Multi-Path TCP in Real-World Setups – An Evaluation in the NORNET CORE Testbed

Thomas Dreibholz
Simula Research Laboratory
Centre for Resilient Networks and Applications
Martin Linges vei 17
1364 Fornebu, Akershus, Norway
dreibh@simula.no

Xing Zhou[1], Fu Fa
Hainan University
College of Information Science and Technology
Renmin Avenue 58
570228 Haikou, Hainan, China
{zhouxing,fufa}@hainu.edu.cn

Abstract—**Nowadays, cloud applications are becoming more and more popular. However, in order for such applications to work, they need a stable Internet connectivity. To avoid the Internet access becoming a single point of failure, redundancy by multi-homing – i.e. simultaneous access to multiple Internet service providers (ISP) – is becoming increasingly common as well. Multi-homing leads to the desire to utilise all network attachment points simultaneously, which is e.g. provided by the Multi-Path TCP (MPTCP) extension for TCP. MPTCP is still under development by researchers and standardisation in the IETF. Particularly, it is necessary to evaluate MPTCP under realistic Internet conditions.**

NORNET CORE is the world's first, large-scale Internet testbed for multi-homed systems and applications. It is therefore a useful platform for evaluating MPTCP. In this paper, we therefore present our NORNET CORE extension that adds MPTCP support to the testbed. Particularly, our extension is now available to all users of NORNET CORE as well, which significantly reduces the effort of MPTCP researchers to evaluate MPTCP and its improvements. In a proof of concept, we furthermore show the strengths and weaknesses of state-of-the-art MPTCP in NORNET CORE, in a configuration covering 29 ISP connections at 14 sites in 5 different countries.[2]

Keywords: NORNET, Multi-Path TCP (MPTCP), Multi-Homing, Multi-Path Transport, Evaluation

I. INTRODUCTION

With cloud computing becoming more and more common, a steadily increasing number of applications nowadays require stable Internet connectivity. However, "anything that can go wrong will go wrong" (Murphy's law) also applies for such applications, and with the Internet connection being a single point of failure, applications become unusable when it fails. So, if Internet connectivity is critical, it must be redundant. Being connected to multiple – hopefully independent – Internet service providers (ISP), which is denoted as *multi-homing*, is therefore becoming increasingly common as well. But when having to pay for multiple ISP connections, which are all working properly during hopefully 99.x% of the time, there is also a growing interest in utilising all connections simultaneously. This is denoted as *multi-path transport*.

However, the Transmission Control Protocol (TCP) [1], which was standardised more than 33 years ago, simply does not support multi-homing (and, of course, not multi-path transport). The Stream Control Transmission Protocol (SCTP) [2] supports multi-homing, and with Concurrent Multipath Transfer for SCTP (CMT-SCTP) [3]–[5] even multi-path transport. But since SCTP is rather new (compared to TCP), it is unknown for and unsupported by many middleboxes like firewalls or Network Address Translation (NAT)/Port Address Translation (PAT) devices. So, deploying it in arbitrary parts of the Internet is not easily possible. Therefore, the TCP extension Multi-Path TCP (MPTCP) [6] is being developed to add multi-homing and multi-path transport to TCP, while maintaining backwards compatibility with existing middlebox devices.

While MPTCP has already been evaluated in certain setups like mobile networks [7], [8] as well as data centre networks [9], and shown to provide certain performance benefits, a larger-scale Internet evaluation in wired networks is still missing. Therefore, in this paper, we introduce our MPTCP extension to the NORNET CORE [10], [11] testbed for multi-homed systems. Based on this extension and as a proof of concept for MPTCP in the NORNET CORE testbed, we then evaluate the performance improvement of state-of-the-art MPTCP in a setup consisting of 14 sites in 5 different countries, and with varying types of network access technologies, in order to show the benefits and weaknesses of today's MPTCP deployment possibilities.

II. MULTI-PATH TCP

The protocol stack [6] of MPTCP is illustrated in Figure 1: The application uses a standard socket API interface to interact with MPTCP, in the same way as with TCP. That is, for the Application Layer, the MPTCP sublayer of the Transport Layer behaves exactly like TCP. So, the application does not have to be aware of MPTCP and can just use it like TCP. This makes a modification of the application unnecessary and, in theory, the same old TCP applications can make use of the performance benefits and redundancy provided by MPTCP. The MPTCP sublayer establishes a separate TCP connection for each path, denoted as *subflow*. A path is defined by the combination of source address and destination address. This is a major difference to SCTP, where a path is only defined by a destination address. The advantage of the MPTCP definition is that all possible paths in the Internet are covered, which is not the case for SCTP [12], [13]. On the Network Layer, both, IPv4 and IPv6 are supported. That is, the same MPTCP

---

[1] Xing Zhou is corresponding author.

[2] Parts of this work have been funded by the Research Council of Norway (Forskingsrådet, prosjektnummer 208798/F50), the National Natural Science Foundation of China (funding numbers 61163014 and 61363008) as well as the International Cooperation Projects of Hainan (funding number KJHZ2013-20).

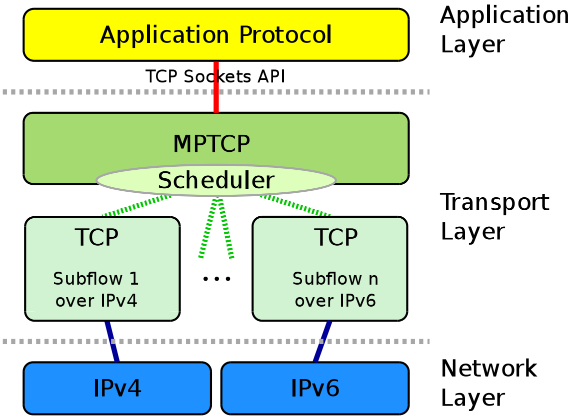


Figure 1. The Multi-Path TCP Protocol Stack

connection – also denoted as MPTCP flow – can at the same time have IPv4- and IPv6-based subflows. The MPTCP scheduler – which is, unlike the protocol behaviour, not yet standardised – decides about the scheduling of data on all available subflows. In case of problems on a path, it may also decide to retransmit data on the same or different subflows.

An MPTCP connection is established by connecting the first TCP subflow with a regular 3-way TCP handshake. The only difference to a regular TCP connection is the usage of the MP_CAPABLE TCP option [6, Subsection 3.1] for signalling MPTCP support. If both sides support it, the new connection becomes an MPTCP connection. Otherwise, the MPTCP side just behaves exactly like TCP. The MPTCP implementation can decide about when to establish further subflows; in order to support all possible paths, it is a full mesh between the local addresses and the remote addresses of each IP protocol. The mapping between a new subflow and an MPTCP connection is signalled by the MP_JOIN option [6, Subsection 3.2]. Clearly, paths can share the same bottleneck. In order to ensure fairness in multi-path setups [14], MPTCP can use resource pooling with coupled congestion control [15], e.g. with the Linked Increases (LIA) algorithm [16].

[17] provides a survey of MPTCP implementations. The two most important ones are the FreeBSD-based implementation[3] by the Centre for Advanced Internet Architectures (CAIA) at Swinburne University [18] as well as the Linux MPTCP implementation[4] by the Université catholique de Louvain (UCL) [19]. The first one is under development with some features still missing, while the latter implementation is by far the most widespread one, and it is used by the IETF MPTCP working group as their reference implementation as well. We are therefore using this implementation, defining the state of the art in MPTCP research and development, for our analysis. Its usage is rather straightforward: MPTCP is activated by setting `sysctl net.mptcp.mptcp_enabled=1`. Then, *all* TCP connections are handled by MPTCP – and use MPTCP if both sides are MPTCP-enabled. That is, the `sysctl` setting acts globally, i.e. there is no possibility to turn MPTCP on or off per socket.

III. The NorNet Testbed

In order to test and evaluate multi-path transport in larger-scale Internet environments, a significant logistical effort is necessary to find remote sites that provide connectivity to multiple ISPs. Therefore, the Simula Research Laboratory [20][5] is establishing the open NorNet testbed[6] [21] for multi-homed systems, consisting of programmable research nodes distributed all over Norway as well as some locations abroad. The NorNet testbed consists of a wireless part, called NorNet Edge [22], as well as the wired part, called NorNet Core [10], [11]. For this paper, we use the wired part and therefore only introduce NorNet Core in more detail here.

The NorNet Core implementation[7] [23]–[25] is based on the PlanetLab [26] testbed software. PlanetLab provides researchers access to virtual Linux machines at remote sites. However, unlike for full virtualisation, it simply uses operating system–level virtualisation to save resources (particularly, memory and disk space). That is, the same kernel is used for all VM instances; they are just separated from each other. For NorNet Core, the development version of the PlanetLab software is used as basis, for the following purposes:

- It is based on Linux Containers (LXC) instead of Linux VServer; it therefore works with unpatched upstream kernels. This allows the usage of modern Linux distributions (currently: Fedora Core 18) and kernels.
- Full IPv6 support is provided.
- The Btrfs [27] file system's cloning feature is utilised to de-duplicate common files among different containers.
- A virtual switch, realized by Open vSwitch, is applied to bridge the containers into the physical network. Particularly, each container gets its own, unshared IPv4 and IPv6 addresses in each of the ISP networks.

It is furthermore possible to instantiate custom virtual machines (e.g. with modified operating systems or kernels) to support special requirements for research experiments.

IV. Supporting MPTCP in NorNet Core

For our MPTCP evaluation experiment, we had to extend NorNet Core appropriately. Our improvements described in this section are now integrated into NorNet Core, providing all users the opportunity to run MPTCP experiments as well.

Clearly, the first step was to actually provide MPTCP support. Therefore, Linux MPTCP – in the stable release 0.88.11 – was integrated together with the stable Linux kernel 3.11.10 into the NorNet Core software distribution. However, due to the limitation of only enabling or disabling MPTCP globally via `sysctl` call, TCP/MPTCP comparisons between the same two endpoints had not been possible. Therefore, we have furthermore developed a patch[8] to add a socket option to the MPTCP API for turning MPTCP support on or off for a certain socket. Our patch actually realises the TCP_MULTIPATH_ENABLE option that is already standardised for MPTCP [28, Subsubsection 5.3.2] but was missing in the implementation. That is, based on the socket option's

---

[3] CAIA FreeBSD MPTCP: http://caia.swin.edu.au/urp/newtcp/mptcp/.

[4] UCL Linux MPTCP: http://www.multipath-tcp.org/.

[5] Simula Research Laboratory: https://www.simula.no.

[6] NorNet testbed https://www.nntb.no.

[7] NorNet Core implementation: http://benlomond.nntb.no/nornet/.

[8] MPTCP socket options patch: http://www.iem.uni-due.de/~dreibh/netperfmeter/download/0001-MPTCP-0.88.11-with-socketoptions.patch.

setting, it is now possible to create pure TCP or MPTCP sockets.

```c
int onoff = 0;
if(setsockopt(sd,
      IPPROTO_TCP, TCP_MULTIPATH_ENABLE,
      &onoff, sizeof(onoff)) < 0) {
   /* handle error condition */
}
```

It is quite straightforward for a researcher to extend his experiment software for TCP/MPTCP comparisons. The new software distribution is now deployed on all NorNet Core research nodes. That is, it is now available for all other NorNet Core users as well. Furthermore, the patch was provided to the upstream MPTCP developers, for inclusion in the next Linux MPTCP release.

NetPerfMeter [29] is an Open Source network performance evaluation tool that allows to evaluate performance like application payload throughput and delays with different transport protocols and concurrent flows. It was already provided in the standard installation of NorNet Core. Therefore, we have adapted NetPerfMeter by adding TCP/MPTCP support. That is, NetPerfMeter is now able to create TCP and MPTCP connections between MPTCP-capable endpoints, with the help of our new socket option. This allows NorNet Core users to run some TCP/MPTCP experiments “out of the box”, without need to install specially-adapted research software first. Of course, it is always possible to install custom software as well.

V. The Experiment Setup

With our new MPTCP extension to NorNet Core, we have set up a proof-of-concept MPTCP experiment.

A. Endpoints

The 29 used endpoints (i.e. providers and sites) for our experiment are listed in Table I. The NorNet Core sites are hosted at universities and research organisations. Therefore, the primary provider of all sites is the corresponding research network provider, i.e. Uninett[9] in Norway, SUNET[10] in Sweden, DFN[11] in Germany, CERNET[12] in China and KanREN[13] in Kansas, U.S.A.. While IPv4 support is of course deployed everywhere, IPv6 is available at most site setups. Although the research network ISPs always provide IPv6 support, IPv6 is simply not yet deployed in some sites’ local networks.

Clearly, research networks are mostly overprovisioned. In order to see the effects of consumer-grade network attachment points, a particular property of NorNet Core is to offer such network attachment points as well. Therefore, the additional connections are a mix of different access technologies, ranging from Asymmetric Digital Subscriber Line (ADSL; like e.g. the PowerTech[14] and Broadnet[15] connections) to consumer fibre lines (like e.g. Altibox[16]) and business-grade fibre connections (e.g. BKK[17]). Wherever available, IPv6 is supported in addition to IPv4 as well. However, unlike for the research network ISPs, it is still challenging go obtain IPv6 access from commercial ISPs.

B. Performance Metric and Research Questions

As performance metric for our proof-of-concept evaluation, we use the application payload throughput. Clearly, this is the most important metric for users of multi-path transport. In our experiment, we would like to answer the following questions:

- Is “MPTCP for everything” (i.e. turning MPTCP support on for all TCP connections and always achieve at least the performance of TCP) nowadays already a good default when using the MPTCP reference implementation (i.e. Linux MPTCP; see Section II)?
- Does MPTCP work with IPv4/IPv6 path combinations as well?
- What are the critical configurations for state of the art MPTCP deployment (i.e. situations where the MPTCP performance is worse than with single-path TCP)?

To answer these questions, we had run at least 20 measurements between each two of the endpoints on different sites given in Table I. Each measurement run consisted of a 30 s bulk data transfer, with a saturated sender. Send and receive buffers were configured to 16 MiB, in order to avoid blocking issues while still keeping the buffers realistically small. The client socket at the local side (i.e. the source endpoint of the bulk data) as well as the server socket at the remote side (i.e. the destination endpoint for the data) had been bound to the corresponding endpoint’s address for TCP and to the *any* address (i.e. 0.0.0.0 for IPv4 and :: for IPv6) for MPTCP. The connection was established by using the remote endpoint’s address. The measurements were made for IPv4 as well as IPv6 sockets separately. Of course, IPv6 runs had only been possible between IPv6-capable endpoints.

All TCP and MPTCP parameters were left at their defaults, in order to get a situation that is realistic for “normal users”. Particularly, this means to use the default Linux congestion control Cubic [30] with uncoupled subflow congestion control of MPTCP flows (since we are using independent ISP subscriptions for the network attachment points; see [14] for general fairness discussions).

VI. Results

A. Gain by Multi-Path Transport Usage

Figure 2 shows the average gain resulting from MPTCP usage in log scale, i.e. $\log(\frac{\text{MPTCP throughput}}{\text{TCP throughput}})$ for the source/destination endpoint relations. The endpoint numbers refer to Table I. Subfigure 2(a) contains the IPv4 values, while Subfigure 2(b) presents the IPv6 results.

For IPv4, it is obvious that MPTCP achieves a significant gain for almost all relations (we will have a detailed look at performance reductions in Subsection VI-B). As expected, the highest performance gains (in red colour) can be achieved when establishing an MPTCP connection via a low-bandwidth ADSL connection (e.g. PowerTech or Broadnet, Endpoints 15 to 22) and adding high-speed subflows (e.g. at the Uninett-connected sites). This also means that MPTCP automatically adapts the scheduling to make use of high-performance paths; the user (or the software) does not need to know the “best” path to establish a connection (by connecting the first subflow;

---

[9] Uninett: https://www.uninett.no.

[10] Swedish University Network (SUNET): http://www.sunet.se.

[11] Deutsches Forschungsnetz (DFN): https://www.dfn.de.

[12] China Research and Educat. Net. (CERNET): http://www.cernet.edu.cn.

[13] Kansas Research and Education Net. (KanREN): http://www.kanren.net.

[14] PowerTech: http://www.powertech.no.

[15] Broadnet: http://www.broadnet.no.

[16] Altibox: https://www.altibox.no.

[17] Bergenshalvøens Kommunale Kraftselskap (BKK): http://bkk.no.

No.	Provider	Attachment Type	Protocol	Site	Location
1	Uninett	Research Fibre	IPv4+IPv6	Simula Research Laboratory	Fornebu, Akershus/Norway
2	Uninett	Research Fibre	IPv4+IPv6	Universitetet i Oslo	Oslo, Oslo/Norway
3	Uninett	Research Fibre	IPv4+IPv6	Høgskolen i Gjøvik	Gjøvik, Oppland/Norway
4	Uninett	Research Fibre	IPv4+IPv6	Universitetet i Tromsø	Tromsø, Troms/Norway
5	Uninett	Research Fibre	IPv4+IPv6	Universitetet i Bergen	Bergen, Hordaland/Norway
6	Uninett	Research Fibre	IPv4	Universitetet på Svalbard	Longyearbyen, Svalbard/Norway
7	Uninett	Research Fibre	IPv4+IPv6	Universitetet i Trondheim	Trondheim, Sør-Trøndelag/Norway
8	Uninett	Research Fibre	IPv4+IPv6	Høgskolen i Narvik	Narvik, Nordland/Norway
9	Uninett	Research Fibre	IPv4+IPv6	Høgskolen i Oslo og Akershus	Oslo, Oslo/Norway
10	Kvantel	Business Fibre	IPv4+IPv6	Simula Research Laboratory	Fornebu, Akershus/Norway
11	Telenor	Consumer ADSL	IPv4	Simula Research Laboratory	Fornebu, Akershus/Norway
12	Telenor	Consumer ADSL	IPv4	Universitetet i Tromsø	Tromsø, Troms/Norway
13	Telenor	Consumer Fibre	IPv4	Universitetet på Svalbard	Longyearbyen, Svalbard/Norway
14	BKK	Business Fibre	IPv4+IPv6	Universitetet i Bergen	Bergen, Hordaland/Norway
15	PowerTech	Consumer ADSL	IPv4+IPv6	Simula Research Laboratory	Fornebu, Akershus/Norway
16	PowerTech	Consumer ADSL	IPv4+IPv6	Universitetet i Oslo	Oslo, Oslo/Norway
17	PowerTech	Consumer ADSL	IPv4+IPv6	Høgskolen i Gjøvik	Gjøvik, Oppland/Norway
18	PowerTech	Consumer ADSL	IPv4+IPv6	Universitetet i Tromsø	Tromsø, Troms/Norway
19	PowerTech	Consumer ADSL	IPv4+IPv6	NTNU Trondheim	Trondheim, Sør-Trøndelag/Norway
20	PowerTech	Consumer ADSL	IPv4+IPv6	Høgskolen i Narvik	Narvik, Nordland/Norway
21	Broadnet	Consumer ADSL	IPv4	Universitetet i Oslo	Oslo, Oslo/Norway
22	Broadnet	Consumer ADSL	IPv4	Høgskolen i Narvik	Narvik, Nordland/Norway
23	SUNET	Research Fibre	IPv4	Karlstads Universitet	Karlstad, Värmland/Sweden
24	DFN	Research Fibre	IPv4+IPv6	Universität Kaiserslautern	Kaiserslautern, Rheinland-Pfalz/Germany
25	DFN	Research Fibre	IPv4+IPv6	Universität Duisburg-Essen	Essen, Nordrhein-Westfalen/Germany
26	Versatel	Consumer ADSL	IPv4	Universität Duisburg-Essen	Essen, Nordrhein-Westfalen/Germany
27	CERNET	Research Fibre	IPv4	Hainan University	Haikou, Hainan/China
28	China Unicom	Consumer Fibre	IPv4	Hainan University	Haikou, Hainan/China
29	KanREN	Research Fibre	IPv4+IPv6	The University of Kansas	Lawrence, Kansas/U.S.A.

Table I
USED NORNET CORE NODES AND INTERFACES

see Section II). So far, for IPv4 sockets, it can clearly be recommended to activate MPTCP globally.

Having a look at the IPv6 results (see Subfigure 2(b)), it is still possible to observe significant performance improvements for most relations. However, the black-coloured points mark relations where MPTCP reaches less than 50% of the TCP performance. Since there is a significant number of such points, it is necessary to investigate performance reductions by MPTCP usage in more detail.

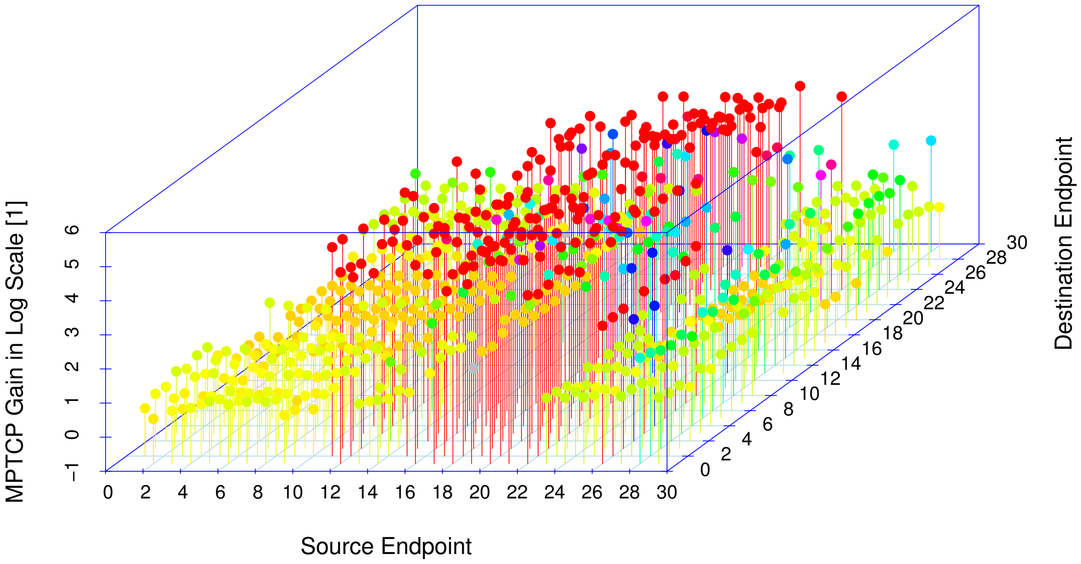
B. Critical Configurations

Figure 3 presents a filtered view of the previous results, showing only points where MPTCP achieves less than 97.5% of the TCP performance. Note, that MPTCP [6] has a slightly higher overhead than TCP. We have therefore set the threshold to 97.5%.

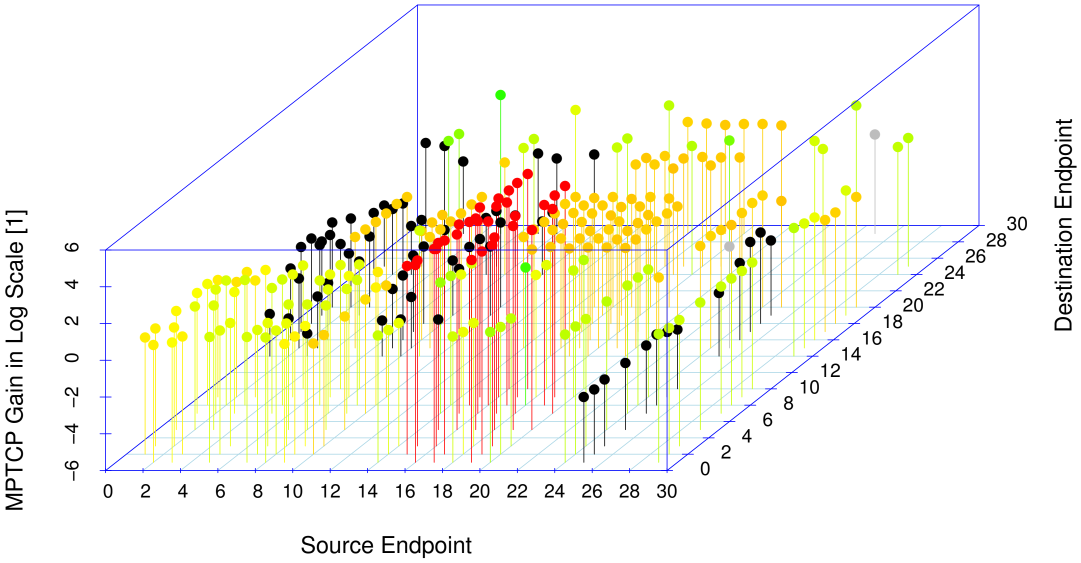
1) IPv4: For IPv4, shown in Subfigure 3(a), only 5 points can be seen. Three of them show a MPTCP performance of at least 80% of the TCP performance (in orange colour), while the remaining two still achieve at least 50% (in grey colour). Note, that 4 of the 5 points have Endpoint 28 (China Unicom at Hainan University/China) as destination. This endpoint is a consumer-grade connection, and it is for all other endpoints at different sites located far away on a different continent. Therefore, such an inter-continental setup is challenging for multi-path transport (see also [12]). Nevertheless, none of all IPv4 MPTCP measurements shows an average performance of less than 50% of the TCP performance.

2) IPv6: However, for IPv6 as shown in Subfigure 3(b), the picture changes significantly: many of the points mark an MPTCP performance of far less than 50% of the TCP performance (in black colour). These points are the most critical ones. Having a closer look, two interesting patterns can be observed: First, no relation from Endpoint 1 and 13 (high-performance fibre lines in Norway) observes a performance reduction with MPTCP. On the other hand, a performance reduction is observable for almost all IPv6 sources sending to Endpoint 25 (DFN at Universität Essen/Germany), which is also a research network connection. Interestingly, Endpoint 24 (DFN at Universität Kaiserslautern/Germany) does not show such a behaviour. Second, a performance loss is shown for many relations from Endpoint 1 and 13 (high-speed) to the ADSL connections (Endpoints 15 to 20) as well as to sites outside of Norway (Endpoint 25 in Germany, Endpoint 29 in the U.S.A.).

That is, MPTCP particularly seems to have issues when sending from a high-performance endpoint to a lower-performance endpoint via an IPv6-capable socket. Note, that when establishing an MPTCP connection with an IPv6 socket, subflows are not restricted to IPv6. Instead, it is possible to use IPv4 subflows as well (see Section II). On the other hand, with an IPv4 socket, MPTCP connections are restricted to IPv4 subflows. That is, during the IPv6 measurements, MPTCP establishes IPv4 subflows over the same ISPs as well, leading to the *possibility* of shared bottlenecks due to some path congruency. Also, we can show in our IPv4/IPv6 path comparison [31] in NORNET CORE, that the path characteristics of both protocols differ frequently. The scheduling and congestion control therefore needs some further evaluation here. That is, care has to still be taken when using MPTCP in IPv4/IPv6 dual-stack setups, and – as future work – further research in such setups is necessary to solve the issues with such configurations.

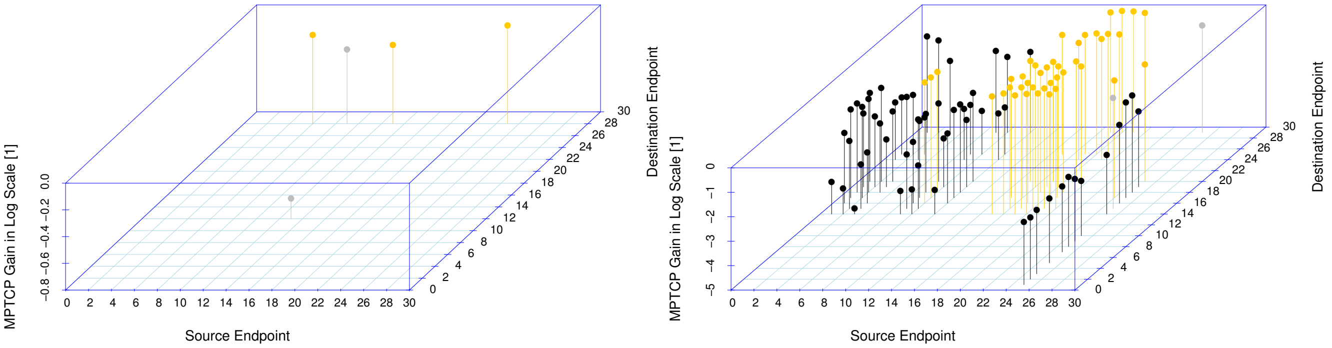


(a) Results for IPv4 Endpoints



(b) Results for IPv6 Endpoints

Figure 2. Log Scale Throughput Gain by Usage of Multi-Path Transport



(a) Results for IPv4 Endpoints

(b) Results for IPv6 Endpoints

Figure 3. Log Scale Throughput Reduction by Usage of Multi-Path Transport

VII. CONCLUSIONS

With multi-homing becoming more and more popular, there is a growing need for multi-path transport. MPTCP, as multi-path transport extension for TCP, is currently under active development in the IETF. However, a larger-scale evaluation in Internet setups is still missing. As part of this work, we have therefore extended the NORNET CORE testbed for multi-homed systems, including the NETPERFMETER performance evaluation software, with MPTCP support. With these extensions, it now becomes relatively easy for researchers to run own MPTCP experiments, in a setup with a set of multi-homed sites in different countries, being connected with different access technologies, over IPv4 as well as IPv6.

In this paper, we have provided a proof of concept for using MPTCP in larger-scale, realistic Internet setups based on the NORNET CORE testbed. We have been able to show that MPTCP already achieves the goal of improving payload throughput in pure IPv4 setups. In combination with IPv6, however, there is still further work necessary. Therefore, our experiment scripts that we have developed for this paper will also be released as Open Source, in order to help MPTCP researchers to improve the protocol, to solve the remaining performance issues and to help the IETF to standardise MPTCP – in order to finally bring the results of multi-path transport research to application by end users. Therefore, even though "the road to hell is paved with unused testbeds" [32], there is good reason to assume that NORNET CORE will be utilised by researchers in order to advance the insights into multi-homed networks and systems for the benefits of all Internet users.

REFERENCES

[1] J. B. Postel, "Transmission Control Protocol," IETF, Standards Track RFC 793, Sep. 1981, ISSN 2070-1721.

[2] R. R. Stewart, "Stream Control Transmission Protocol," IETF, RFC 4960, Sep. 2007, ISSN 2070-1721.

[3] T. Dreibholz, "Evaluation and Optimisation of Multi-Path Transport using the Stream Control Transmission Protocol," Habilitation Treatise, University of Duisburg-Essen, Faculty of Economics, Institute for Computer Science and Business Information Systems, Mar. 2012.

[4] P. D. Amer, M. Becke, T. Dreibholz, N. Ekiz, J. R. Iyengar, P. Natarajan, R. R. Stewart, and M. Tüxen, "Load Sharing for the Stream Control Transmission Protocol (SCTP)," IETF, Individual Submission, Internet Draft draft-tuexen-tsvwg-sctp-multipath-09, Oct. 2014.

[5] J. R. Iyengar, P. D. Amer, and R. R. Stewart, "Concurrent Multipath Transfer using SCTP Multihoming over Independent End-to-End Paths," *IEEE/ACM Transactions on Networking*, vol. 14, no. 5, pp. 951–964, Oct. 2006, ISSN 1063-6692.

[6] A. Ford, C. Raiciu, M. Handley, and O. Bonaventure, "TCP Extensions for Multipath Operation with Multiple Addresses," IETF, RFC 6824, Jan. 2013, ISSN 2070-1721.

[7] S. Ferlin, T. Dreibholz, and Özgü Alay, "Multi-Path Transport over Heterogeneous Wireless Networks: Does it really pay off?" in *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, Austin, Texas/U.S.A., Dec. 2014, pp. 5005–5011, ISBN 978-1-4799-3512-3.

[8] ——, "Tackling the Challenge of Bufferbloat in Multi-Path Transport over Heterogeneous Wireless Networks," in *Proceedings of the IEEE/ACM International Symposium on Quality of Service (IWQoS)*, Hong Kong/People's Republic of China, May 2014, pp. 123–128, ISBN 978-1-4799-4852-9.

[9] C. Raiciu, C. Pluntke, S. Barré, A. Greenhalgh, D. Wischik, and M. Handley, "Data Center Networking with Multipath TCP," in *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks*, Monterey, California/U.S.A., Oct. 2010, pp. 1–6, ISBN 978-1-4503-0409-2.

[10] E. G. Gran, T. Dreibholz, and A. Kvalbein, "NorNet Core – A Multi-Homed Research Testbed," *Computer Networks, Special Issue on Future Internet Testbeds*, vol. 61, pp. 75–87, Mar. 2014, ISSN 1389-1286.

[11] T. Dreibholz and E. G. Gran, "Design and Implementation of the NorNet Core Research Testbed for Multi-Homed Systems," in *Proceedings of the 3nd International Workshop on Protocols and Applications with Multi-Homing Support (PAMS)*, Barcelona, Catalonia/Spain, Mar. 2013, pp. 1094–1100, ISBN 978-0-7695-4952-1.

[12] M. Becke, H. Adhari, E. P. Rathgeb, F. Fa, X. Yang, and X. Zhou, "Comparison of Multipath TCP and CMT-SCTP based on Intercontinental Measurements," in *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, Atlanta, Georgia/U.S.A., Dec. 2013.

[13] H. Adhari, "Does the End-User Really have benefits from using Multipath Transport?" in *Proceedings of the 2nd International NorNet Users Workshop (NNUW-2)*, Fornebu, Akershus/Norway, Aug. 2014.

[14] M. Becke, T. Dreibholz, H. Adhari, and E. P. Rathgeb, "On the Fairness of Transport Protocols in a Multi-Path Environment," in *Proceedings of the IEEE International Conference on Communications (ICC)*, Ottawa, Ontario/Canada, Jun. 2012, pp. 2666–2672, ISBN 978-1-4577-2052-9.

[15] D. Wischik, M. Handley, and M. B. Braun, "The Resource Pooling Principle," *ACM SIGCOMM Computer Communication Review (CCR)*, vol. 38, no. 5, pp. 47–52, Oct. 2008, ISSN 0146-4833.

[16] C. Raiciu, M. Handley, and D. Wischik, "Coupled Congestion Control for Multipath Transport Protocols," IETF, RFC 6356, Oct. 2011, ISSN 2070-1721.

[17] P. L. Eardley, "Survey of MPTCP Implementations," IETF, Individual Submission, Internet Draft draft-eardley-mptcp-implementations-survey-02, Jul. 2013.

[18] N. Williams, L. Stewart, and G. Armitage, "Design Overview of Multipath TCP version 0.4 for FreeBSD-11," Swinburne University of Technology, Centre for Advanced Internet Architectures (CAIA), Melbourne/Australia, Tech. Rep. 140822A, Aug. 2014.

[19] C. Raiciu, C. Paasch, S. Barré, A. Ford, M. Honda, F. Duchêne, O. Bonaventure, and M. Handley, "How Hard Can It Be? Designing and Implementing a Deployable Multipath TCP," in *Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation (NSDI)*, San Jose, California/U.S.A., Apr. 2012, pp. 1–14.

[20] A. Tveito, A. M. Bruaset, and O. Lysne, *Simula Research Laboratory – by thinking constantly about it*. Heidelberg, Baden-Württemberg/Germany: Springer, Nov. 2009, ISBN 978-3642011559.

[21] T. Dreibholz, "NorNet at NICTA – An Open, Large-Scale Testbed for Multi-Homed Systems," Invited Talk at National Information Communications Technology Australia (NICTA), Sydney, New South Wales/Australia, Jan. 2015.

[22] A. Kvalbein, D. Baltrūnas, K. R. Evensen, J. Xiang, A. Elmokashfi, and S. Ferlin, "The NorNet Edge Platform for Mobile Broadband Measurements," *Computer Networks, Special Issue on Future Internet Testbeds*, vol. 61, pp. 88–101, Mar. 2014, ISSN 1389-1286.

[23] T. Dreibholz, J. Bjørgeengen, and J. Werme, "Monitoring and Maintaining the Infrastructure of the NorNet Testbed for Multi-Homed Systems," in *5th International Workshop on Protocols and Applications with Multi-Homing Support (PAMS)*, Gwangju/South Korea, Mar. 2015.

[24] T. Dreibholz, "The NorNet Core Testbed – Introduction and Status in August 2014," in *Proceedings of the 2nd International NorNet Users Workshop (NNUW-2)*, Fornebu, Akershus/Norway, Aug. 2014.

[25] ——, "An Experiment Tutorial for the NorNet Core Testbed," in *Proceedings of the 2nd International NorNet Users Workshop (NNUW-2)*, Fornebu, Akershus/Norway, Aug. 2014.

[26] L. Peterson and T. Roscoe, "The Design Principles of PlanetLab," *Operating Systems Review*, vol. 40, no. 1, pp. 11–16, Jan. 2006, ISSN 0163-5980.

[27] O. Rodeh, J. Bacik, and C. Mason, "BTRFS: The Linux B-Tree Filesystem," *ACM Transactions on Storage*, vol. 9, no. 3, pp. 1–32, Aug. 2013, ISSN 1553-3077.

[28] M. Scharf and A. Ford, "Multipath TCP (MPTCP) Application Interface Considerations," IETF, Informational RFC 6897, Mar. 2013, ISSN 2070-1721.

[29] T. Dreibholz, M. Becke, H. Adhari, and E. P. Rathgeb, "Evaluation of A New Multipath Congestion Control Scheme using the NetPerfMeter Tool-Chain," in *Proceedings of the 19th IEEE International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, Hvar/Croatia, Sep. 2011, pp. 1–6, ISBN 978-953-290-027-9.

[30] S. Ha, I. Rhee, and L. Xu, "CUBIC: A New TCP-friendly High-Speed TCP Variant," *ACM Operating Systems Review (SIGOPS)*, vol. 42, no. 5, pp. 64–74, Jul. 2008, ISSN 0163-5980.

[31] F. Golkar, T. Dreibholz, and A. Kvalbein, "Measuring and Comparing Internet Path Stability in IPv4 and IPv6," in *Proceedings of the 5th IEEE International Conference on the Network of the Future (NoF)*, Paris/France, Dec. 2014, ISBN 978-1-4799-7531-0.

[32] E. K. Çetinkaya and J. P. G. Sterbenz, "Programmable Networking with GpENI," The University of Kansas, Communication Networks Laboratory, Presentation, May 2011.