Validation of MPTCP performance enhancement algorithm in real PS-LTE Environment

B.G Lee, S.B Song, S.Ryu, J.Y Lee
Electrical and Electronic Engineering
Yonsei University
Seoul, Korea
lbg0207@naver.com, glistarttl@gmail.com, powerfulrs@gmail.com, jyl@yonsei.ac.kr

Abstract**—** Reliability is very important factor in disaster service than any others. Connection delays or interruptions in providing disaster services can lead to catastrophic consequences. However, in the event of a disaster, there may be a temporary or permanent transmission path break in data communication due to various reasons. Therefore, in order to improve the reliability and performance of the disaster service in the disaster communication network, our study is conducted to examine the Constraint-based Proactive Scheduling(CP-Scheduling) algorithm[1] and the feedback-based path failure detection(FPF) algorithm[2] which are applied for MPTCP in PS-LTE Environment.

Keywords—disaster; reliability; CP-Scheduling; FPF; PS-LTE; MPTCP;

I. Introduction

Disaster communications refers to all of the communications that contribute to the resolution of a situation using communication means and information and communication technologies in the event of a disaster. By 2017, there were about 600,000 devices for disaster communications. However, the most of devices are supporting only voice data. It can be seen that there are many differences from other services in which multimedia services are provided. As the voice data is only provided, it cannot be possible to understand the disaster situation quickly. Therefore, there is a need for a multimedia service suitable for disaster service for identification and resolution of a rapid disaster situation. Disaster communication network research is underway at various research institutes for such multimedia services.

When a disaster occurs, the condition of the network differs from the daily life. Therefore, unlike original networks, disaster communication networks require additional new functions. An example is the multi-path capability for reliable transmission. Multipath functionality is the ability of a device to select links, or use them at the same time, over connectable multiple links rather than communicating over only one designated network. In the case of applying the network using the multipath, it is advantageous that reliable data transmission is possible because continuous data transmission & reception through other connected link is possible even if sudden specific link failure occurs in the disaster.

However, this is a conceptual result, and it is not possible to obtain an ideal result when applied to a real network. This is caused by the difference between the receive buffer problem and the multipath performance in the process of managing the multipath. Therefore, in this paper, CP-Scheduling algorithm [1] and FPF algorithm[2], which is one of the algorithms that can solve the above problem, will be applied to actual PS-LTE disaster network to examine performance.

The rest of this paper is organized as follows. Section 2 explains the background of the algorithm and how each algorithm works. Section 3 will examine the results of operating in PS-LTE. Finally, Section 4 will draw conclusions based on the above.

II. Background and Related Work

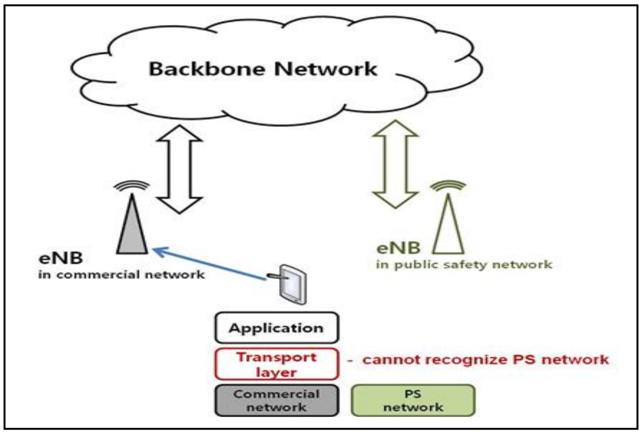

Fig. 1. Problem of Transmission Protocol

Today's devices can have multiple network interfaces, including servers. Device hosts with multiple network interfaces select one of the multiple interfaces to increase network reliability and to take advantage of the offload due to the use of additional networks. If not, using multiple networks can be a big problem from an application point of view. If the application wants to change the network interface for some reason, there is a drawback that the prior network connection must be disconnected and newly connected to the desired network. That is, the network to be newly connected also go through the previous connection & termination session.

Multi-path functionality is a way to overcome above drawback. However, most devices do not use it. This is because commonly used transport protocols do not support this multi-path functions. Currently, protocols used mainly in

978-1-5386-4646-5/18/$31.00 ©2018 IEEE

transport layer are transmission control protocol (TCP) and user datagram protocol (UDP). However, as mentioned above, these protocols do not support multiple networks. That is, although the device recognizes both the commercial communication network and the PS-network interface as shown in Fig. 1, the transport layer protocol (TCP or UDP) transmits data through only one interface. To use multi-path function, we can make it by changing application layer. First, we need to detect multiple interfaces in the application layer and pass the appropriate action to the lower layer accordingly. In practice, however, this approach can be a big problem in compatibility because it runs differently depending on the service, and because the application itself can become heavy, it is not suitable and is rarely used. Therefore, it is necessary to develop a new protocol in order to perform the multipath function without problem. Therefore, the Internet Engineering Task Force standardized Multipath TCP (MPTCP), an extension of TCP. The MPTCP is a protocol that allows data transmission by using two or more network interfaces at the same time, rather than using only one of the available network interfaces as a network interface. This protocol is compatible with the original TCP (single TCP) protocol, so it can be used without giving up the prior method. That is, it can be used without any additional device change, thereby improving the reliability.

However, the first thing to consider when using MPTCP instead of TCP is that sub-flows can interfere with each other. In principle, each sub-flow performs network task independently according to each network conditions. Nonetheless, each sub-flows are affecting different sub-flows. This is because the application manages the data sequence number (DNS) that the protocol uses to be seen as the original TCP and uses only one receive buffer. Therefore, as the size of the receiver buffer decreases, overflow of data occurs, which causes a huge throughput degradation. Also, since they cannot operate independently of each other, they cannot cope with sudden failure, and normal sub-flow has a problem in that it is necessary to wait for data transmission of abnormal sub-flow to be completed.

For example, in the event of a sudden disaster, it is expected that data transmission will fail frequently due to temporary disconnection of the transmission path because of destruction of buildings, or semi-permanent disconnection of the transmission path because of destruction of the base station. In the event of a temporary or semi- permanent disruption of some of the MPTCP paths in such a disaster situation, the MPTCP will not know this and will continue to try to send meaningless data through the disconnected path. In addition, this behavior can also cause transmission disturbances on stable paths which have no interruptions. That is, all stable paths are interrupted due to the meaningless transmission of the disconnected path.

In addition, MPTCP may cause problems depending on various transmission environments of each sub-flow. For example, packets sent on the fast sub-flow will be sent later but will accumulate in the receiver's buffer with out-of-order packets due to the difference in transmission time between paths. This drains the receiver buffer (RWND). As a result, an overflow occurs in the buffer, which leads to a transmission interruption phenomenon. Of course, this phenomenon does not occur if the size of the receiver buffer is sufficient. However, since the concept of sufficiency is considerably large, it is necessary to limit the size of the buffer in order to operate it, and therefore, phenomena occur frequently. Therefore, we choose CP-Scheduling algorithm[1] and FPF algorithm[2] to solve the above two problems.

A. *Constarint-based Proactive Scheduling*

This algorithm first grasps the performance of the path based on the Round Trip Time. It should then determine whether the bad sub-flow is interfering. Therefore, if the out-of-order data value due to the difference between the RTT value of the best performance sub-flow and the RTT value of the suspected sub-flow is larger than the required buffer size of the suspected sub-flow, then it does not use the that suspected sub-flow.

B. *Feedback-based Path Failure Detection Algorithm*

If the sender or receiver suspects that the path has been disconnected, the algorithm transmits an ACK according to a newly defined format containing information about the suspected path through the stable path. After exchanging feedback information for path failure detection, the sender determines whether a suspected path has occurred or not based on the information in the ACK of the new format. To accomplish this, the algorithm requires two core options. First of all, what should the recipient suspect and what to send the suspicious contents to the sender, and whether the sender should cut off the path. First, the receiver computes an estimate of the transmitted data based on the number of the out-of-order received packet. It then compares it to the WINDOW size of the recipient and analyzes the received packets for each sub-flow if the estimate is larger, and suspects the sub-flow in which there is no out-of-sequence packets. If the sender does not receive an ACK or the sender WINDOW is exhausted, the suspicion will start.

III. Demonstration of Algorithm

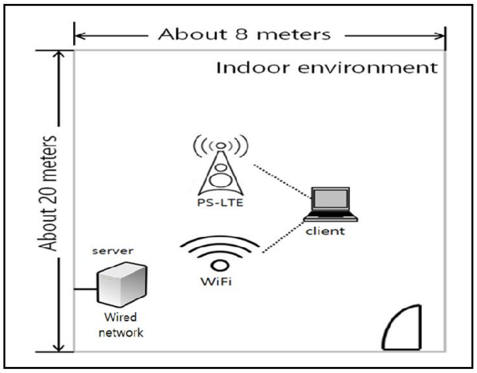

Fig. 2. Experiment Environment using PS-LTE

In this section, the algorithm is examined using PS-LTE provided by Korea Railroad Research Institute. The examination environment is shown in Fig. 2 and the performance of each sub-flow link is shown in Table 1.

TABLE I.

Wireless Link	Average RTT(ms)	Average throughput(Mbyte/s)
PS-LTE(1)	24.16	18.03
WiFi(2)	6.65	64.72

[a] Link performance per sub-flow

A. Constarint-based Proactive Scheduling

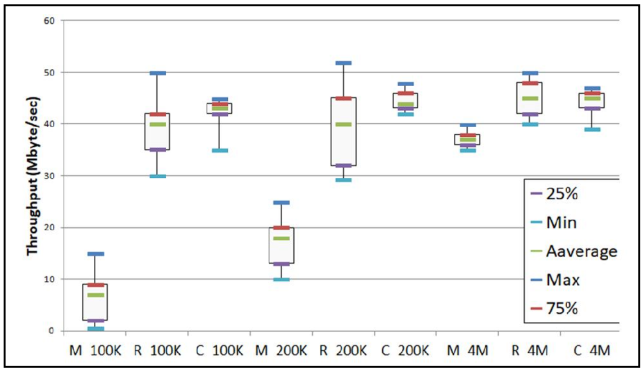

Fig. 3. Throuput Compare Original MPTCP and CP-Scheduling MPTCP

For this experiment, we used the method of varying the size of the receiving buffer and obtaining throughput. In Fig. 3, M denotes Original MPTCP, R denotes Radical CP-scheduling, and C denotes conservative CP-scheduling. And X-axis means buffer size. When the size of the buffer is sufficiently large by 4M, the performance difference between original MPTCP and CP-Scheduling applied MPTCP is not remarkable. However, as the size of the receive buffer is decreased, the throughput difference become increases. When the size of the buffer is reduced by 200K, the data value of the buffer increases due to the delay difference. At this time, the size of the buffer is limited, which leads to the degradation of the throughput. However, when the algorithm is applied, it can be seen that this delay difference is recognized and the use of the low-performance link is restricted to prevent the degradation of throughput.

B. Feedback-based Path Failure Detection Algorithm

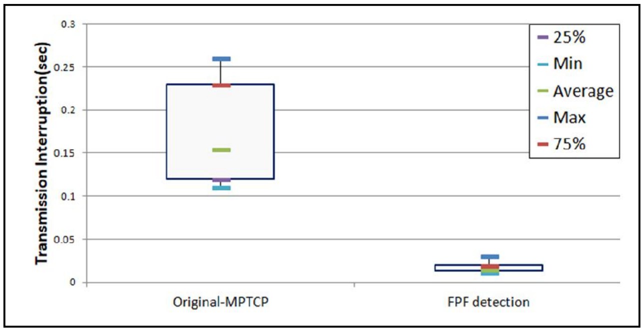

Fig. 4. Transmission Interruption compare Orginal MPTCP and FPF MPTCP

In this experiment, we fixed the size of the buffer to 300KB and compared the transmission interruption time for this experiment. So we can compare only transmission interruption time without buffer blocking. Then, Sub-flow2 (WiFi) is disconnected after 5 seconds of transmission. The sender then sends the packet until it has exhausted the entire sender window. As shown in Fig. 4, it can be seen that the average transmission interruption time is 0.35 seconds for original MPTCP and 0.01 seconds for FPF applied MPTCP. In other words, it can be seen that the MPTCP using the proposed algorithm resumes transmission faster than the original MPTCP. This is because the sub-flow 1 is interrupted by path failure of sub-flow 2. The original MPTCP cannot resume sub-flow 1's transmission until retransmission timeout occurs, however the FPF applied MPTCP can transmit through the stable sub-flow 1 without sending the data to the corresponding path, so that the transmission interruption time is remarkably reduced.

IV. CONCOLUSION

We have examined the CP scheduling[1] and FPF algorithm[2] in the PS-LTE disaster network to verify that it can be used in a disaster situation. We can confirm that the link failure can be detected by the FPF algorithm, and it is confirmed that the CP-scheduling can prevent the throughput degradation due to buffer blocking which is caused by performance difference of the links. Therefore, it is expected that it will be possible to provide stable multimedia service in a disaster situation. We will also study the stability of the link by learning the RTT value and the size of the receive buffer after introducing AI, the hottest field currently.

Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (NRF2017R1A2B4008503) and MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2017-2017-0-01633) supervised by the IITP(Institute for Information & communications Technology Promotion)

References

[1] B. H. Oh and J. Lee, "Constraint-based proactive scheduling for MPTCP in wireless networks", Computer Networks, vol 91, pp. 548-563 NOVEMBER 2015.

[2] B. H. Oh and J. Lee, "Feedback-Based Path Failure Detection and Buffer Blocking Protection for MPTCP," in IEEE/ACM Transactions on Networking, vol. 24, no. 6, pp. 3450-3461, December 2016. doi: 10.1109/TNET.2016.2527759.

[3] T. A. Le, R. Haw, C. S. Hong, and S. Lee, “A multipath cubic TCP congestion control with multipath fast recovery over high bandwidthdelay product networks,” IEICE Trans. Commun., vol. E95-B, no. 7, pp. 2232–2244, Jul. 2012.

[4] A. Ford, C. Raiciu, M. Handley, and O. Bonaventure, “TCP Extensions for Multipath Operation With Multiple Addresses”, document RFC 6824, Jan. 2013