

# Investigating the Throughput Performance of the MPT-GRE<sup>1</sup> Network Layer Multipath Library in Emulated WAN Environment

S. Szilágyi<sup>1,\*</sup>, I. Bordán<sup>1</sup>

<sup>1</sup>University of Debrecen, Faculty of Informatics  
26 Kassai Way, 4028, Debrecen, Hungary  
\* e-mail: szilagyi.szabolcs@inf.unideb.hu

Submitted: 10/07/2021; Accepted: 27/08/2021; Published online: 18/10/2021

**Abstract:** Nowadays there is a growing demand for a much faster and more secure communication without borders through the internet, which is provoking more and more both network designers and manufacturers of communication devices. Thanks to the BYOD trend, our communication devices can be used at work. They generally have several built-in network interfaces (e.g. Ethernet, Wi-Fi, 4G). Theoretically, using interface/connection in parallel, we could speed up data transmission, and thus communication, by aggregating the channel capabilities of the interfaces. On the other hand, we could make data transmission more reliable by applying redundancy to the system. Unfortunately, traditional IP-based communications do not allow the use of parallel interfaces in a given communication session, leaving the hardware capabilities of our communications devices virtually untapped. To address this issue, we have developed a multipath communication solution called MPT-GRE, which we have already tested in several laboratory environments. The measurement results were published in our previous articles. In this paper we are going to test it in a much more realistic environment, using the Dummynet WAN emulation software. The measurement results confirmed that the MPT-GRE multipath solution is able to aggregate the performance of physical connections efficiently in the emulated Fast Ethernet IPv4 WAN environment as well.

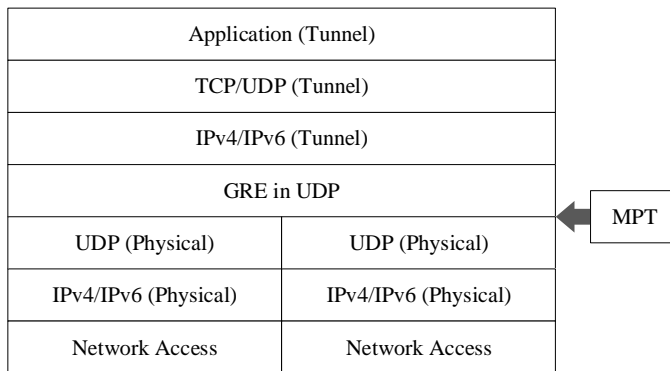
---

<sup>1</sup>MPT-GRE is a MultiPath Technology developed by our research group at the University of Debrecen, Hungary. Its operating principle is based on the Generic Routing Encapsulation in UDP standard. Practically, MPT-GRE is a multipath extension of the GRE-in-UDP standard described in RFC8086.

*Keywords: MPT-GRE; multipath communication; Dummynet; throughput; WAN Emulator*

## 1. Introduction

This paper is an extended version of our former conference paper [1].



*Figure 1. Layered architecture of the MPT-GRE*

Multipath communication technologies are one of the hot research topics nowadays. What better proof of this than Apple and Cisco integrating MPTCP<sup>2</sup>, considered as the flagship of multipath technologies, into their operating systems. With the help of multipath communication, we can increase throughput, while also employing redundant data paths.

In our earlier publications (see e.g. [2], [3], [4], [5], [6], [7], [8], [9]) we have presented a multipath communication technology (MPT-GRE<sup>3</sup> [10]) developed by our research group, which we have built on the standardized GRE-in-UDP tunneling technology<sup>4</sup>. The layered architecture of the MPT-GRE can be seen in Fig. 1.

The MPT-GRE software reads the input packet (IPv4 or IPv6 packet) from the tunnel interface at the sender site. This packet is encapsulated into a new UDP

<sup>2</sup>The MPTCP Project official website: <https://www.multipath-tcp.org/>

<sup>3</sup>The MPT-GRE Project official website: <https://irh.inf.unideb.hu/~szilagyi/index.php/en/mpt/>

<sup>4</sup>GRE-in-UDP Encapsulation standard: <https://tools.ietf.org/html/rfc8086>

segment, and it is sent out to a path chosen from the multiple paths possibilities. At the receiver site the header of the incoming UDP segment is de-encapsulated, and the data (which is the original packet coming from the sender’s tunnel interface) is transmitted to the tunnel interface of the receiver host. The (logical) connection between the tunnel interfaces of the peers is a direct, point-to-point connection.

We have examined its effectiveness with the help of numerous scenarios in our test environment, comparing results with MPTCP as a reference. All of the scenarios have showed that our MPT-GRE solution is capable of efficient path-aggregation both in Fast Ethernet and Gigabit Ethernet IPv4/IPv6 environments. The testing environments we have used previously (see e.g. Fig. 2) can be considered ideal in the sense that they did not contain any network environment parameters that could negatively affect network performance (e.g. delay, jitter, packet-loss). For this reason, we find it important to further examine the effectiveness of MPT-GRE in a more realistic environment (see e.g. Fig. 3).

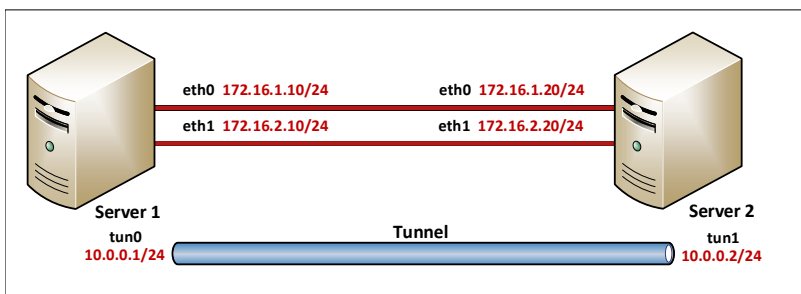


Figure 2. Our previous “ideal” testbed

When wanting to test a newly developed networking software in a realistic environment, we practically have three possibilities:

- internet
- network simulation
- network emulation

Using the internet can be a given in case of single-path communication. However, to be able to test multipath systems, the presence of dual-home technology is essential,

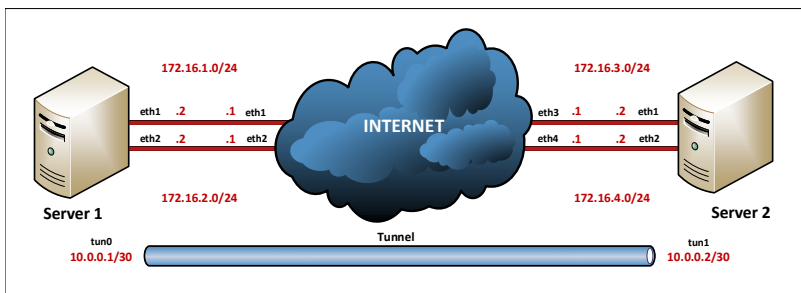


Figure 3. A real network testbed example

i.e. we need to have multiple ISP connections available.

Network simulation aims to replicate the key parameters of desired network environments with the help of mathematical models, with greater or lesser success.

The essence of network emulation is to replicate real network behaviors. Two main types exist:

- hardware realization (see e.g. Fig. 4)
- software implementation

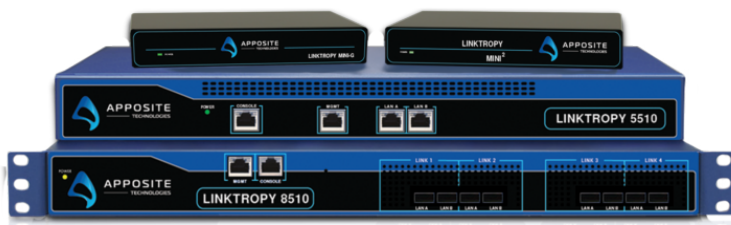


Figure 4. Hardware-based network emulators

The first contains advanced technological solutions, but it is in turn quite a costly method. The latter is not always capable of providing a reliable and precise test

environment, but it is cost-effective. Some examples of network emulator software: Dummy Cloud<sup>5</sup>, Dummynet<sup>6</sup>, NETEM<sup>7</sup>, NIST Net<sup>8</sup>, SoftPerfect Connection Emulator<sup>9</sup>, WANem<sup>10</sup>.

Given that a hardware implementation of a WAN emulator sufficient for our goals would be around 6000 EUR + VAT<sup>11</sup>, after having reviewed the software solutions, our choice was Dummynet.

The Dummynet WAN emulator was developed in 2010 at the University of Pisa, and later got integrated into the FreeBSD operating system<sup>12</sup>. It provides a suitable framework for testing multipath solutions, enabling the setup of packet-delay, jitter and packet-loss network parameters [11]. It also has good documentation, including numerous code examples<sup>13</sup>.

## 2. Measurement environment

To perform our measurements, we created two types of environments, namely a dual-path Fast Ethernet IPv4 and a dual-path Gigabit Ethernet IPv4 WAN emulated measurement environment (see Fig. 5). We downloaded a 1 GB file from the fileserver on the left onto the server on the right. Network parameters were controlled on the intermediate server that had Dummynet installed on the kernel level.

All three machines were running the Linux Ubuntu operating system. We examined the effect of packet-delay, jitter and packet-loss on file download speed, download time, and CPU performance. Bash and Python scripts – available on our website<sup>14</sup> – were used to automate the measurement process. We repeated each series of measurements ten times.

---

<sup>5</sup>Dummy Cloud official website: <http://www.dummycloud.com/>

<sup>6</sup>Dummynet Project official website: <http://info.iet.unipi.it/luigi/dummynet/>

<sup>7</sup>NetEm's manual page: <https://man7.org/linux/man-pages/man8/tc-netem.8.html>

<sup>8</sup>NIST Net home page: <https://www-x.antd.nist.gov/nistnet/>

<sup>9</sup>SoftPerfect Connection Emulator: <https://www.softperfect.com/products/connectionemulator/>

<sup>10</sup>WANem official web page: <http://wanem.sourceforge.net/>

<sup>11</sup>Linktropy Mini-G's price at June 2021: <https://www.digital-hands.eu/products/apposite-network-emulation/linktropy-mini-g/>

<sup>12</sup>FreeBSD Manual Pages: <https://www.freebsd.org/cgi/man.cgi?dummynet>

<sup>13</sup>Using Dummynet in FreeBSD: [http://noahdavids.org/self\\_published/using\\_dummynet.html](http://noahdavids.org/self_published/using_dummynet.html)

<sup>14</sup>Our test scripts can be downloaded from: <https://nas01.inf.unideb.hu/share.cgi?ssid=03CsnIS>

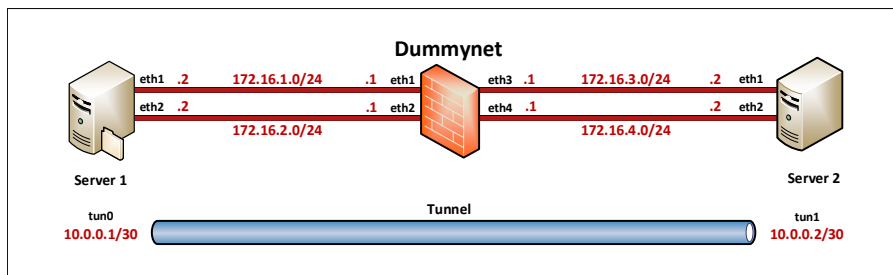


Figure 5. Our new measurement testbed with Dummynet

### 3. Measurement results

#### 3.1. Measurement results in dual-path Fast Ethernet IPv4 WAN emulated environment

First, we checked how packet-delay affected download speed (see Fig. 6).

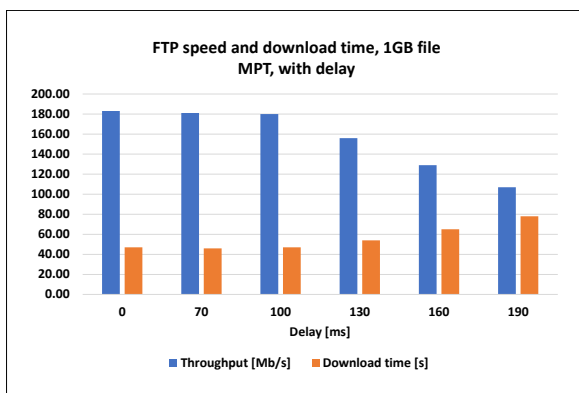


Figure 6. The effect of the delay on the FTP throughput and download time

We gradually increased delay values on a scale of 0-190 ms on the first path using Dummynet. Everything proved to be stable until 100 ms. Above 100 ms, we experienced a continuous decrease in file download speeds. Using the 190 ms delay

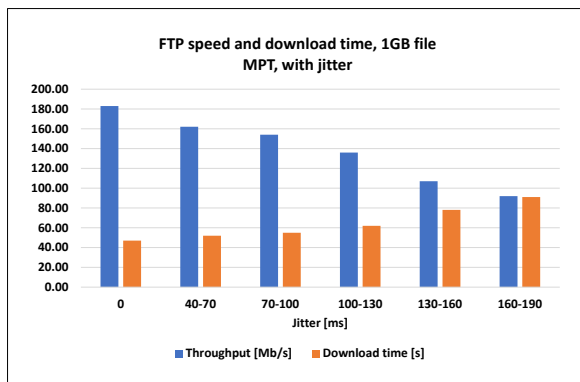


Figure 7. The effect of the jitter on the FTP throughput and download time

value, the download speed decreased to 107 Mb/s, while download time increased from 47 seconds to 78 seconds

A similar effect could be experienced in the case of increasing jitter values on a single path (see Fig. 7). With a 160-190 ms delay fluctuation, the download speed practically decreased by half, while the download time doubled.

Applying even a minimal data-loss rate (1 %), we witnessed a drastic performance decline (see Fig. 8). The download speed fall to a quarter, while the download time quadrupled. Therefore, we did not experiment with further data-loss rate values.

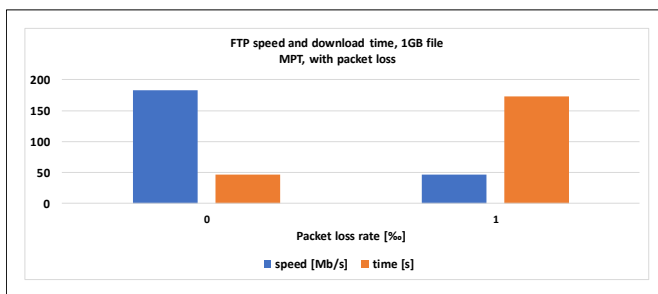


Figure 8. The effect of the packet loss on the FTP throughput and download time

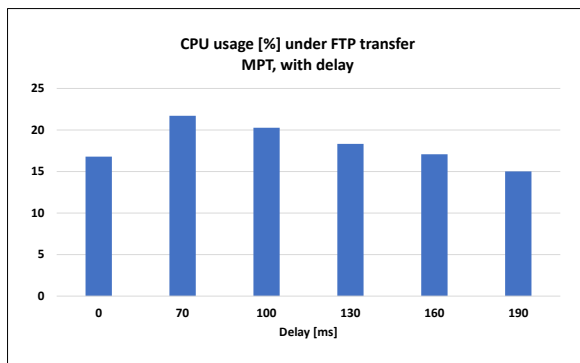


Figure 9. The effect of the delay on the CPU usage on Server 2

Regarding the effect of packet-delay on CPU performance, we did not experience significant fluctuation (see Fig. 9). CPU utilization hovered between 15-22% in every case.

Introducing jitter however, had noticeable effects on CPU utilization (see Fig. 10). With higher jitter values, we experienced a drop in CPU utilization.

While examining CPU loads, the effect of packet-loss also proved to be drastic (see Fig. 11). Using a data-loss rate of 1 %, utilization dropped from 17 to 5.7 percent.

We also carried out further measurements, mixing the parameters of the different paths. E.g. using only delay on one path, while using only jitter on the other. These scenarios brought similar results as well.

### 3.2. Measurement results in dual-path Gigabit Ethernet IPv4 WAN emulated environment

We also extended our measurements to a Gigabit Ethernet emulated WAN environment, examining the same network parameters as for the Fast Ethernet one. Unfortunately, we found from the outset that, for some reason, MPT is unable to function effectively in such a medium. A minimum (1 ms) delay also reduced the file download speed to less than 1Gb/s, and when using jitter, the transfer rate was a few kb/s. For this reason, we installed MPTCP, which was used as a reference for our measurement

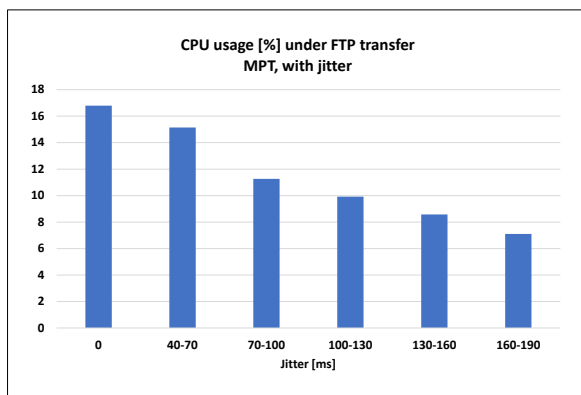


Figure 10. The effect of the jitter on the CPU usage on Server 2

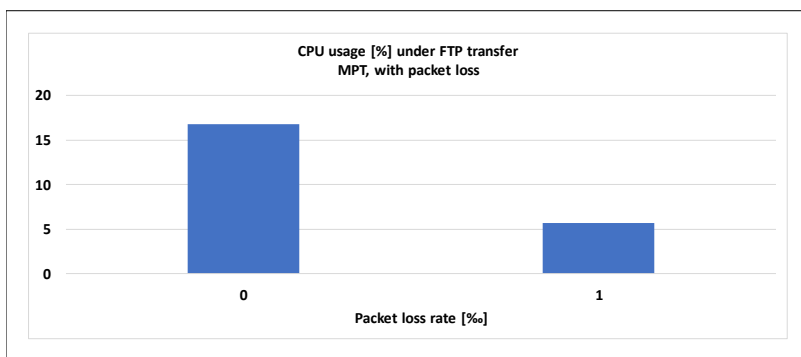


Figure 11. The effect of the packet loss on the CPU usage

series. In the following, we make a performance comparison of the two multipath communication solutions.

In the first round, we examined the effect of the delay on the file download speed, indicating the reference value measured in single path communication session in each case. As shown in Fig. 12, in a delay-free environment, both environments effectively summed the available route capacity. Gradually increasing the rate of latency, MPT performance dropped below 250 Mbps, while MPTCP performed excellently.

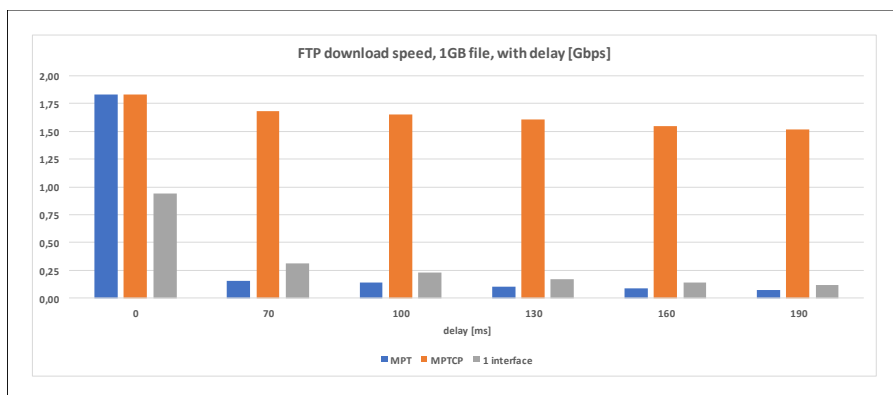


Figure 12. The effect of the delay on the FTP throughput in case of MPT-GRE and MPTCP

With the use of jitter (see Fig. 13), the download speed of the MPT was reduced to 400 kbps already in the first step (applying 40-70 ms jitter), therefore we didn't even try any more measurements. On the other hand, MPTCP was able to produce throughput around 260 Mbps.

As for the effect of the data loss rate (see Fig. 14), even a 1% rate drastically degrades MPT's performance. In fact, it achieves almost the performance of a single-path environment.

Based on our measurements, it can be stated that the delay itself has a minimal effect on CPU performance (see Fig. 15). Using a delay of 70-190 ms, MPT performed slightly better than MPTCP. In both cases, the CPU utilization rate is below 10%.

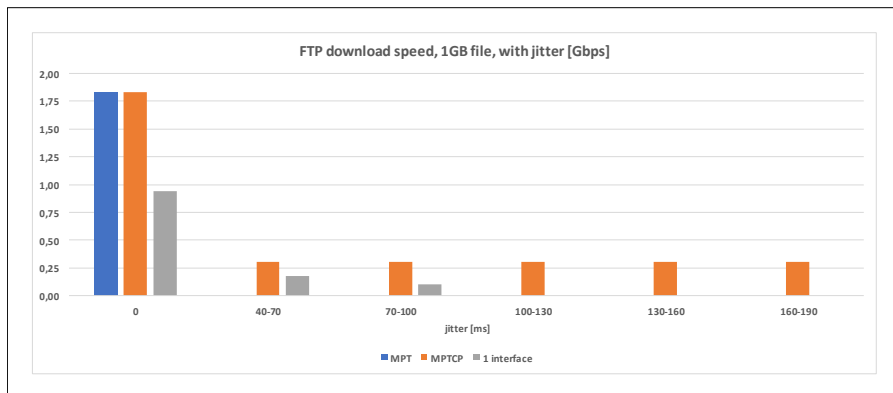


Figure 13. The effect of the jitter on the FTP throughput in case of MPT-GRE and MPTCP

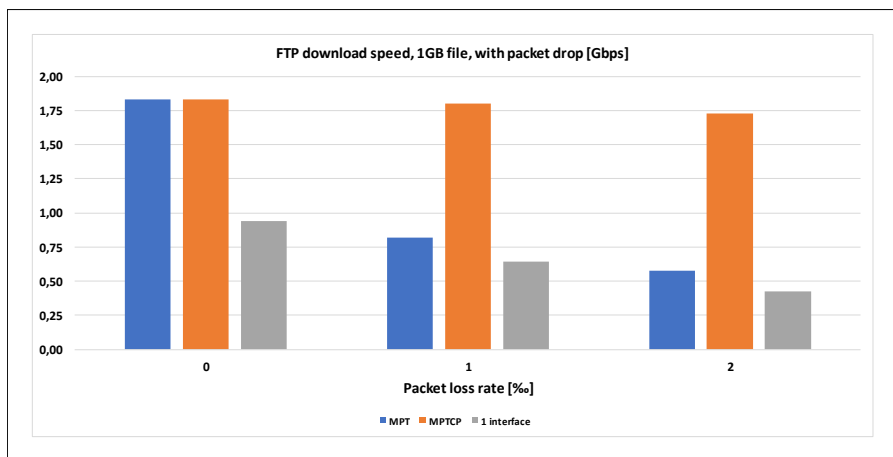


Figure 14. The effect of the packet loss on the FTP throughput in case of MPT-GRE and MPTCP

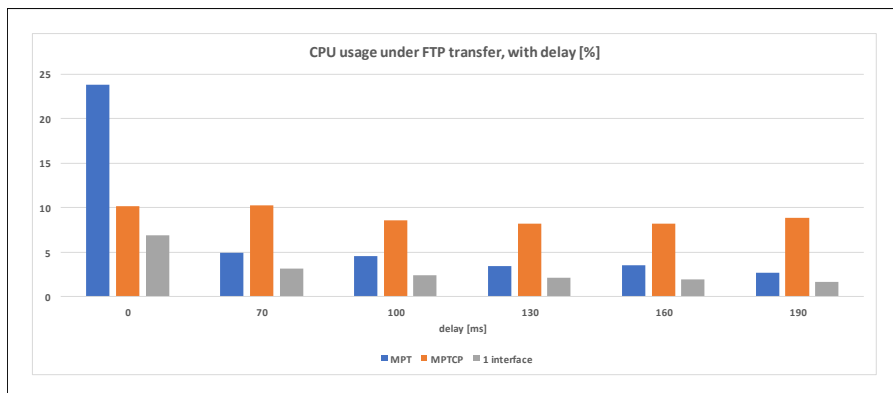


Figure 15. The effect of the delay on the CPU usage in case of MPT-GRE and MPTCP

However, the rate of data loss has a more significant effect on MPT’s CPU utilization (see Fig. 16). Applying a data loss rate of 1‰, the CPU utilization of the MPT is 18%, while that of the MPTCP is around 9%.

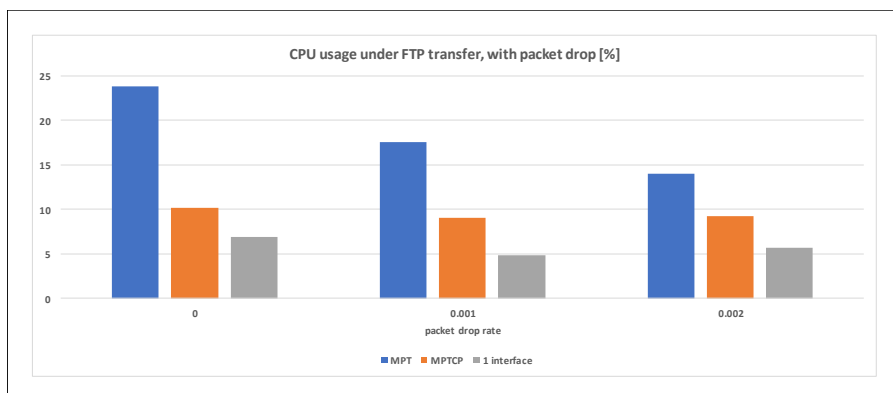


Figure 16. The effect of the packet loss on the CPU usage in case of MPT-GRE and MPTCP.

## **4. Conclusion**

In our current paper, we extended the performance-analysis of our own multipath solution, MPT-GRE, using an emulated WAN environment. Firstly, we examined the effect of different network parameters, like e.g. packet-delay, jitter and data-loss rate on file download speed, download time, and CPU utilization in dual-path Fast Ethernet IPv4 WAN emulated environment. The worst performance we experienced was with the application of the 1 % packet-loss rate. We further extended our measurements to a Gigabit Ethernet environment. In this case, the MPT performed very poorly, even with the introduction of a minimum delay, jitter or data loss rate. Because of these, we extended our measurements to the reference MPTCP multipath environment (see e.g. [12], [13], [14], [15], [16]). In his case, too, the results of the measurements were much weaker than expected, but he definitely performed better than the MPT. As a self-check, MPT- and MPTCP-free baseline measurements were performed in a single-path environment. Even so, very poor results were obtained. Finally, we concluded that the Dummynet WAN emulation environment may not be the most suitable solution for testing multipath Gigabit Ethernet environments and that MPT-GRE needs to be further optimized for more efficient performance.

Among our future goals we plan to improve our MPT solution, testing other software-based WAN emulators, and we also would like to get hands-on experience with the capabilities offered by hardware WAN emulators.

## **Acknowledgement**

This work was supported by the construction EFOP-3.6.3-VEKOP-16-2017-00002. The project was supported by the European Union, co-financed by the European Social Fund.

## **References**

- [1] S. Szilágyi, I. Bordán, Throughput Performance Measurement of the MPT-GRE Multipath Technology in Emulated WAN Environment, in: Proceedings of the 1st Conference on Information Technology and Data Science: CITDS 2020, Debrecen, Hungary, 2020, pp.187–195.  
URL <http://ceur-ws.org/Vol-2874/short17.pdf>

- [2] S. Szilágyi, I. Bordán *et al.*, Throughput Performance Comparison of MPT-GRE and MPTCP in the Gigabit Ethernet IPv4/IPv6 Environment, *Journal of Electrical and Electronics Engineering* 12 (1) (2019) pp. 57–60.  
URL [https://electroinf.uoradea.ro/images/articles/CERCETARE/Reviste/JEEE/JEEE\\_V12\\_N1\\_MAY\\_2019/10%20paper%200414%20SZILAGYI.pdf](https://electroinf.uoradea.ro/images/articles/CERCETARE/Reviste/JEEE/JEEE_V12_N1_MAY_2019/10%20paper%200414%20SZILAGYI.pdf)
- [3] B. Almási, S. Szilágyi, Throughput Performance Analysis of the Multipath Communication Library MPT, in: Proceedings of the 36th International Conference on Telecommunications and Signal Processing: TSP 2013, Rome, Italy, 2013, pp.86–90.  
doi: <https://doi.org/10.1109/TSP.2013.6613897>
- [4] B. Almási, S. Szilágyi, Investigating the Throughput Performance of the MPT Multipath Communication Library in IPv4 and IPv6, *International Journal of Advances in Telecommunications, Electrotechnics, Signals and Systems* 5 (1) (2016) pp. 53–60.  
doi: <https://doi.org/10.11601/ijates.v5i1.148>
- [5] B. Almási, G. Lencse, S. Szilágyi, Investigating the Multipath Extension of the GRE in UDP Technology, *Computer Communications* 103 (1) (2017) pp. 29–38.  
doi: <https://doi.org/10.1016/j.comcom.2017.02.002>
- [6] S. Szilágyi, I. Bordán *et al.*, MPT-GRE: A Novel Multipath Communication Technology for the Cloud, in: Proceedings of the 9th IEEE International Conference on Cognitive Infocommunications: CogInfoCom 2018, Budapest, Hungary, 2018, pp.81–86.  
doi: <https://doi.org/10.1109/CogInfoCom.2018.8639941>
- [7] S. Szilágyi, F. Fejes, R. Katona, Throughput Performance Comparison of MPT-GRE and MPTCP in the Fast Ethernet IPv4/IPv6 Environment, *Journal of Telecommunications and Information Technology* 3 (2) (2018) pp. 53–59.  
doi: <https://doi.org/10.26636/jtit.2018.122817>
- [8] S. Szilágyi, I. Bordán *et al.*, Throughput Performance Analysis of the Multipath Communication Technologies for the Cloud, *Journal of Electrical and Electronics Engineering* 12 (2) (2019) pp. 69–72.  
URL [https://electroinf.uoradea.ro/images/articles/CERCETARE/Reviste/JEEE/JEEE\\_V12\\_N2\\_OCT\\_2019/13%20paper%201008%20SZILAGYI.pdf](https://electroinf.uoradea.ro/images/articles/CERCETARE/Reviste/JEEE/JEEE_V12_N2_OCT_2019/13%20paper%201008%20SZILAGYI.pdf)
- [9] S. Szilágyi, I. Bordán, The Effects of Different Congestion Control Algorithms over Multipath Fast Ethernet IPv4/IPv6 Environments, in: Proceedings of the

11th International Conference on Applied Informatics: ICAI 2020, Eger, Hungary, 2020, pp.341–349.

URL <http://ceur-ws.org/Vol-2650/paper35.pdf>

- [10] G. Lencse, S. Szilágyi *et al.*, MPT Network Layer Multipath Library, Internet Draft, version 8, 2021. [cited 2021-06-30]

URL <https://datatracker.ietf.org/doc/html/draft-lencse-tsvwg-mpt-08>

- [11] D. T. Tieu *et al.*, Automatic test framework for video streaming quality assessment, in: 2015 2nd National Foundation for Science and Technology Development Conference on Information and Computer Science: NICS, IEEE, 2015, pp. 214–218.

doi: <https://doi.org/10.1109/NICS.2015.7302193>

- [12] T. Zhang, S. Zhao, B. Cheng, Multipath Routing and MPTCP-Based Data Delivery Over Manets, *IEEE Access* 8, (2020) pp. 32652–32673.

doi: <https://doi.org/10.1109/ACCESS.2020.2974191>

- [13] J. Huang *et al.*, Tuning high flow concurrency for MPTCP in data center networks, *Journal of Cloud Computing* 9 (13), (2020) pp. 32652–32673.

doi: <https://doi.org/10.1186/s13677-020-00160-3>

- [14] P. Ignaciuk, M. Morawski, Discrete-Time Sliding-Mode Controllers for MPTCP Networks, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, (2019) pp. 1–11.

doi: <https://doi.org/10.1109/TSMC.2019.2959630>

- [15] H. Tong, T. Wang *et al.*, Mobility-Aware Seamless Handover With MPTCP in Software-Defined HetNets, *IEEE Transactions on Network and Service Management* 18 (1), (2021) pp. 498–510.

doi: <https://doi.org/10.1109/TNSM.2021.3050627>

- [16] M. Chen, T. Dreibholz *et al.*, Improvement and Implementation of a Multi-Path Management Algorithm based on MPTCP, in: Proceedings of the 2020 IEEE 45th Conference on Local Computer Networks: LCN 2020, 2020, pp.134–143.

doi: <https://doi.org/10.1109/LCN48667.2020.9314778>



This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution NonCommercial (CC BY-NC 4.0) license.