A Study on Transmission-Control Middleware on an Android Terminal in a WLAN Environment

Hiromi Hirai, Kaori Miki, Saneyasu Yamaguchi, Masato Oguchi*

Department of Information Science, Ochanomizu University
2-1-1 Ohtsuka, Bunkyo-ku, Tokyo 112-8610 Japan
E-mail: hiromi@ogl.is.ocha.ac.jp

In recent years, the digital convergence era has arrived. Computers are required to be convenient anywhere and anytime. The huge demand for portable computers produced the smartphone, a mobile computer. However, the architecture of the smartphone is different from that of the general-purpose PC because of the poor function of the I/O interface and the limited hardware of the mobile phone. To overcome this disadvantage, the smartphone is constantly connected to the Internet and obtains information through the Cloud. The proliferation of SNS services has also stimulated users to publish information with their smartphones. For instance, users can upload their videos to Youtube and synchronize their calendars or address books in the Cloud. Thus, the performance of the smartphone is measured not only by its processing power but also by its network availability. We focus on the networking system of the smartphone to improve upon it.

We adopt Android as the OS to be researched because it has the highest share in the international smartphone market. Moreover, Android's source code is open, and developers can freely customize or remodel the code[1].

The smartphone is a useful device that enables users to connect to the Internet, regardless of time and location. If a user uses a smartphone while traveling, the device connects the user from access point to access point. However, if a large number of smartphones connect to an access point at the same time, congestion will frequently occur.

In this study, we attempt to develop a middleware through which each internet device can share its own Congestion Window (CWND) to predict network traffic[2]. To date, a collection of various TCPs have been developed. The previous TCPs predicted the traffic via packet loss or delay. All of them are designed for general-purpose PC to control the data flow independently. Android is one of OSs developed for mobile device. It is adopted TCP-cubic[3] and the highest value is limited so as not to interrupt own working system. TCP-cubic has loss-based congestion control, with which CWND is decreased drastically by a packet loss. In WLAN environment, packet loss does not always indicate congestion because of noise[4]. Actually packet loss rate in WLAN is much higher than that in wired LAN. Nowadays, the technology of cloud is also growing as rapidly as that of smartphone. Smartphone often depends on cloud for processing. In this study, the case is assumed in which smartphone transmits huge packets to cloud server as shown Figure 1. In recent years, smartphone tends to enjoy mass rapid communications. Cloud service supplies reliable WAN to be accessed from any part of the world. Therefore, most of the packet loss takes place between smartphone and an access point. Moreover the value of CWND is important because RTT is expected high.

Figure 1. Model of smartphones connected to cloud

If only one Android terminal monopolizes an access point or base station without any obstacles, the terminal is supposed to maintain the highest communication throughput. Nevertheless, the terminals are scrambling against each other for bandwidth while a large number of nodes usually connect to an access point. In this case, the transport layer plays a critical role because it controls the data flow. CWND also indirectly influences the communication throughput in an Android terminal, as demonstrated by [5].

Additionally, in a crowded situation, CWND becomes insecure and is excessively restricted. CWND, which controls the amount of data segments to flow, is expected to increase as much as possible. However, a huge packet from a terminal whose CWND is excessively fixed may disturb the whole network and even interrupt the transmission with frequent packet losses and retransmissions. Therefore we developed original TCP[6] which is designed to adjust the sender's CWND high and transmit packets aggressively. It is argued that aggressive TCP might interrupt the others' transmission. However, it should be allowed for an smartphone to use it when no other devices are communicating.

Thus, TCP switching is important for each terminal because the terminals need to be able to adapt to their environments. Because the conventional transport layer for mobile terminals is designed to behave modestly to avoid jamming the network traffic, transmissions by mobile terminals have more latency. However, in recent years, mobile terminals have also come to enjoy rapid communication on a massive scale (i.e., an adequate mechanism is required to control mobile communication.) Therefore, we propose a transmission-control middleware to flexibly control TCP. Figure 2 is the result of experiment.

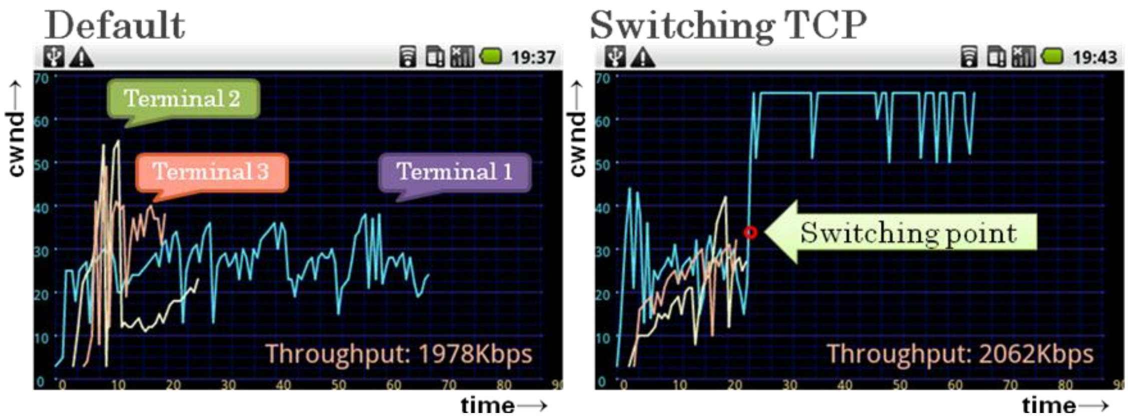

Figure 2. The behavior of congestion control

References

1. android developers: http://developer.android.com

2. W. Richard Stevens, TCP/IP Illustrated, Volume 1: The Protocols, Addison-Wesley, 1994.

3. S.Ha, I.Rhee and L.Xu, "CUBIC: A New TCP-Friendly High-Speed TCP Variant", SIGOPS Operating Systems Review, 42(5):64-74, 2008.

4. L.A. Grieco and S.Mascolo, "Performance Evaluation and Comparison of Westwood+, New Reno, and Vegas TCP Congestion Control", in Proc. Computer Communication Review, 34(2), pp. 25-38, 2004.5. G. H. Fredrickson, *Annu. Rev. Mater. Sci*. **1996**, *26*, 501.

5. H.Hirai, K.Miki, S.Yamaguchi, and M.Oguchi, "A Visualization Tool for Communication Performance of Android Terminal", The 73rd National Convention of IPSJ, 5V-9, March 2011.

6. K.Miki, S.Yamaguchi, and M.Oguchi, "Kernel Monitor of Transport Layer Developed for AndroidWorking on Mobile Phone Terminals", in Proc. the Tenth International Conference on Networks (ICN2011), St. Seattle, U.S.A. April 2010.