

Performance Comparison of Congestion Control Strategies for Multi-Path TCP in the NORNET Testbed

Fa Fu*, Xing Zhou*, Thomas Dreibholz[†], Keying Wang*, Feng Zhou* and Quan Gan[‡]

*Hainan University, College of Information Science and Technology, Haikou, China
Email: {fufa,zhouxing}@hainu.edu.cn, {1475062613,969493314}@qq.com

[†]Simula Research Laboratory, Centre for Resilient Networks and Applications, Fornebu, Norway
Email: dreibh@simula.no

[‡]China Unicom, Hainan Branch, Haikou, China
Email: 18608902600@wo.com.cn

Abstract—Multi-path transport has become a hot topic in Internet protocol research with the evolution of emerging technologies, particularly with the market penetration of access terminals having multiple network interfaces (e.g. smartphones with LTE/UMTS and Wi-Fi interfaces). Multi-Path TCP (MPTCP) is an extension of TCP that allows a connection to create several subflows for utilizing multiple network paths. Using multiple end-to-end TCP connections as subflows, MPTCP distributes data to different subflows over multiple ISPs, so as to enhance network robustness and improve throughput.

This paper first presents MPTCP's architecture and multi-path congestion control algorithm concepts. Then, it examines three test scenarios in the NORNET testbed, particularly highlighting the performance difference between using uncoupled and coupled congestion controls in multi-homed, real-world Internet setups. The results show that MPTCP with coupled CCs gets more benefits than TCP and demonstrates the lower aggressiveness in comparison to MPTCP with uncoupled CCs.

Keywords: Multi-Path Transport, Multi-Path TCP (MPTCP), Congestion Control, Performance Analysis

I. INTRODUCTION

Over the past few years, Internet applications have developed a lot, and the needs of users for network bandwidth are also increasing dramatically. But the widely known Transport Layer protocol Transmission Control Protocol (TCP [1]), which only uses a single path to transmit data, exposes its limitations gradually. Meanwhile, as the Internet access technologies have become diverse and the cost of network access equipment reduces, also more and more popular smart mobile devices have several network interfaces. For example, each smartphone is nowadays equipped with at least one LTE/UMTS interface as well as a Wi-Fi and a Bluetooth interface. Also, it is easy to plug in multiple network cards into PCs, in order to connect them to different Internet Service Providers (ISP). This multi-homing capability in fact makes it easily possible to use multiple network interfaces for concurrent multi-path transmission.

¹This work has been funded by 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).

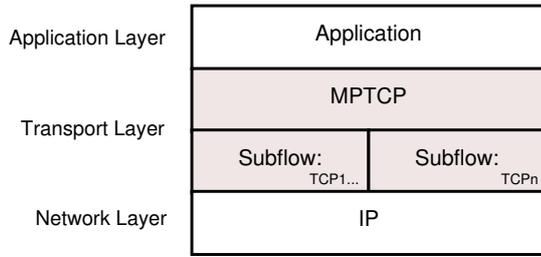
²Xing Zhou is corresponding author.

In October 2009, the Internet Engineering Task Force (IETF) founded the Multi-Path TCP Working Group (MPTCP WG), in order to develop and standardize MPTCP. Its first MPTCP standards document, RFC 6181 [2], was released in 2011. MPTCP is an extension of TCP, and it allows a connection to create several TCP-based subflows for providing multi-path transport services. With multiple subflows, MPTCP distributes data to different paths, so as to enhance network robustness and improve throughput. Furthermore, on the wire, MPTCP subflows look and behave like regular TCP connections. Therefore, MPTCP particularly works over existing middleboxes (firewalls, etc.), which simplifies its deployment possibilities in real-world Internet setups. This makes MPTCP different from the Stream Control Transmission Protocol (SCTP [3]–[5]), which – as “new” Transport Layer protocol – is unsupported by most of today’s middlebox devices.

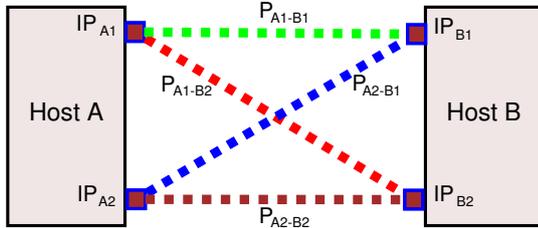
While MPTCP has already been evaluated e.g. in simulations [6] and e.g. in data center setups [7], [8], not many evaluations have been made in larger-scale Internet setups with real “consumer-grade” Internet connections. Therefore, in this paper, we present the results of such Internet measurements with a focus on congestion control strategies. Our paper is structured as follows: first, we introduce the existing congestion control mechanisms for multi-path transfer. After that, the measurement scenario setup based on the NORNET testbed is introduced. This is followed by the evaluation of two uncoupled congestion controls (Cubic and Reno) and four coupled congestion controls (Balía, LIA, OLIA and Weighted Vegas) among Internet sites. While our evaluation is based on MPTCP, it is important to note that the results are generic and may be adapted to other protocols (particularly also to SCTP) as well. Finally, we conclude our work and give an overview of future goals for coupled and uncoupled congestion control mechanisms.

II. THE BASICS OF MULTI-PATH TCP

MPTCP [2], [9] is an extension of TCP that allows to concurrently make use of multiple available paths for data transmission by distributing the data of an MPTCP connection



(a) The Multi-Path TCP Protocol Stack



(b) Paths Combinations Example

Figure 1. MPTCP Architecture and Paths Combinations

to multiple TCP-based subflows. The protocol stack and path management of MPTCP are illustrated in Figure 1. In principle, a full-mesh MPTCP connection consists of several TCP-like connections (called subflows) using the different network paths available. For example, a MPTCP connection between Peer A (P_A) and Peer B (P_B) – as illustrated in Subfigure 1(b) – is initiated by setting up a regular TCP connection between the two endpoints via one of the available paths, e.g., IP_{A1} to IP_{B1} . During the connection setup, the new TCP option `MP_CAPABLE` is used to signal the intention to use multiple paths to the remote peer [9]. Once the initial connection is established, additional sub-connections are added. This is done similar to a regular TCP connection establishment by performing a three-way handshake with the new TCP option `MP_JOIN` present in the segment headers. Performing the three-way handshake for each new subflow is necessary to mimic a regular TCP connection to middleboxes on a path.

By default, MPTCP uses all available address combinations to set up subflows, resulting in a full mesh using all available paths between the endpoints. This differs from the behavior of SCTP, where only a remote peer address defines a path [3]. On the other hand, as [10] shows, this allows utilization of all possible paths. The option `ADD_ADDR` is used to announce an additional IP address to the remote host. In the case of Subfigure 1(b), the MPTCP connection is first set up between IP_{A1} and IP_{B1} . Both hosts then include all additional IP addresses in an `ADD_ADDR` option, since they are both multi-homed. After that, an additional subflow is started between IP_{A2} and IP_{B1} by sending a SYN packet including the `MP_JOIN` option. The same is done with two additional sub-connections between IP_{A2} and IP_{B2} as well as IP_{A1} and IP_{B2} . The result of these operations is the use of 4 subflows, using direct as well as cross paths: P_{A1-B1} , P_{A1-B2} , P_{A2-B1} and P_{A2-B2} . That is, finally, the full mesh of paths has been established.

For mobility support, as well as in order to adapt to changing network conditions, the `ADD_ADDR` option and its opposite `REMOVE_ADDR` for removing addresses can be used to inform a peer about IP address changes during the whole MPTCP connection lifetime.

III. MULTI-PATH CONGESTION CONTROL

Similar to TCP, also MPTCP needs congestion control to adapt the transmission rates on each path to changing network conditions and congestion.

A. Congestion Control Goals

Congestion control is used to adjust the congestion windows within subflows, in order to control each subflow’s data-transmission rate [11]. In order to achieve TCP-friendly Internet deployment, the following three rules [12], [13] of practical multi-path congestion control should be achieved:

- *Rule 1 (“Improve Throughput”)*: A multi-path flow should perform at least as well as a single path flow would on the best of the paths available to it.
- *Rule 2 (“Do no Harm”)*: A multi-path flow should not take up more capacity from any of the resources shared by its different paths than if it were a single flow using only one of these paths. This guarantees it will not unduly harm other flows.
- *Rule 3 (“Balance Congestion”)*: A multi-path flow should move as much traffic as possible off its most congested paths, subject to meeting the first two goals.

Rules 1 and 2 together ensure fairness at the bottleneck. Rule 3 can make use of the concept of resource pooling [14]: if each multi-path flow sends more data through its least-congested path, the traffic in the network will move away from congested areas. This improves robustness and overall throughput, among other things. The way to achieve resource pooling is to effectively “couple” the congestion control loops for the different subflows.

B. Multi-Path Congestion Control

Congestion control strategies for multi-path transport can be categorized into two groups: uncoupled and coupled strategies.

1) *Uncoupled Congestion Control*: Uncoupled congestion control is the simplest form of congestion control for MPTCP. Each subflow is handled like an independent TCP connection, with its own instance of a TCP congestion control. However, this solution is unsatisfactory, as it gives the multi-path flow an unfair [15] share when the paths taken by its different subflows share a common bottleneck.

Reno [16], Cubic [17], BIC, H-TCP, Hybla, Westwood, Vegas etc. belong to uncoupled congestion controls; each strategy has different advantages and disadvantages. Due to space limitations, we can only give a very brief overview here. A good introduction to the algorithms and their underlying ideas can be found in [18]. For example, Vegas adjusts its congestion window size based on the round trip time (RTT) value. Reno is the most widely used congestion control algorithm, but its bandwidth utilization is not very high, and as the network link bandwidths upgrade, this disadvantage will become more and more obvious. BIC may be too aggressive in low RTT and

low-speed networks; Cubic is a modified version of BIC and the current default algorithm for TCP in Linux. During steady state, Cubic increases the congestion window size aggressively when the window is far from the saturation point, and then slowly when it is close to the saturation point. This feature allows Cubic to be very scalable when the bandwidth-delay product of the network is large, and at the same time, be highly stable and also fair to standard TCP flows.

2) *Coupled Congestion Control*: The basic idea to solve the unfairness issue of uncoupled congestion control on shared bottlenecks (see [15] for a detailed discussion) is to couple the congestion windows of all subflows of an MPTCP connection with the resource pooling principle [14]: detecting shared bottlenecks [19] reliably is difficult, but it is just one part of a bigger issue. This bigger question is how much bandwidth a multi-path user should use in total, even if there is no shared bottleneck [15], [20]. The main idea is that by using a coupled congestion control method, the transport protocol can change the congestion window of each subflow and ensure bottleneck fairness and fairness in the broader, network sense. Several approaches to handle this issue are available, for example LIA (Linked Increases Algorithm [12]), OLIA (Opportunistic LIA [21]), Balia (Balanced LIA [22]) and wVegas (Weighted Vegas [23]). Coupled congestion control only applies to the increase phase of the congestion avoidance state, specifying how the congestion window inflates upon receiving an acknowledgement. Other phases are the same as in standard TCP.

IV. SCENARIO SETUP

A. The NORNET CORE Testbed

The NORNET CORE [24], [25] testbed is the world's first, open, large-scale Internet testbed for multi-homed systems and applications. A unique characteristic of NORNET CORE is that most sites are multi-homed to several ISPs. Particularly, it is currently used for research on topics like multi-path transport and resilience. Researchers can run experiments on distributed, programmable nodes which spread over four continents (Europe, Asia, Australia, America) and provide access to multiple different ISPs with different access technologies. Clearly, a key feature of NORNET CORE is to work in the real-world Internet. The all sites' information of the NORNET CORE testbed is shown in Table I [26]. ISPs marked with ⁶ provide IPv6 support as well.

B. The NETPERFMETER Tool

For our evaluations, we are using the NETPERFMETER [5], [27], [28] tool. Compared to other network measurement tools, the key goal of NETPERFMETER is to provide a tool for the performance comparison of multiple transport connections and protocols. That is, it is possible to configure different flows between two systems by using varying parameters, in order to run a configured measurement, collect the obtained results and post-process them for statistical analyses. Particularly, MPTCP is supported by NETPERFMETER as well [29].

C. Scenario Setup

In this paper, we use three different kinds of experiments for analyzing the throughput behavior of different CCs in

the NORNET CORE testbed. The measurements are performed between the Universitetet i Tromsø (UiT) and the Universitetet i Bergen (UiB) sites (see Table I). Each measurement has a duration of 60 seconds and has been repeated 50 times in order to avoid outliers from background traffic noises. The plotted results show the average values of these runs with their corresponding 95% confidence intervals. The MPTCP path management has been set up to "fullmesh", and the send and receive buffer sizes have been set to 1600 KiB (i.e. large enough to cope with the scenarios used in this paper).

1) *Scenario I: Basic TCP Test*: The first experiment between UiT (connected to 3 ISPs) and UiB (connected to 2 ISPs) is aimed at obtaining the basic TCP performance with multiple congestion control strategies: uncoupled Cubic and Reno, as well as coupled LIA, OLIA, Balia and wVegas. At the UiT site, there are the ISPs PowerTech, Telenor and Uninett. PowerTech and Telenor are both consumer-grade Asymmetric Digital Subscriber Lines (ADSL), while Uninett is the Norwegian research network provider (i.e. high-speed, business-grade fiber line). For the UiB site, the two ISPs are respectively Uninett and BKK. Both ISPs provide business-grade fiber lines.

2) *Scenario II: MPTCP Test with Uncoupled Congestion Controls*: In the second scenario, we analyze the MPTCP performance with the uncoupled Cubic and Reno congestion control strategies. That is, we compare MPTCP to the basic TCP behavior.

3) *Scenario III: MPTCP Test with Coupled Congestion Controls*: For the last experiment, we analyze the performance of the coupled Balia, LIA, OLIA and wVegas congestion control strategies, in order to compare coupled congestion control with the basic TCP behavior.

V. RESULTS AND ANALYSIS

A. Basic TCP Results

The results for Scenario I (see Subsubsection IV-C1) are shown in Figure 2. Due to the high bandwidth differences between business-grade fiber links (Uninett, BKK) and ADSL links (PowerTech, Telenor), the figure has been split into a fiber-based part (Subfigure 2(a)) and an ADSL-based part (Subfigure 2(b)). From the basic TCP congestion control comparison, we can see the similar behavior as expected for all congestion controls. Since there is only one path (i.e. a single subflow), coupled congestion control has no effect. The only significant difference – as expected – is the achieved payload throughput: more than 12 Mbit/s over the fiber lines and only around 0.22 Mbit/s over the ADSL upstream links.

Obviously, it strongly depends on the chosen source/destination address pair of a TCP connection whether a high throughput (fiber lines) or a low throughput (ADSL lines) can be achieved. In many cases, however, the user does not know (and should not need to know!) such details. Therefore, multi-path transport should use all paths in a way that it always achieves the best-possible performance for the user.

B. Comparing MPTCP with Uncoupled Congestion Controls

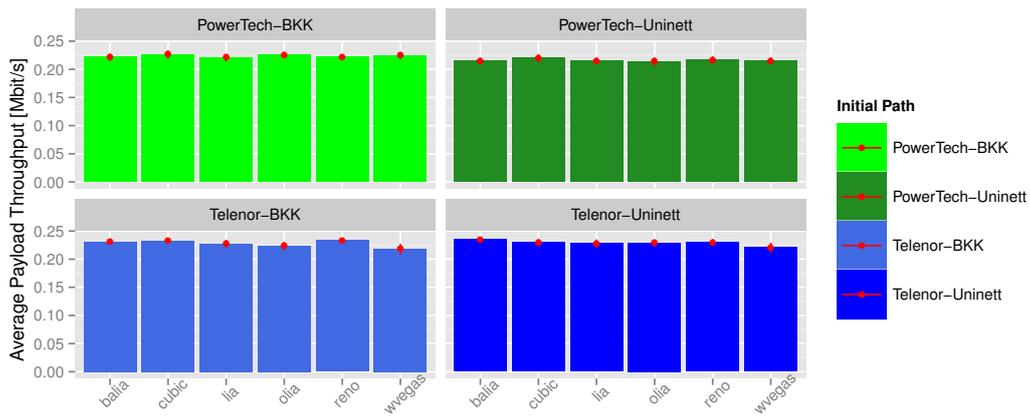
In Scenario II, MPTCP is used (see Subsubsection IV-C2), and Figure 3 presents the measured payload throughput results for using the uncoupled congestion controls Cubic and

Table I
THE SITES OF THE NORNET TESTBED

Index	Site	Abbreviation	ISP 1	ISP 2	ISP 3	ISP 4
1	Simula Research Laboratory	Simula	Uninett ⁶	Kvantel ⁶	Telenor	PowerTech ⁶
2	Universitetet i Oslo	UiO	Uninett ⁶	Broadnet	PowerTech ⁶	–
3	Høgskolen i Gjøvik	HiG	Uninett ⁶	PowerTech ⁶	–	–
4	Universitetet i Tromsø	UiT	Uninett	Telenor	PowerTech ⁶	–
5	Universitetet i Stavanger	UiS	Uninett	Altibox ⁶	PowerTech ⁶	–
6	Universitetet i Bergen	UiB	Uninett ⁶	BKK ⁶	–	–
7	Universitetet i Agder	UiA	Uninett ⁶	PowerTech ⁶	–	–
8	Universitetet på Svalbard	UNIS	Uninett	Telenor	–	–
9	Universitetet i Trondheim	UiT	Uninett ⁶	PowerTech ⁶	–	–
10	Høgskolen i Narvik	HiN	Uninett ⁶	Broadnet	PowerTech ⁶	–
11	Høgskolen i Oslo og Akershus	HiOA	Uninett ⁶	–	–	–
30	Karlstads Universitet	KAU	SUNET	–	–	–
40	Universität Kaiserslautern	TUKL	DFN ⁶	–	–	–
41	Hochschule Hamburg	HAW	DFN ⁶	–	–	–
42	Universität Duisburg-Essen	UDE	DFN ⁶	Versatel	–	–
88	Hainan University	HU	CERNET	China Unicom	–	–
100	The University of Kansas	KU	KanREN ⁶	–	–	–
120	Universidade Federal de São Carlos	UFSCAR	RNP	–	–	–
160	Korea University	Korea	KREONET	–	–	–
200	National ICT Australia	NICTA	AARNet	–	–	–



Congestion Control
(a) Fiber-Based Internet Connections



Congestion Control
(b) ADSL-Based Internet Connections

Figure 2. Basic TCP Congestion Control Comparison

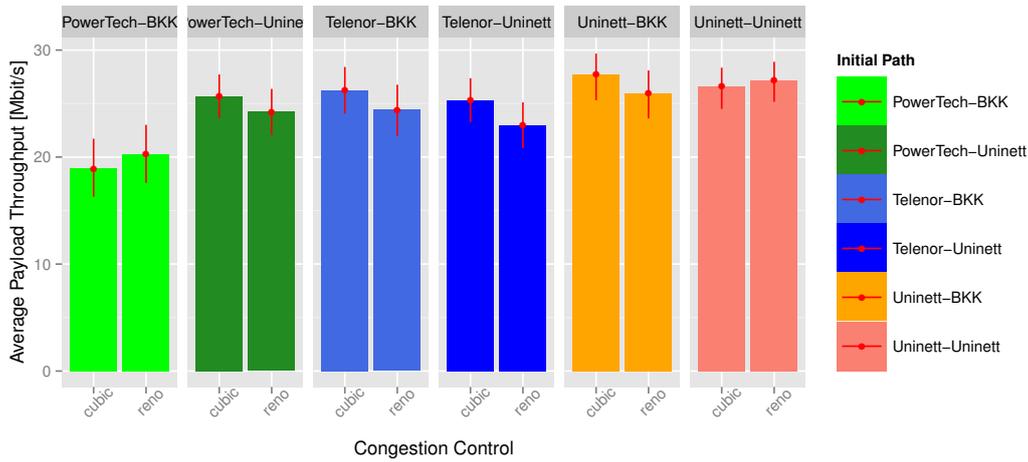


Figure 3. Comparison of Uncoupled Congestion Controls for MPTCP

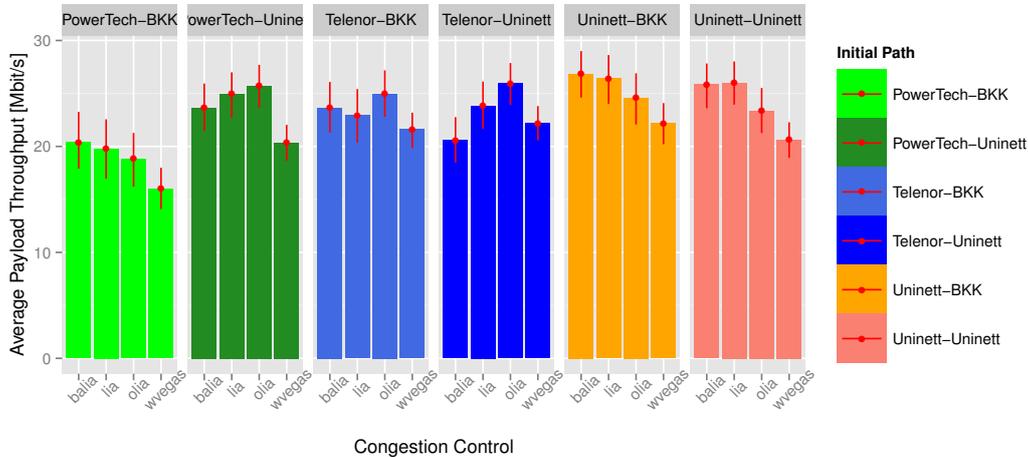


Figure 4. Comparison of Coupled Congestion Controls for MPTCP

Reno. First of all, MPTCP shows the significant advantage of bandwidth aggregation with multi-path transport, regardless of which subflow is chosen as the initial path for connection establishment. That is, the goal of improving the bandwidth is already reached, with values between 20 Mbit/s and 27 Mbit/s.

When establishing the MPTCP connection initially over fiber lines (i.e. Uninett-BKK, Uninett-Uninett; see the two rightmost blocks), the bandwidth is slightly better than for the four ADSL relations (see the first four blocks). The reason is that the MPTCP connection establishment process initiates the subflows sequentially (see Section II). That is, when the connection is initially established over one of the ADSL relations, some time of the 60 s connection runtime (over which the average throughput is measured) passes while only utilizing the slow ADSL paths. However, once all subflows are established, there is no throughput difference any more.

A further observation is the performance difference between Cubic and Reno: in most of the cases, Cubic [17] reaches a slightly better throughput than Reno [16]. Therefore, the usage of Cubic as default for Linux seems to be reasonable.

However, since uncoupled congestion controls are used,

MPTCP behaves unfairly on shared bottlenecks. While the ISP connections are different (and multiple definitions of “fairness” could be discussed [15]), the general “safe” solution is to use a coupled congestion control.

C. Comparing MPTCP with Coupled Congestion Controls

In order to show the effect of coupled congestion controls, Figure 4 for Scenario III (see Subsubsection IV-C3) presents the average throughput results for Balia, LIA, OLIA and wVegas. From the throughput perspective, wVegas [23] achieves the lowest throughput of all four congestion controls in most cases. The performance of Balia [22], LIA [12] and OLIA [21] is better, but without a clear “winner”. However, throughput is only one goal (Rule 1: “Improve Throughput”) of coupled congestion controls, as introduced in Subsection III-A. Important is that they behave fairly (Rule 2: “Do no Harm”) and move congestion (Rule 3: “Balance Congestion”). Considering the different ISP relations in NORNET CORE as disjoint, the price to pay for these two goals is the throughput difference between the uncoupled congestion control case (see Subsection V-B) in Figure 3 and the coupled congestion control case in Figure 4.

In our setup (which should not have major shared bottlenecks), it is mostly only a slight throughput reduction. However, a detailed analysis of the NORNET CORE ISP interconnections and their dynamics [30] is necessary as part of future work.

VI. CONCLUSION

In this paper, we have presented the measurement results for congestion control performance in three multi-homed, real-world Internet scenarios that are based on the NORNET CORE testbed. We have shown that multi-path transport with MPTCP achieves a significant throughput performance benefit in such setups, regardless of the paths chosen for connection establishment. Furthermore, we have shown that uncoupled congestion control with Cubic provides a slightly better performance than Reno. Ensuring fairness to concurrent TCP flows by using coupled congestion controls comes with a slight performance penalty, even in our setup with different ISPs. However, the overall performance improvement of multi-path transport remains significant.

As part of future work, after the successful proof of concept in this paper, we are going to make a large-scale analysis in the NORNET CORE testbed, examining the performance among the various, globally distributed sites of the testbed. Our goal is to help MPTCP researchers to improve the algorithms, to solve the remaining performance issues and to help the IETF to standardize MPTCP – in order to finally bring the results of research on multi-path transport to the application by “normal” Internet users.

REFERENCES

- [1] J. B. Postel, “Transmission Control Protocol,” IETF, Standards Track RFC 793, Sep. 1981, ISSN 2070-1721.
- [2] M. B. Braun, “Threat Analysis for TCP Extensions for Multipath Operation with Multiple Addresses,” IETF, Informational RFC 6181, Mar. 2011, ISSN 2070-1721.
- [3] R. R. Stewart, “Stream Control Transmission Protocol,” IETF, RFC 4960, Sep. 2007, ISSN 2070-1721.
- [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-tuxen-tsvwg-sctp-multipath-10, May 2015.
- [5] 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.
- [6] M. Becke, “Revisiting the IETF Multipath Extensions on Transport Layer,” Ph.D. dissertation, University of Duisburg-Essen, Faculty of Economics, Institute for Computer Science and Business Information Systems, Nov. 2014.
- [7] 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.
- [8] 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.
- [9] 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.
- [10] 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.
- [11] S. C. Nguyen and T. M. T. Nguyen, “Evaluation of Multipath TCP Load Sharing with Coupled Congestion Control Option in Heterogeneous Networks,” in *Proceedings of Global Information Infrastructure Symposium (GIIS)*, Da Nang/Vietnam, Aug. 2011, pp. 1–5, ISBN 978-1-4577-1262-3.
- [12] C. Raiciu, M. Handley, and D. Wischik, “Coupled Congestion Control for Multipath Transport Protocols,” IETF, RFC 6356, Oct. 2011, ISSN 2070-1721.
- [13] C. Raiciu, D. Wischik, and M. Handley, “Practical Congestion Control for Multipath Transport Protocols,” University College London, London/United Kingdom, Tech. Rep., Nov. 2009.
- [14] 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.
- [15] 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.
- [16] M. Allman, V. Paxson, and E. Blanton, “TCP Congestion Control,” IETF, Standards Track RFC 5681, Sep. 2009, ISSN 2070-1721.
- [17] 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.
- [18] M. Welzl, *Network Congestion Control: Managing Internet Traffic*. Chichester, West Sussex/United Kingdom: John Wiley & Sons, 2005, ISBN 978-0-470-02528-4.
- [19] M. M. Yousaf, M. Welzl, and B. Yener, “On the Accurate Identification of Network Paths having a Common Bottleneck,” in *Proceedings of the ACM SIGCOMM Conference*, Seattle, Washington/U.S.A., Aug. 2008, poster Presentation.
- [20] F. Fa, X. Zhou, X. Yang, H. Adhari, and E. P. Rathgeb, “Performance Analysis of MPTCP and CMT-SCTP Multi-Path Transport Protocols,” *Computer Engineering and Applications*, vol. 49, no. 21, pp. 79–82, Nov. 2013, ISSN 1002-8331.
- [21] R. Khalili, N. G. Gast, M. Popović, and J.-Y. L. Boudec, “Opportunistic Linked-Increases Congestion Control Algorithm for MPTCP,” IETF, Individual Submission, Internet Draft draft-khalili-mptcp-congestion-control-05, Jul. 2014.
- [22] A. Walid, Q. Peng, J. Hwang, and S. H. Low, “Balanced Linked Adaptation Congestion Control Algorithm for MPTCP,” IETF, Individual Submission, Internet Draft draft-walid-mptcp-congestion-control-03, Jul. 2015.
- [23] M. Xu, Y. Cao, and E. Dong, “Delay-based Congestion Control for MPTCP,” IETF, Individual Submission, Internet Draft draft-xu-mptcp-congestion-control-02, Jul. 2015.
- [24] 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.
- [25] T. Dreibholz and E. G. Gran, “Design and Implementation of the NorNet Core Research Testbed for Multi-Homed Systems,” in *Proceedings of the 3rd 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.
- [26] T. Dreibholz, “The NorNet Testbed – A Large-Scale Experiment Platform for Real-World Experiments with Multi-Homed Systems,” Invited Talk at Swinburne University, Centre for Advanced Internet Architectures (CAIA), Melbourne, Victoria/Australia, Jan. 2015.
- [27] —, “NetPerfMeter: A Network Performance Metering Tool,” *Multipath TCP Blog*, Sep. 2015.
- [28] 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.
- [29] T. Dreibholz, X. Zhou, and F. Fa, “Multi-Path TCP in Real-World Setups – An Evaluation in the NorNet Core Testbed,” in *5th International Workshop on Protocols and Applications with Multi-Homing Support (PAMS)*, Gwangju/South Korea, Mar. 2015, pp. 617–622, ISBN 978-1-4799-1775-4.
- [30] 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, pp. 611–616, ISBN 978-1-4799-1775-4.