected during the measurement period.
- On workdays most congestion took place during the afternoon and early evening (11:00 - 18:00). For sites with a bandwidth of 100 Mbit/s the decrease factor was about 50%. The sites with a bandwidth of 10 Mbit/s showed less severe dips in performance.
- The (minimum) round-trip times were influenced little by the throughput performance decreases. The most generic cause of longer round-trip values was growing queueing times. Of course, the influence of queueing times is relatively small because of the long distances of European connections.
- Occasionally the minimum round-trip values increased strongly. This was probably due to rerouting on TEN-155 or NRN networks.

- The site topology used (figure 3.1), which was a compromise between available workstations and the goal to monitor the central TEN-155 ring, was useful to monitor an optimal number of connections, but was not very useful for detecting causes of congestion. To do this, a setup with hosts in the neighbourhood of the various provider PoPs would be more suitable. Using provider workstations at the PoPs would be optimal, but "normal" workstations are also required to keep the end-user's perspective. The PoP workstations should only be used as "control sites" for the provider connections, while the "normal workstations" should be used more intensively. Therefore, it makes no sense to measure the full matrix, but some redundancy is required to be able to analyse the sources of congestion.
- In our setup (figure 3.1) some sites were too far away from the TEN-155 PoPs to separate NRN congestion from the congestion on the European backbone. This was especially true for the sites in the United Kingdom and Germany.

Round-trip

Description	Value
Blocksize	64 bytes
Total # measurements	40
The first # values not used	2

Throughput (measured with netperf)

Description	Value	
	Batch 1	Batch 2
Test duration	10 seconds	10 seconds
Test type to perform	TCP_STREAM	TCP_STREAM
Send size	32768 bytes	65536 bytes
Local socket send/receive buffer sizes	32768 bytes	65536 bytes
Remote socket send/receive buffer sizes	32768 bytes	65536 bytes

Table 3.4: Measurement parameters used in the performance tests.

4 Bibliography

- [1] R. L. A. Cottrel, C. A. Logg, and D. E. Martin, "What is the Internet doing? Performance and reliability monitoring for the HEP Community", *Computer Physics Communications*, vol. 110, pp. 142-148, May 1998.
- [2] S. Bradner, "Benchmarking terminology for network interconnection devices", RFC 1242, Internet Engineering Task Force, July 1991.
URL: <http://www.ietf.org/rfc/rfc1224.txt>.
- [3] G. Kessler and S. Shepard, "A primer on Internet and TCP/IP tools and utilities", RFC 2151, Internet Engineering Task Force, June 1997.
URL: <http://www.ietf.org/rfc/rfc2151.txt>.
- [4] J. Postel, "Internet Control Message Protocol", RFC 792, Internet Engineering Task Force, September 1981. URL: <http://www.ietf.org/rfc/rfc0792.txt>.

Appendix A: Results of Measurements

In this appendix the results of the two batches of performance measurements on the European research network backbone are presented. The issues behind these measurements are given in the introduction while the test setup is further explained in chapter 3. Chapter 3 also makes clear why two batches were used. The throughput measurements and round-trip measurements are presented in this order for the two batches.

A.1 Throughput

In this section the average throughput values, calculated as a function of the hour when the measurements were performed, are presented for both batches. The mean values are determined separately for workdays (Monday - Friday) and for the weekend. For both batches the full measurement period as specified in section 3.2 was used. Please note that average throughput values are also available from the Web presentation of the package, but only on a monthly basis. Therefore, we prefer to recalculate these averages for the available weeks off-line. Please note that these averages can be calculated using the scripts which are included in the package.

The results are presented in the form of graphs of throughput measurements between the sites specified in figure 3.1. Each graph displays the average results of the throughput measurements during workdays or weekends. The graph displays all site connections that use the same TEN-155 link between two adjacent PoPs. The routes in table 3.2 were used to determine which TEN-155 links form the route between two sites. The plots for all TEN-155 links, displayed in figure 3.1, are listed in the next two subsections: each batch is presented in a subsection.

Before presenting the throughput results we would like to emphasise that with the setup used it is not always possible to point to precisely where the causes of performance decreases are situated, because test sites were not available at all PoPs. However, the main goal of this project was to get an end-user perspective of the European backbone network performance. Further analysis and measurements should bring more clarity.

Connection	Figure	Bandwidth in Mbit/s
London ↔ Amsterdam	A.1	10
London ↔ Paris	A.2	10
Amsterdam ↔ Brussels	A.3	10
Paris ↔ Brussels	A.4	10
Amsterdam ↔ Frankfurt	A.5	100
Paris ↔ Frankfurt	A.6	10, 100
Amsterdam ↔ Geneva	A.7	10, 100
Frankfurt ↔ Geneva	A.8	100

Table A.1: List of figures for batch 1 that show the throughput results of all sites which are connected by the corresponding direct TEN-155 links between two adjacent PoPs. It also specifies the bandwidth of the sites connections.

A.1.1 Batch 1

Table A.1 contains a list of figures for batch 1 that show the throughput results for all sites that are connected by the corresponding direct TEN-155 links (see table 3.3). Due to a better throughput performance the results of the tests with EUCS are preferred to the results from MCC. This table also shows whether the connections between the sites have bandwidths of 10 Mbit/s and/or 100 Mbit/s. Additionally, figure A.9 shows the workday throughput values for the SURFnet connection between UU-36 and SARA for batch 1. The graphs presenting the throughput results from batch 1 lead to the following conclusions:

- The main difference between workdays and the weekend was that during the weekend there is no performance loss during the daytime.
- The hosts in the United Kingdom and Brussels were connected to the network with bandwidths of 10 Mbit/s. This made it difficult to compare the throughput measurements to these hosts with the tests between hosts with bandwidths of 100 Mbit/s. However, by focusing on relative quantities like relative performance loss, a comparison of the results was possible.
- The largest performance decreases were on workdays at about 15:00. The deepest dips in performance were on the connections Amsterdam ↔ Frankfurt and Paris ↔ Frankfurt, but not for the connection Prague ↔ Frankfurt. The DFN connection ZAM ↔ Frankfurt was probably not the cause for the performance loss, because the dips in performance for the tests between ZAM and CIC (top plot in figure A.8) were much less. It is also unlikely that the SURFnet connection UU-36 ↔ Amsterdam was the main reason, because the performance decrease in the tests between ZAM and CERN was also much less (top plot in figure A.7).
- The performance decrease at 21:00 for all connections with UU-36 was probably introduced somewhere in the University of Utrecht or SURFnet network: these dips are not found in any other connection pair.
- The performance of the connection CERN ↔ CIC was much lower than the connection UU-36 ↔ CIC (figure A.5). The reason for this is unclear.
- The performance dip in outgoing traffic from UU-36 is a local phenomenon which has been found in more tests with hosts in the Physics Department of Utrecht University. We do not have a clear explanation for this phenomenon.
- During the measurement period no dramatic performance decreases were found on any of the tested connections.

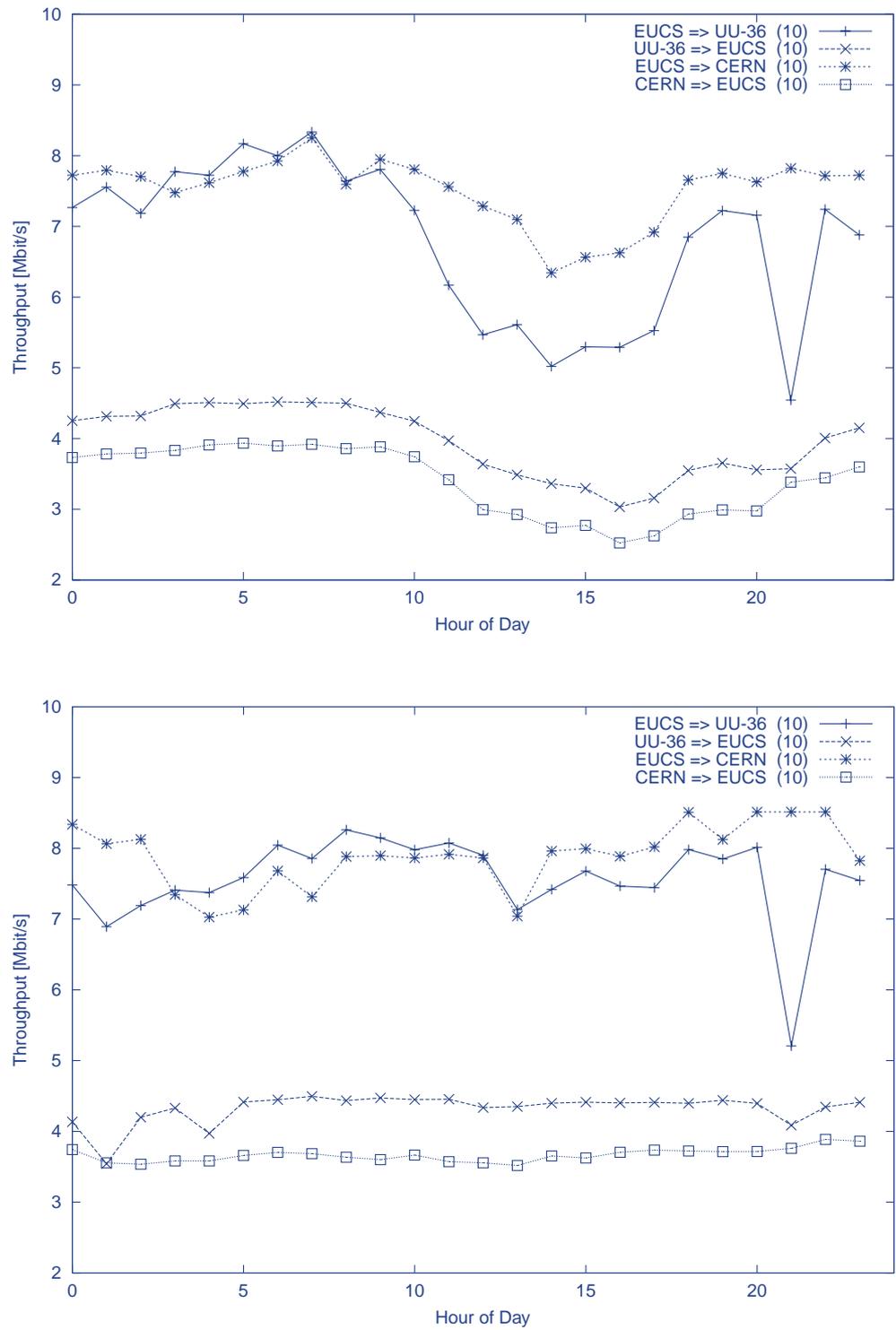


Figure A.1: Average workday (top plot) and weekend (bottom plot) throughput values in batch 1 for the connection London -- Amsterdam. The numbers between brackets are the connection bandwidths in Mbit/s.

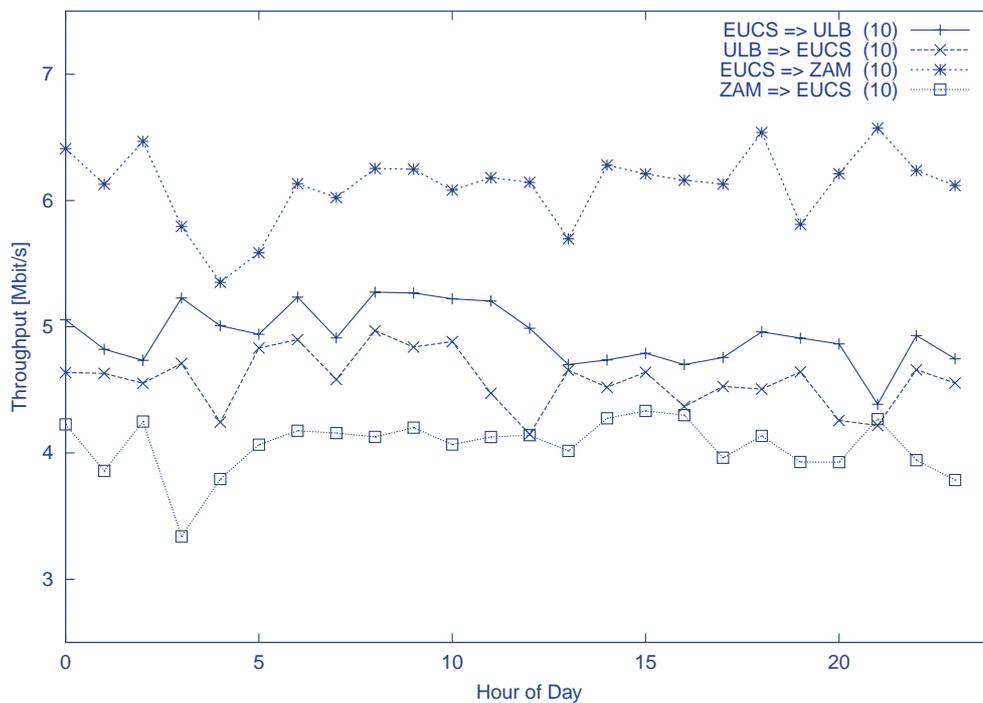
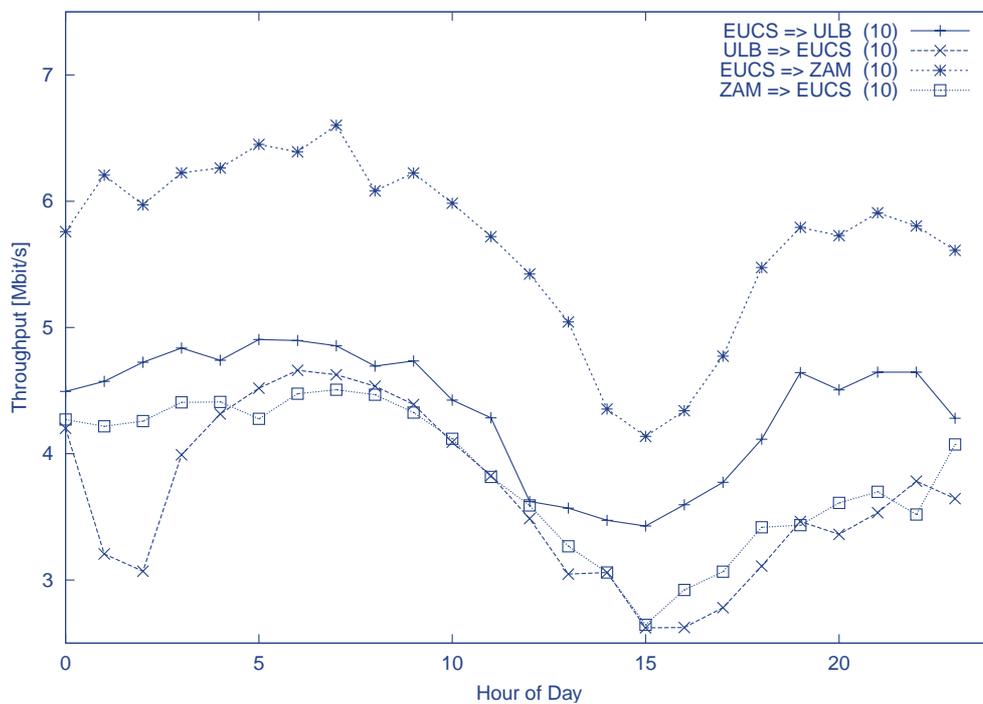


Figure A.2: Average workday (top plot) and weekend (bottom plot) throughput values in batch 1 for the connection London -- Paris. The numbers between brackets are the connection bandwidths in Mbit/s.

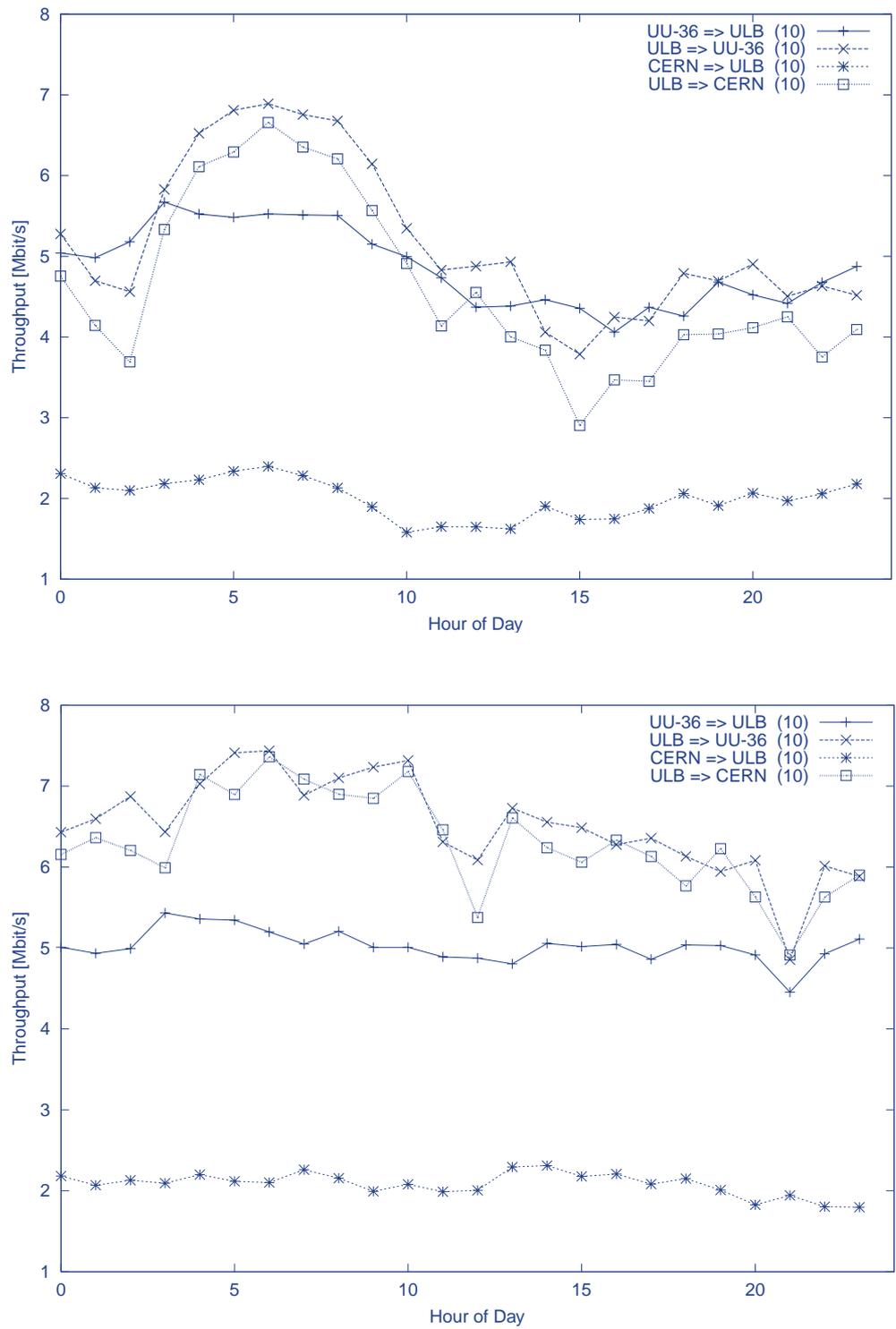


Figure A.3: Average workday (top plot) and weekend (bottom plot) throughput values in batch 1 for the connection Amsterdam — Brussels. The numbers between brackets are the connection bandwidths in Mbit/s.

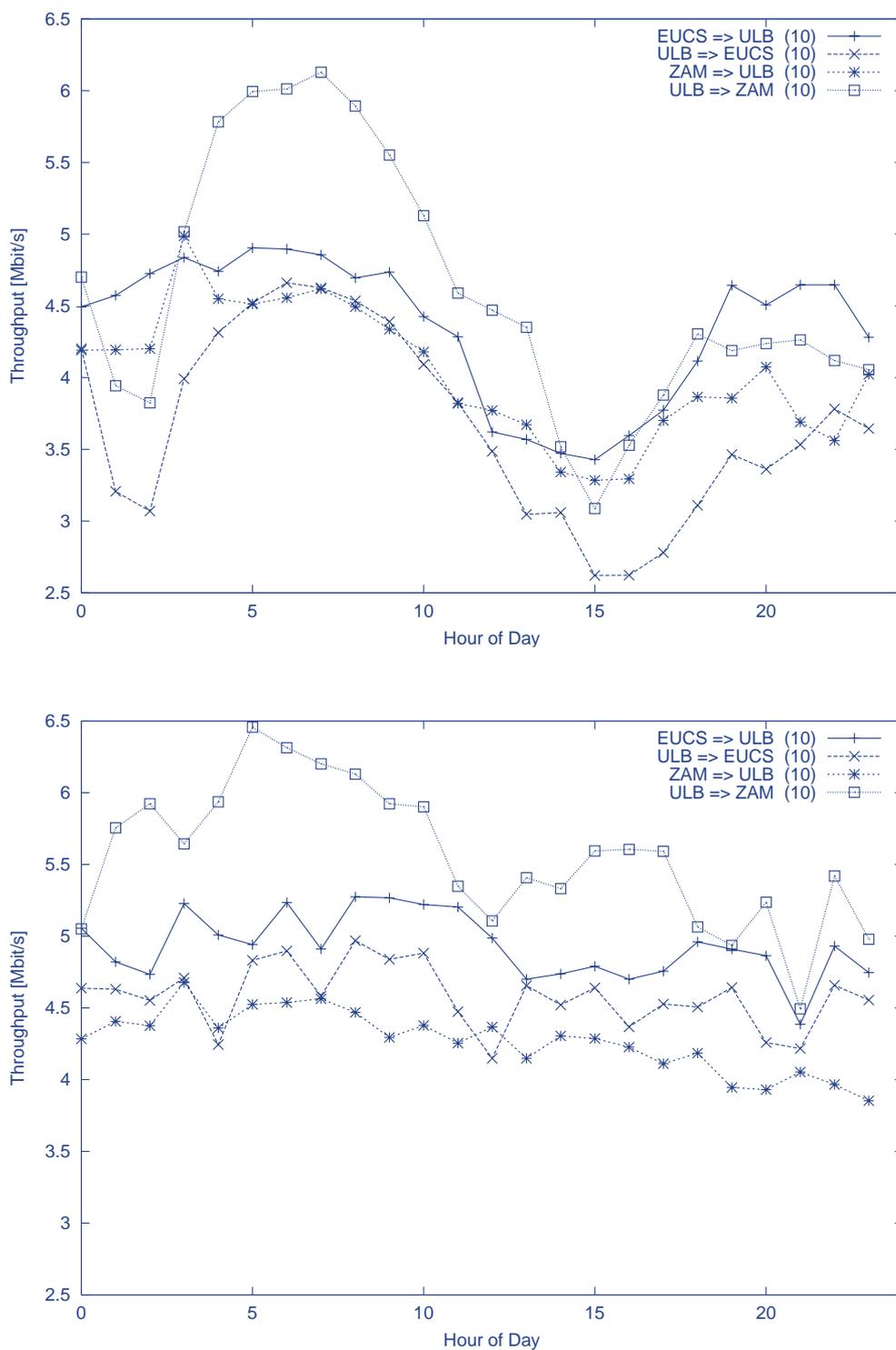


Figure A.4: Average workday (top plot) and weekend (bottom plot) throughput values in batch 1 for the connection Paris — Brussels. The numbers between brackets are the connection bandwidths in Mbit/s.

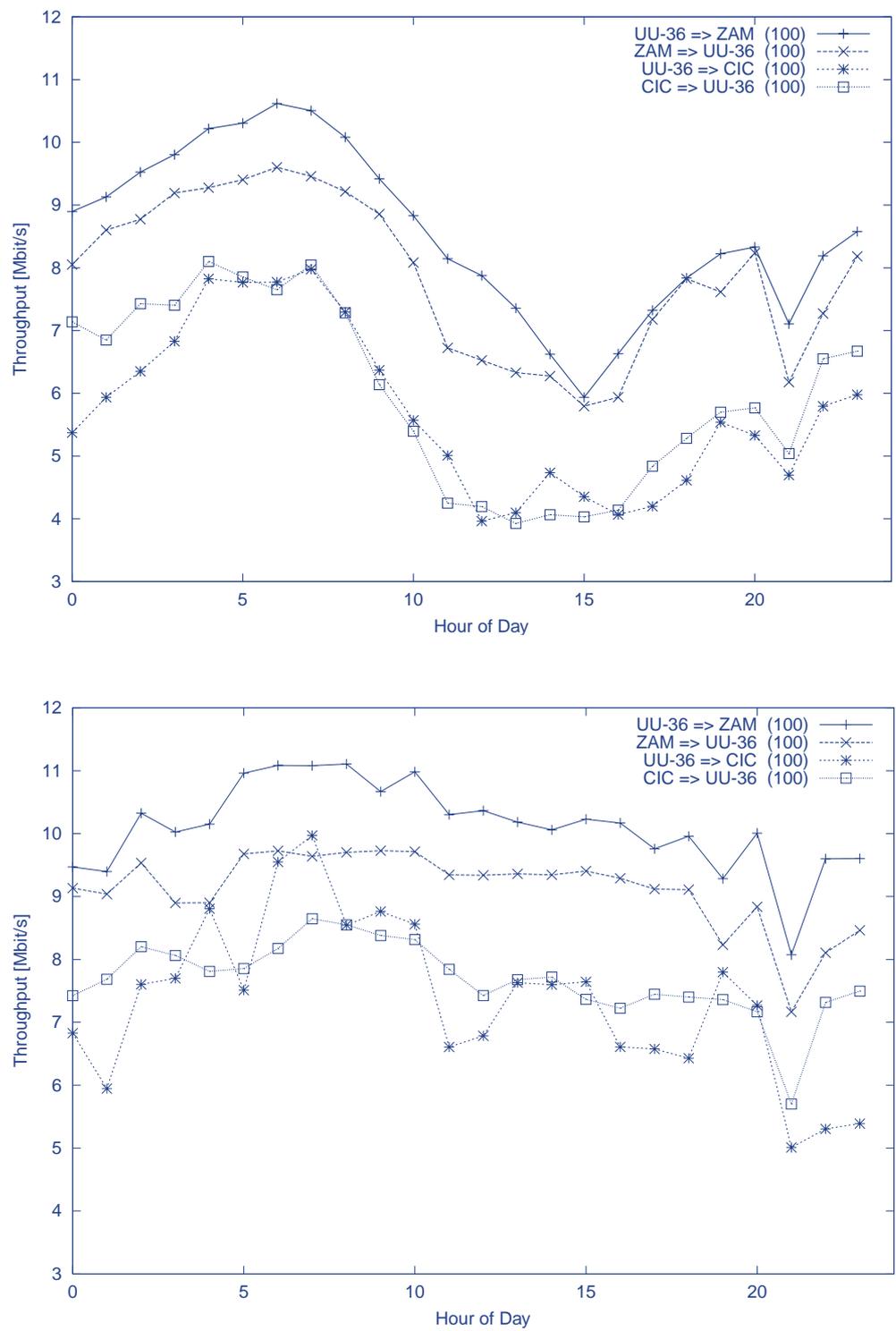


Figure A.5: Average workday (top plot) and weekend (bottom plot) throughput values in batch 1 for the connection Amsterdam — Frankfurt. The numbers between brackets are the connection bandwidths in Mbit/s.

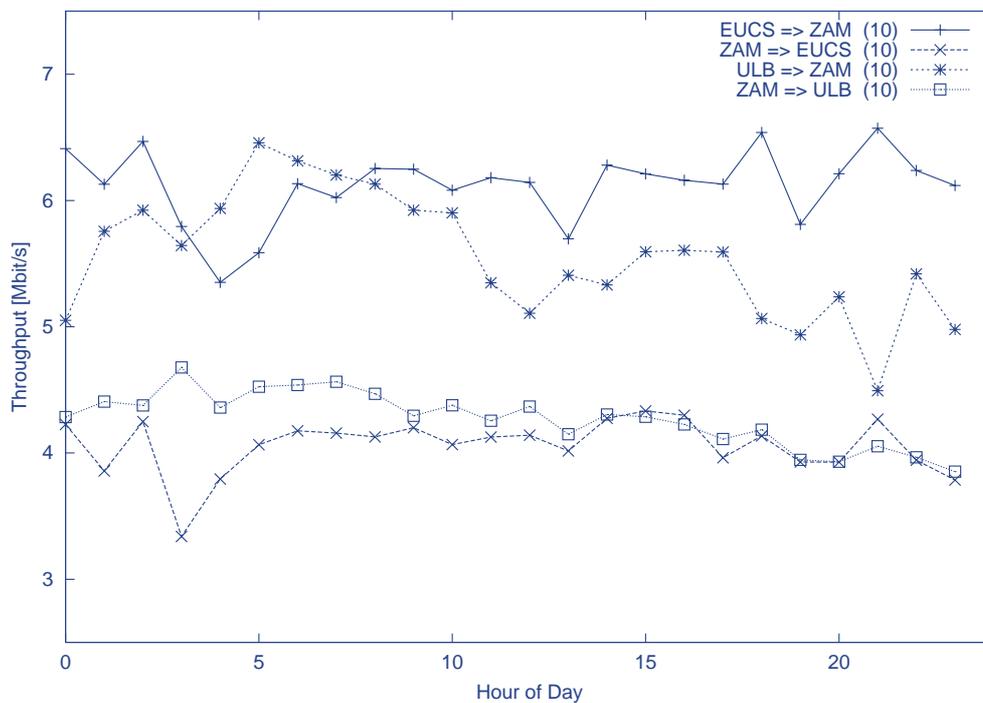
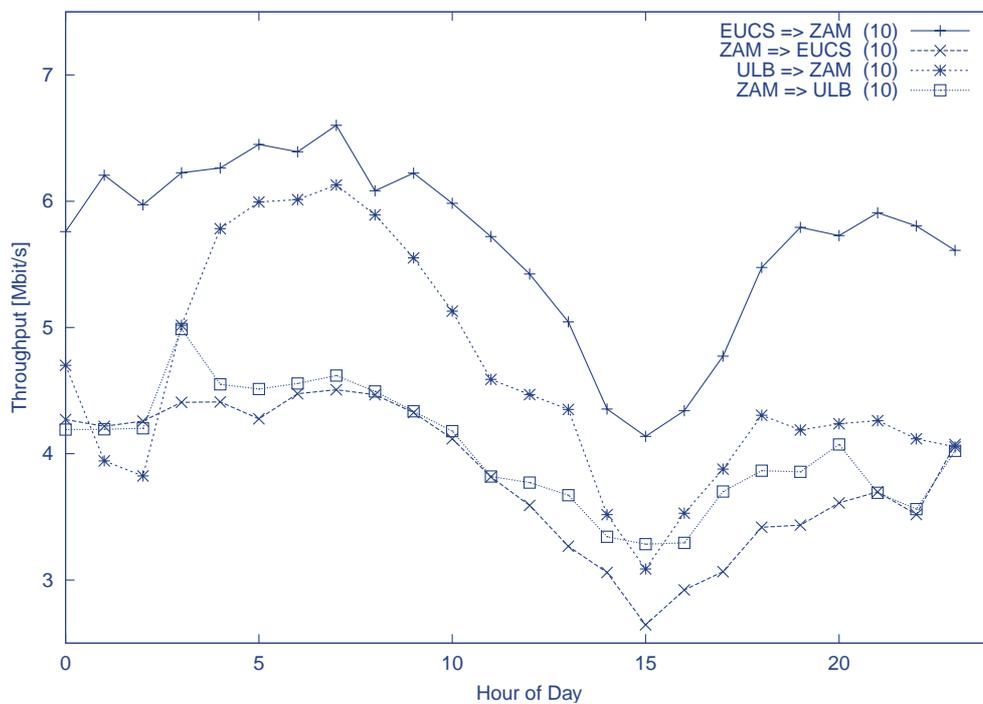


Figure A.6: Average workday (top plot) and weekend (bottom plot) throughput values in batch 1 for the connection Paris — Frankfurt. The numbers between brackets are the connection bandwidths in Mbit/s.

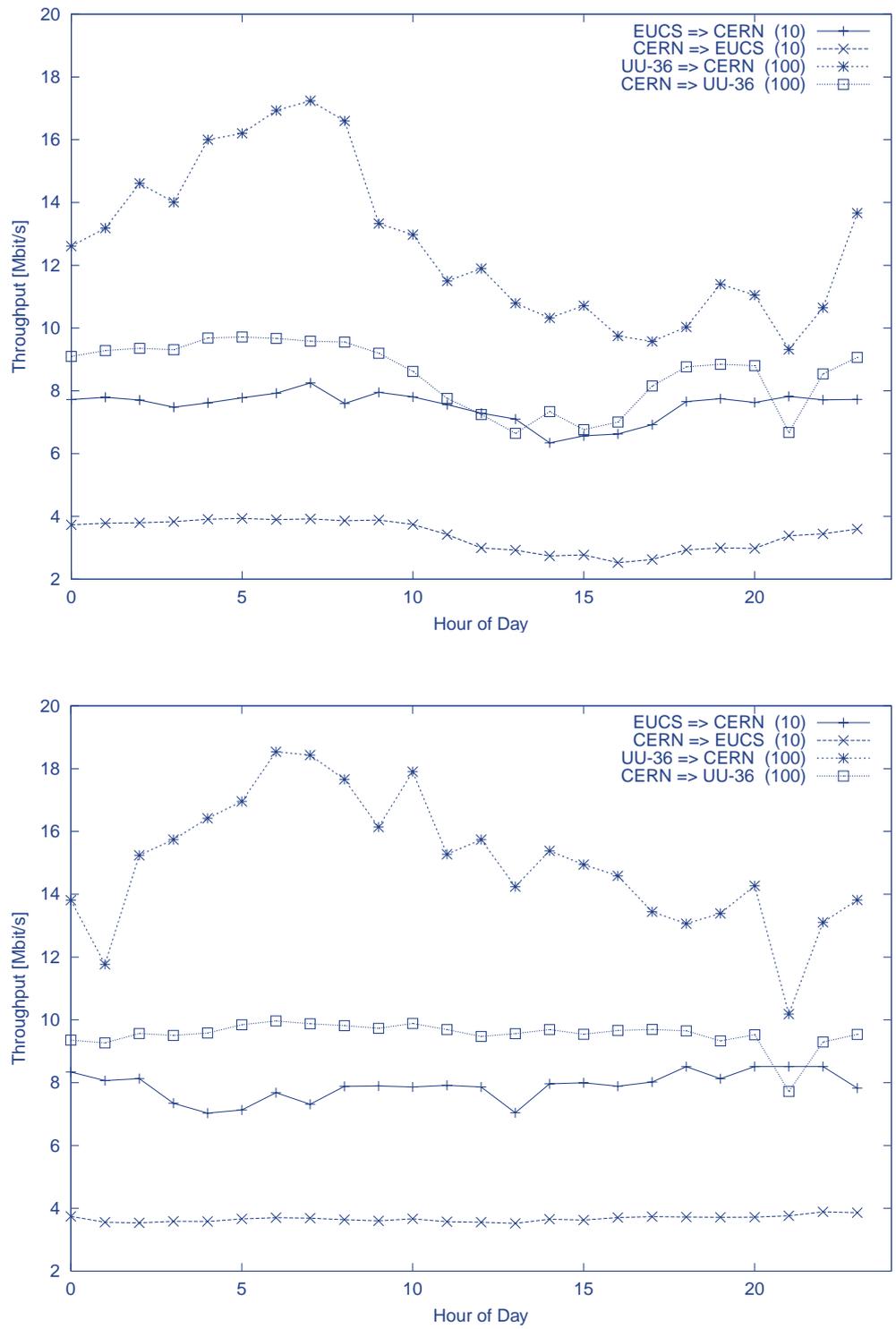


Figure A.7: Average workday (top plot) and weekend (bottom plot) throughput values in batch 1 for the connection Amsterdam — Geneva. The numbers between brackets are the connection bandwidths in Mbit/s.

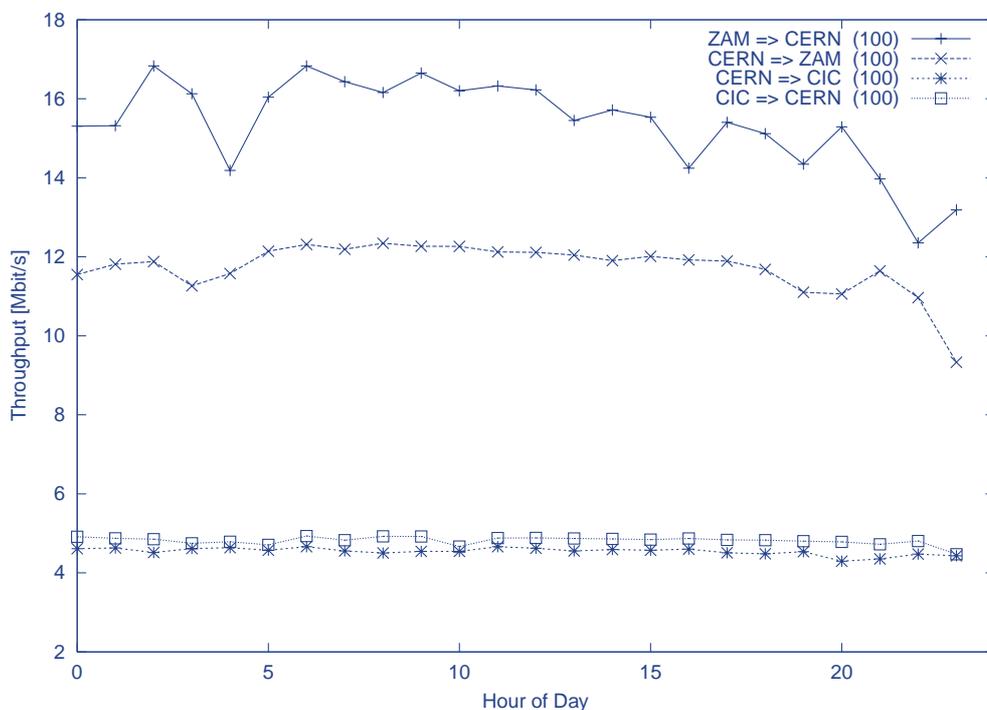
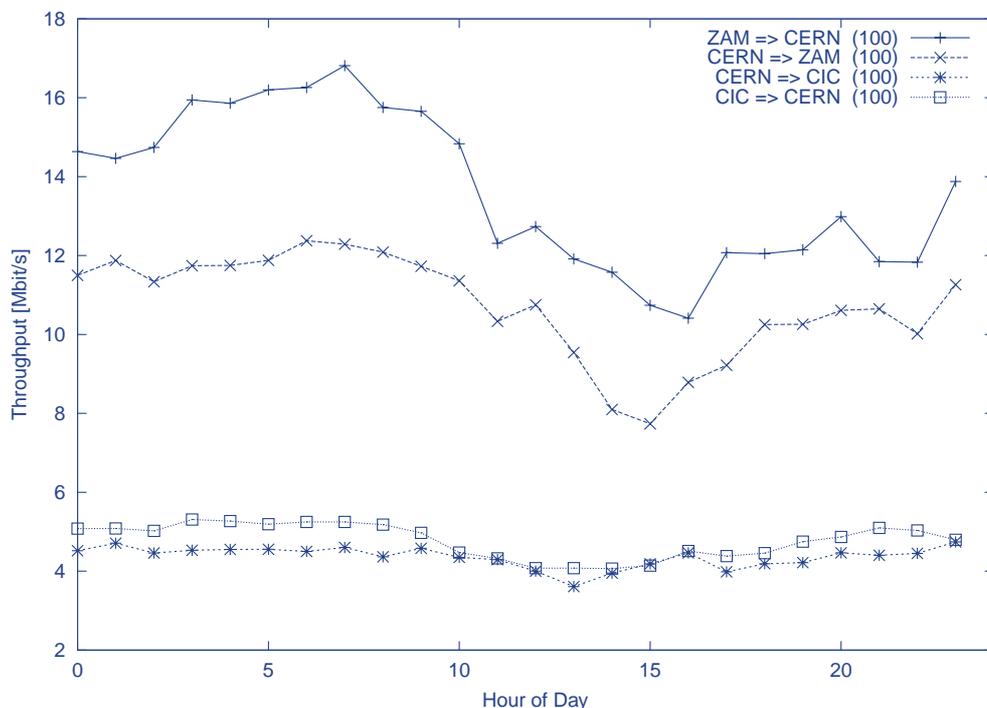


Figure A.8: Average workday (top plot) and weekend (bottom plot) throughput values in batch 1 for the connection Frankfurt -- Geneva. The numbers between brackets are the connection bandwidths in Mbit/s.

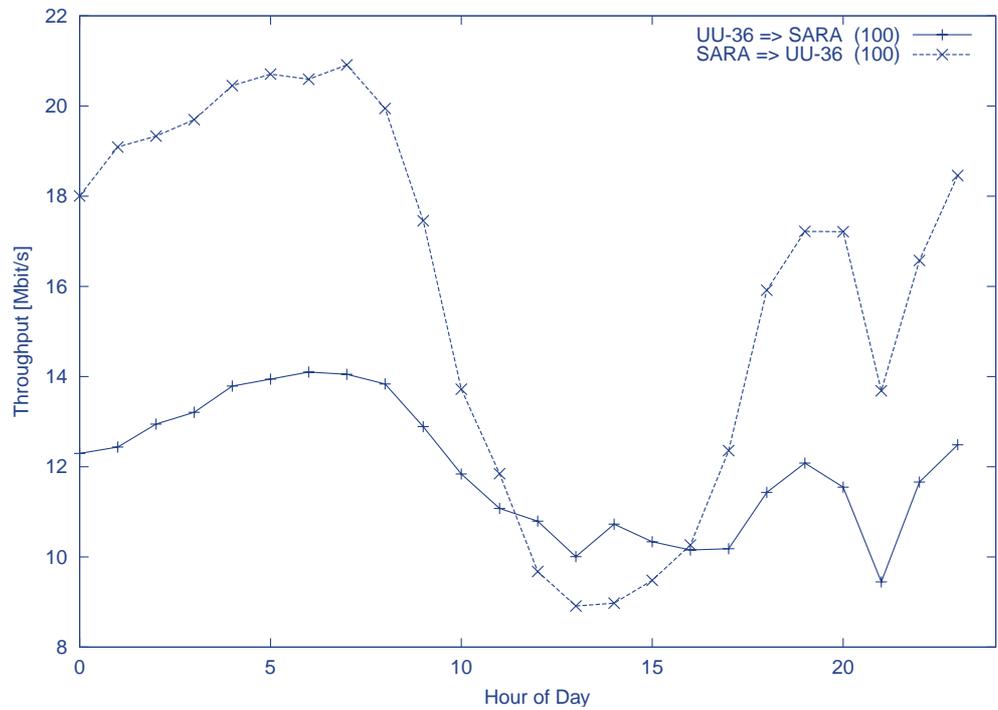


Figure A.9: Average workday throughput values in batch 1 for the SURFnet connection between UU-36 and SARA. The numbers between brackets are the connection bandwidths in Mbit/s.

A.1.2 Batch 2

Table A.2 contains a list of figures for batch 2 that show the throughput results of all sites which are connected by the corresponding direct TEN-155 links (see table 3.3). In addition, figure A.15 shows the workday throughput values for the SURFnet connection between UU-36 and SARA for batch 2.

Connection	Figure	Bandwidth in Mbit/s
London ↔ Amsterdam	A.10	10
Amsterdam ↔ Frankfurt	A.11	100
London ↔ Frankfurt	A.12	10
Amsterdam ↔ Geneva	A.13	10, 100
Frankfurt ↔ Geneva	A.14	100

Table A.2: List of figures for batch 2 that show the throughput results of all sites which are connected by the corresponding direct TEN-155 links between two adjacent PoPs and specifies the bandwidth of the site's connection.

The ULB site in Brussels became unavailable shortly after batch 2 was started. Therefore, no results for connections with Brussels are presented in batch 2 and the London - Paris - Frankfurt connection is displayed as an ongoing link because both sublinks carried the same traffic. The following conclusions can be drawn when comparing the throughput results graphs from batch 1 and 2:

- The peak performances in batch 2 were somewhat higher and more even than in batch 1, while the performance decreases for all throughput results in batch 2 were in general somewhat lower and deeper than in batch 1. This may have been caused by the larger TCP sliding window size used in batch 2 (the sliding window size follows from the socket sizes), which is better when there is little traffic but worse at times of congestion because the adjustment of the sliding window sizes takes more time. An alternative explanation for the performance decreases may be the overall growth of Internet traffic.
- Differences between performance peaks and dips are especially true for ZAM and they are particularly clear from the measurements with CERN (see figures A.8 and A.14). Also remarkable in these plots is the asymmetry in the measurements between ZAM and CERN.
- These performance differences between both batches are also clearly seen in the SURFnet control measurements between UU-36 and SARA (see figures A.9 and A.15).
- A comparison of the SURFnet control measurements (figures A.9 and A.15) with the other throughput results shows that the profiles of plots from the SURFnet and TEN-155 throughput results on workdays had roughly the same shape.
- Besides these relative small differences the overall results of both batches were comparable.

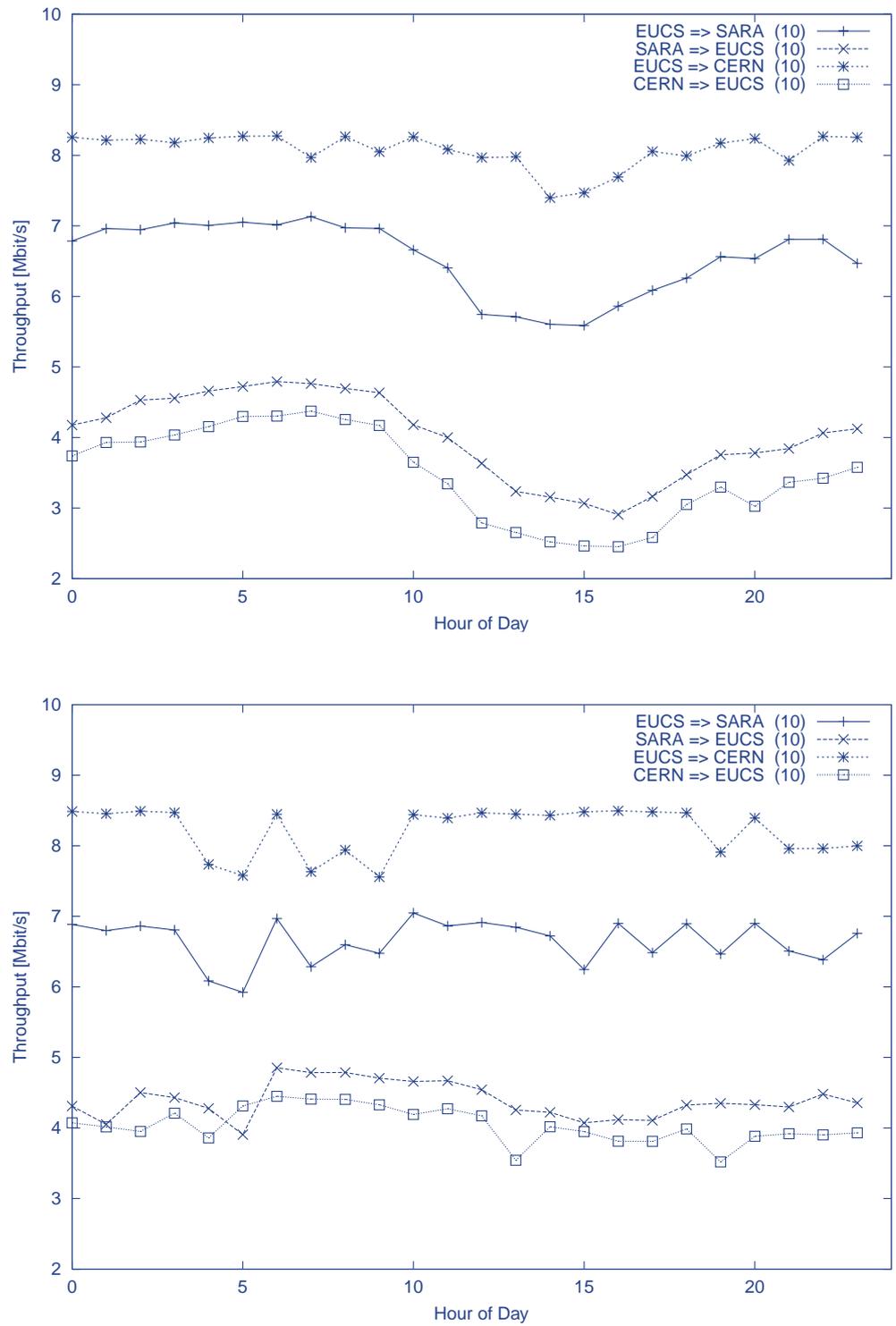


Figure A.10: Average workday (top plot) and weekend (bottom plot) throughput values in batch 2 for the connection London ↔ Amsterdam. The numbers between brackets are the connection bandwidths in Mbit/s.

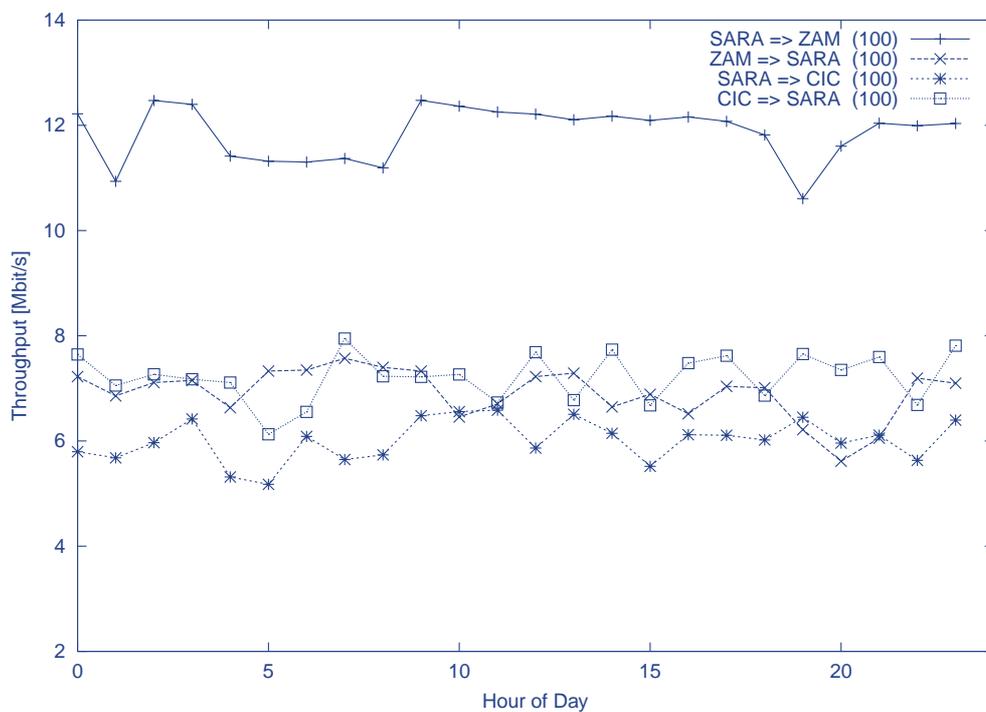
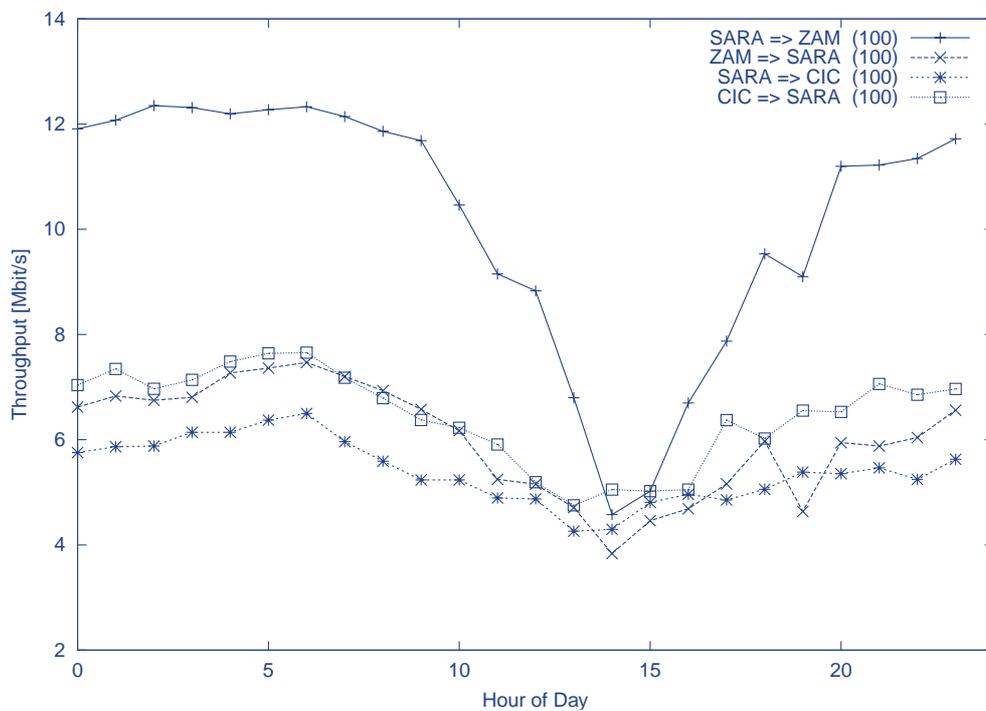


Figure A.11: Average workday (top plot) and weekend (bottom plot) throughput values in batch 2 for the connection Amsterdam ↔ Frankfurt. The numbers between brackets are the connection bandwidths in Mbit/s.

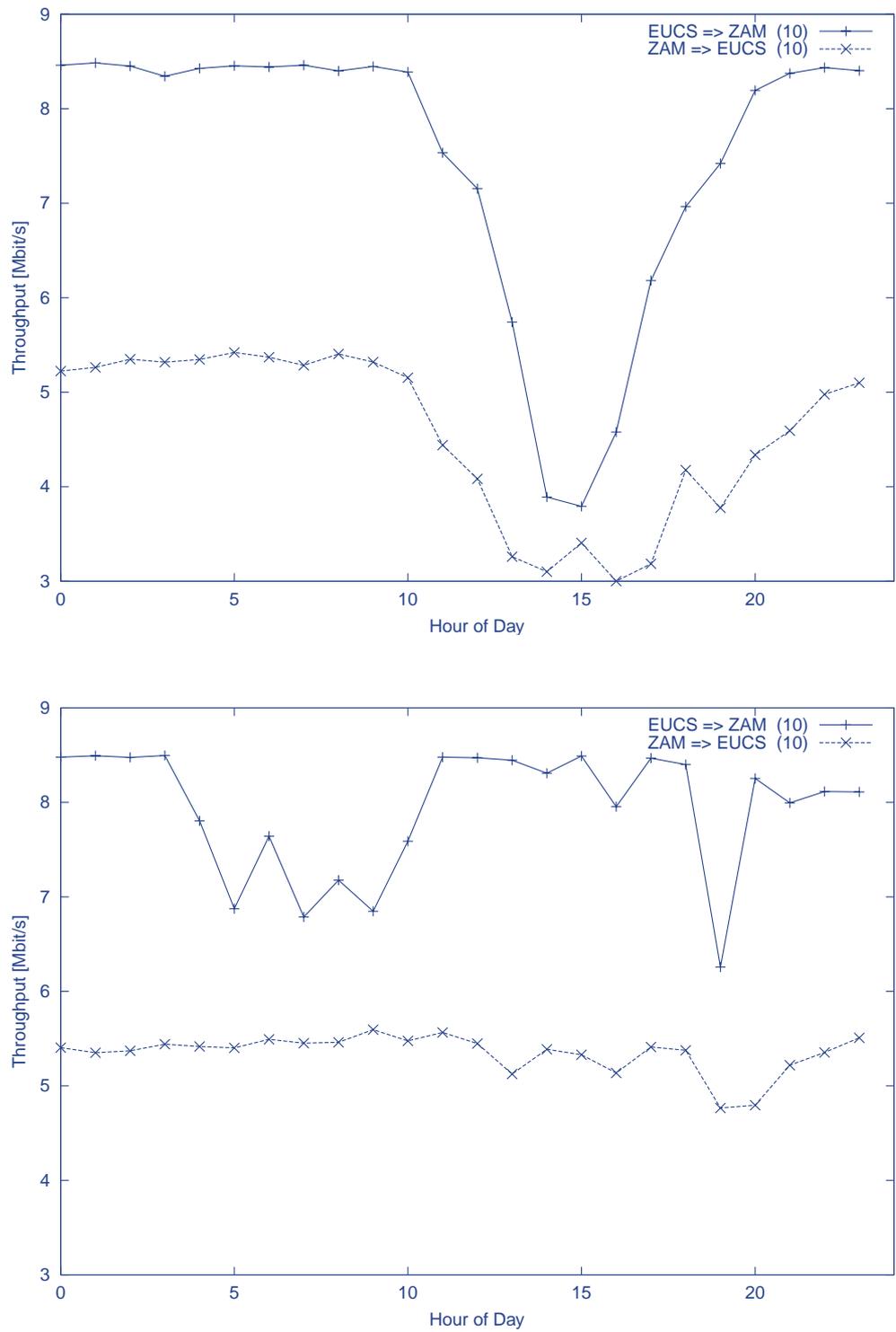


Figure A.12: Average workday (top plot) and weekend (bottom plot) throughput values in batch 2 for the connection London ↔ Frankfurt. The numbers between brackets are the connection bandwidths in Mbit/s.

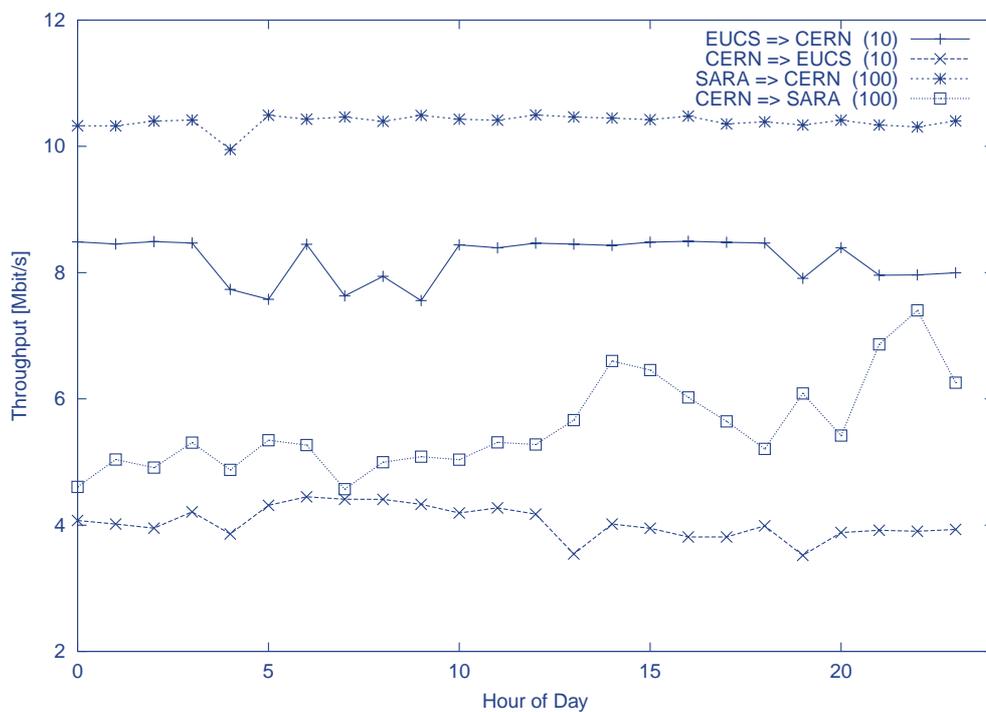
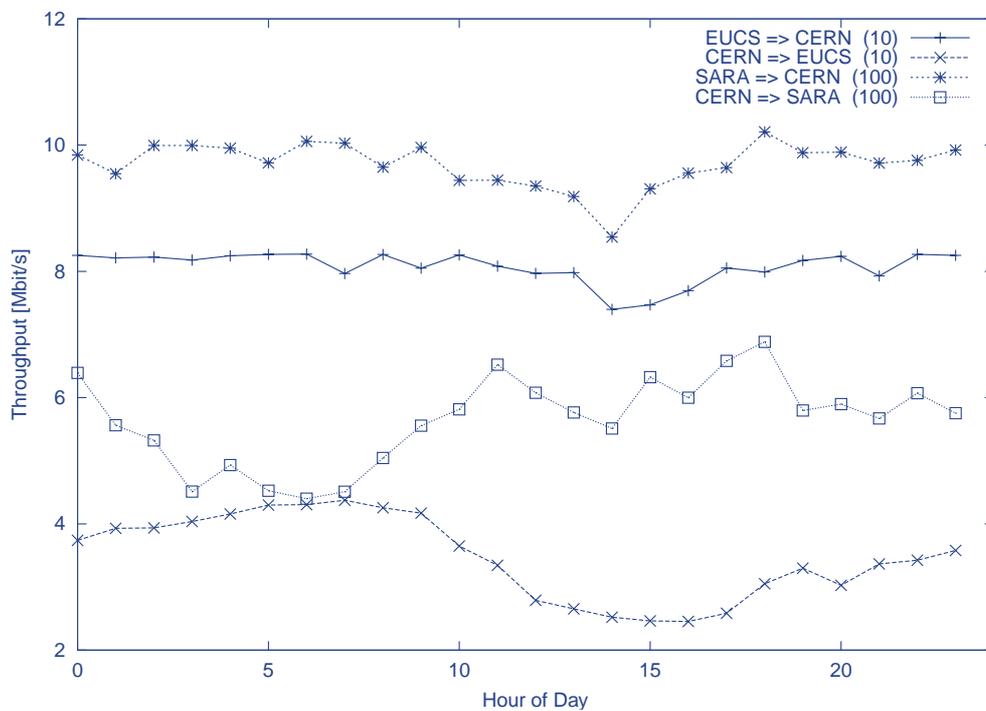


Figure A.13: Average workday (top plot) and weekend (bottom plot) throughput values in batch 2 for the connection Amsterdam → Geneva. The numbers between brackets are the connection bandwidths in Mbit/s

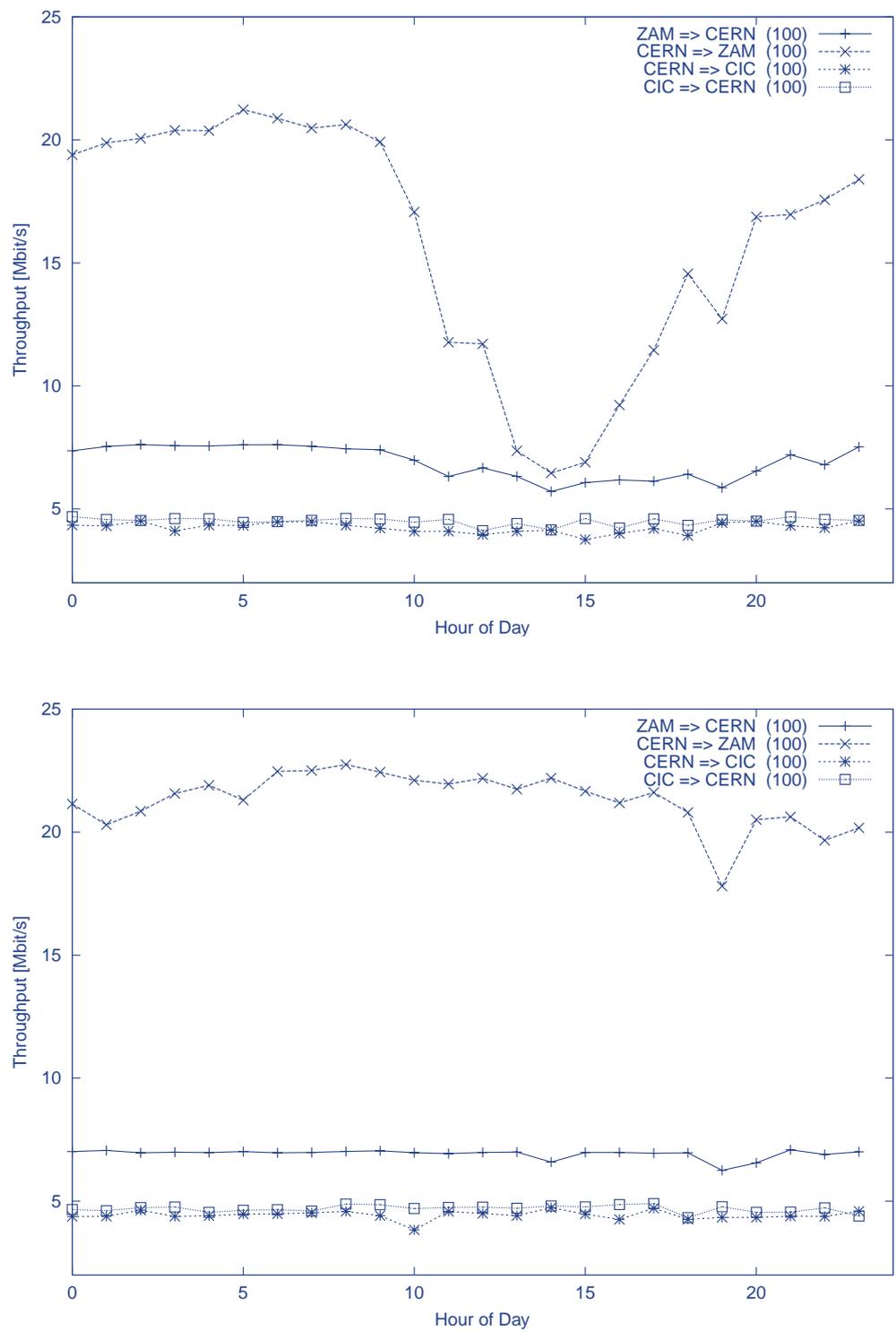


Figure A.14: Average workday (top plot) and weekend (bottom plot) throughput values in batch 2 for the connection Frankfurt → Geneva. The numbers between brackets are the connection bandwidths in Mbit/s.

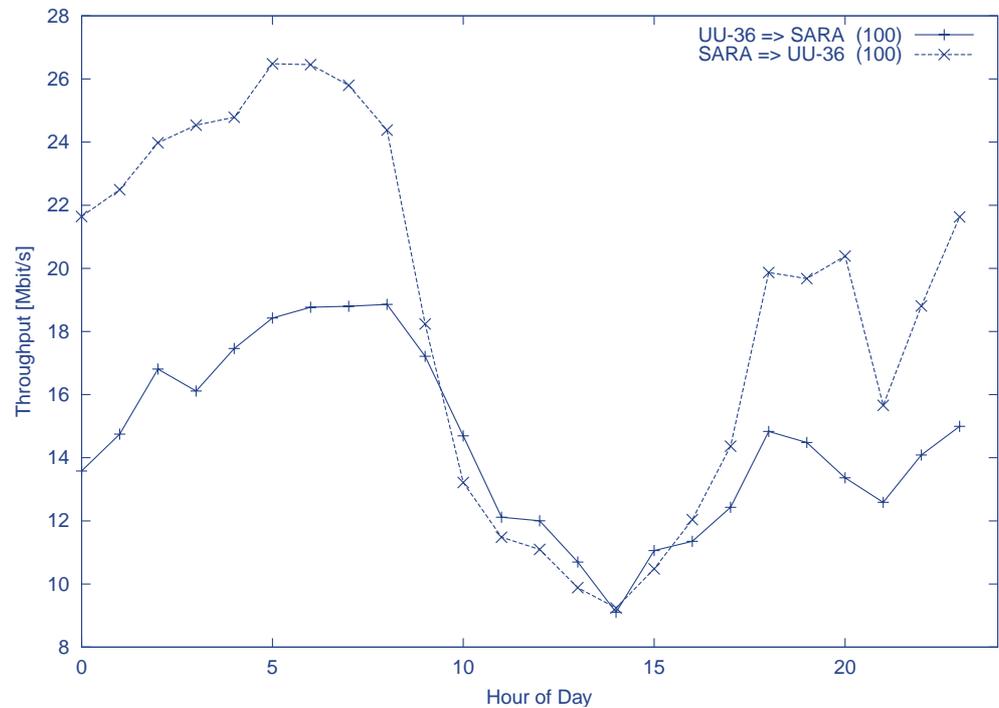


Figure A.15: Average workday throughput values in batch 2 for the SURFnet connection between UU-36 and SARA. The numbers between brackets are the connection bandwidths in Mbit/s.

A.2 Round-trip

In this section we present various aspects of the round-trip results. Due to the direct dependency on unpredictable queuing mechanisms, congestion, etc., round-trip results have a more stochastic nature than the throughput results, where the macroscopic measurement time functioned like a kind of integrator. In the following two sections the minimum round-trip values for all connections in both batches and graphs of typical minimum round-trip times are presented.

A.2.1 Minimum round-trip

Due to the stochastic nature it did not make much sense here to calculate average values as was done in the throughput section. However, the minimum round-trip value represents the limit where all stochastic influences are disappearing. Therefore, it is a measure of the travel time combined with the minimal duration time in the network routers or switches, where the queuing times are also discarded.

To be able to give a stable overview of the minimum round-trip values for the most predominant connections tested in the network performance monitor experiment, a histogram of minimum round-trip values for the connections in both batches was calculated. The bin size was 1 ms. The overall minimum round-trip time was represented by the bin with the largest size, which in most cases was significantly larger than the next largest bin. In our opinion this representation is much more stable and meaningful than an overall minimum round-trip time while taking an average would be violating the limit-like nature of this parameter. The round-trip values obtained for both batches are displayed in table A.3.

From these tables the following conclusions can be drawn:

- Minimum round-trip times corresponded reasonably well with geographical distances.
- The minimum round-trip times of the hosts participating in both batches were similar. For some connections the minimum round-trip times in batch 2 were 1 ms shorter.
- The SARA site is situated near the SURFnet PoP. Therefore, this host could be used to compare round-trip times from direct TEN-155 routes via Amsterdam with the sum of the minimum round-trip times from the sites, using the direct route, to SARA. In batch 2 the only route which could be tested in this way was London - Amsterdam - Geneva (table 3.2) which was used by the sites EUCS and CERN (figure 3.1). The minimum round-trip times, which were the same for both directions, were:

Sum of connections EUCS - SARA and SARA - CERN: 40.0 ms.
 Direct connection EUCS - CERN: 36.5 ms.

Minimum round-trip times batch 1 [ms]

R → C	EUCS	UU-36	ULB	ZAM	CERN	CIC
EUCS	-	23.5	35.5	31.5	37.5	-
UU-36	23.5	-	12.5	17.5	23.5	21.5
ULB	36.5	12.5	-	41.5	32.5	-
ZAM	32.5	17.5	40.5	-	20.5	17.5
CERN	37.5	22.5	31.5	20.5	-	22.5
CIC	-	21.5	-	18.5	22.5	-

Minimum round-trip times batch 2 [ms]

R → C	EUCS	SARA	ZAM	CERN	CIC
EUCS	-	19.5	31.5	36.5	-
SARA	19.5	-	15.5	20.5	18.5
ZAM	31.5	15.5	-	20.5	17.5
CERN	36.5	20.5	20.5	-	22.5
CIC	-	19.5	18.5	22.5	-

Table A.3: Bin value of the largest bin from a histogram taken at all minimum round-trip times for the most tested connections in both batches. The bin size was 1 ms. The tests were performed in the direction from the table rows to the table columns (R → C).

A.2.2 Typical minimum round-trip plots

In this subsection the following plots are presented from typical round-trip measurements in both batches.

- In figure A.16 the minimum round-trip values for all connections with ZAM in week 13 from batch 1 are displayed.
- Figure A.17 shows the minimum round-trip times for all connections with EUCS in week 17 from batch 2.

It appears that the data from EUCS were less stochastic than from ZAM.

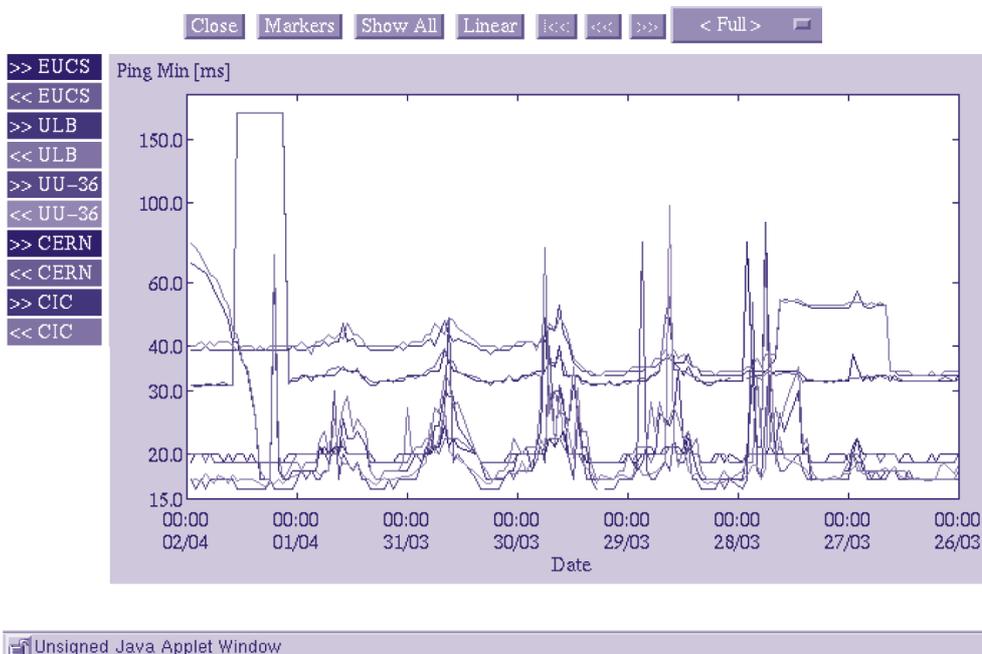


Figure A.16: Minimum round-trip times for all batch 1 connections with ZAM in week 13.

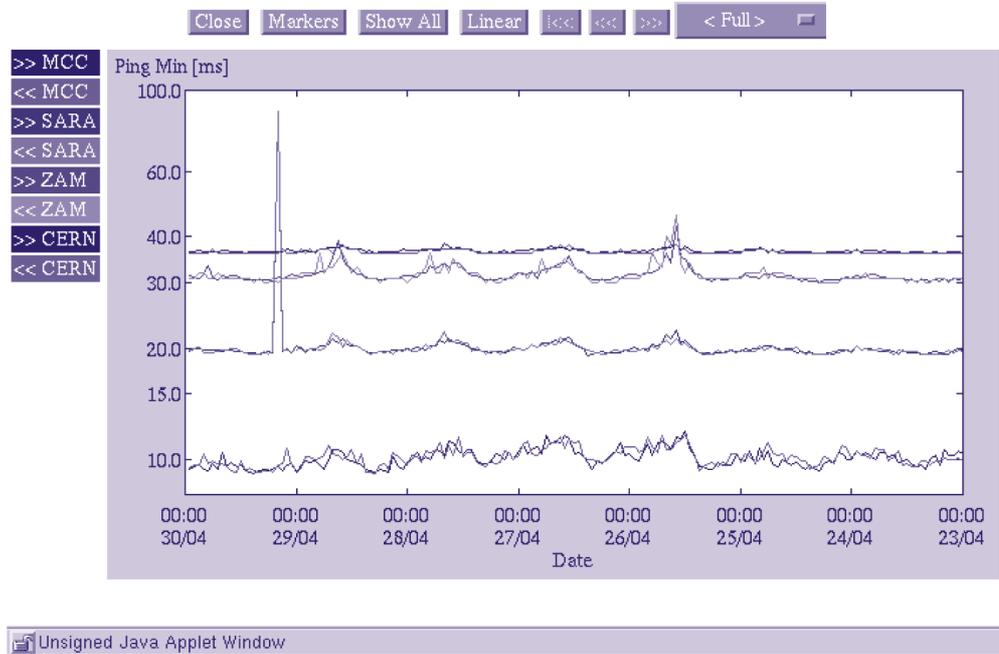


Figure A.17: Minimum round-trip times from batch 2 for all connections with EUCS in week 17.

