How to Deploy Timer for Retransmission

Jiamin Cheng, Lei Chen(✉), Ping Cui, Yuan An, and Kailiang Zhang

Jiangsu Province Key Laboratory of Intelligent Industry Control Technology, Xuzhou University of Technology, Xuzhou 221018, China
chenlei@xzit.edu.cn

Abstract. In some emerging mobile applications, the UDP has advantage for service latency. However, retransmission mechanism is essential to deal with the losing packets while the service is based on UDP protocol. The timer for retransmission could be deployed on sender side or receiver side. To compare these two approaches, we investigate the relationship between service latency and losing packet rate in simulation. The results show that deploying the timer on receiver side obtains higher performance than the traditional method does within unstable channel environment. A new retransmission algorithm is proposed to emerging mobile applications.

Keywords: Retransmission · Timer · Mobile computing · IOT · UDP · TCP

1 Introduction

A large number of mobile applications are emerging as 5G networks are being deployed. To satisfy the traffic demand, some new approaches are proposed to improve network routing and measurement [1–3]. Based on user behavior [4–6] and traffic flow analysis methods [7, 8], we can improve the transmission process. Some new scheduling strategies are proposed to improve user scheduling [9–12], to raise spectral resources utilization [13–16], save energy [17–19], guarantee quality of service [20–23]. Based on end side network traffic analysis [24–26], we can manage and improve the end user experience [27–30]. Through traffic reconstruction [31–33], we can characterize the traffic flow and the transport protocol can be redesigned [34–36]. However, the high latency of edge wireless networks deteriorates the user experience of mobile applications. In some cloud based services, especially vehicular service, the TCP protocol causes extra delay [37].

For reliable transportation, the timer is very important [38, 39]. In the network protocol layer, the commonly used transport layer protocols are transmission control protocol (TCP) and user datagram protocol (UDP). TCP provides the high reliability. UDP takes lower delay. TCP is a connection-oriented transport layer protocol, which requires the establishment of a connection by “three handshakes” before carrying out transmission work. It is also a kind of packet loss detection and message order guarantee. It can guarantee the correctness and correct order of the data, but the relative transmission speed is slow. UDP is a kind of connection-oriented transport layer protocol, that is, the sending and receiving parties can communicate without establishing a connection, and

H. Song and D. Jiang (Eds.): SIMUtools 2020, LNICST 370, pp. 256–265, 2021.
https://doi.org/10.1007/978-3-030-72795-6_20

provides transaction-oriented simple and unreliable information delivery services. The transmission speed of UDP is fast and the system overhead is small.

To satisfy the high reliability and low latency demand, we need to combine transmission and UDP protocols. TCP is a connection-oriented service, resulting in the disadvantages of slow transmission rate, low efficiency, and high overhead and easy to attack. UDP is a simple datagram-oriented, unconnected transport layer protocol. Without TCP's handshake, confirmation, Windows, retransmission, congestion control and other mechanisms, user data transmission rate has been improved to some extent. UDP is more efficient than TCP and is suitable for communications or broadcast communications that require high speed transmission and real-time. The internet of vehicles requires the transmission of a large amount of voice information and road information. The speed of data feedback is required to be high. If the speed is relatively slow, it cannot guarantee the access to real-time road information, resulting in inaccurate information acquisition and user experience will decline. A lot of research shows that the UDP protocol is superior to TCP protocol in networks performance. Since the unstable channel of mobile communication, however, UDP needs an additional retransmission mechanism.

As a common approach, some applications adopt UDP in network layer, and take retransmission mechanism in application layer. In this paper, we reconsider the retransmission mechanism. We compare the approach of deploying retransmission timer on sender side and the approach of deploying retransmission timer on receiver side. In [40], a timer is deployed in client to accelerate the retransmission. We further let the receiver totally trigger the retransmission.

This paper is divided into four sections. In the second section, we describe the two approaches and give the algorithms. In the third section, we show and analyze the simulation results. We conclude in the fourth section.

2 Alternative Retransmission Mechanism and Algorithm

2.1 Deploy Timer on Sender Side

Like TPC, we can set timer on sender side. The retransmission mechanism is illustrated in Fig. 1.

1) The sender sends packets to receiver, and start timer.
2) If receiver receives packets, it will send acknowledgement packet to sender to confirm the received packets.
3) When timer is timeout, the sender will check ACK message and resend unconfirmed packets. The timer is restarted. The loss of ACK message and data packets will cause resending.
4) Loop the process until all packets are confirmed by ACK message.

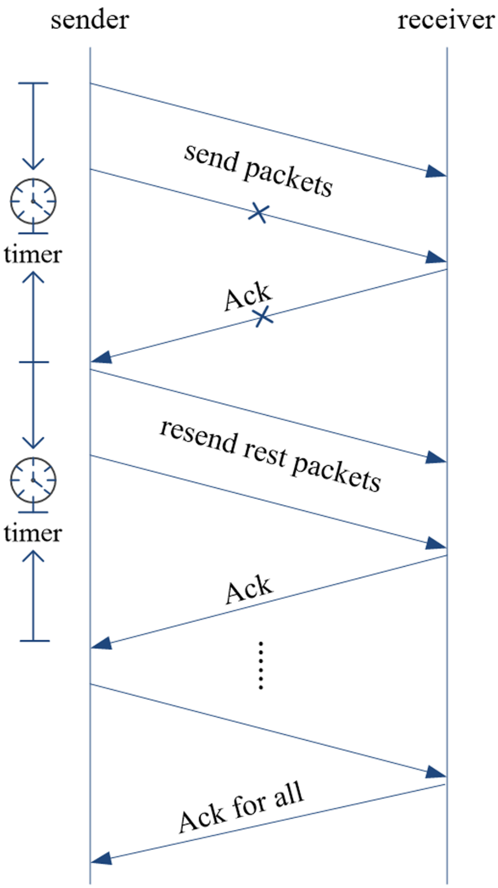

Fig. 1. Timer on sender.

2.2 Deploy Timer on Receiver Side

Unlike TPC, we deploy timer in receiver. The sender firstly tells receiver the number of packets that will be transmitted. The receiver requests for these packet and gives the sequence number it has not received. The timer is deployed on receiver side in this approach. If timeout, the receiver will send a new request message, namely NACK. The retransmission mechanism is illustrated in Fig. 2.

1) The receiver sends request message to sender, and start timer.
2) If receiver receives packets, it will record the sequence numbers of the packets.
3) When timer is timeout, the receiver will check received packets. If it has not get all wanted packets, it will send a new request message to sender and restart the timer.
4) Loop the process until all packets are received by receiver.

In this model, the request message can also be called NACK message which includes packet numbers the receiver want and has not received. The second model can reduce duplicate packets caused by ACK message losing in crowed channel.

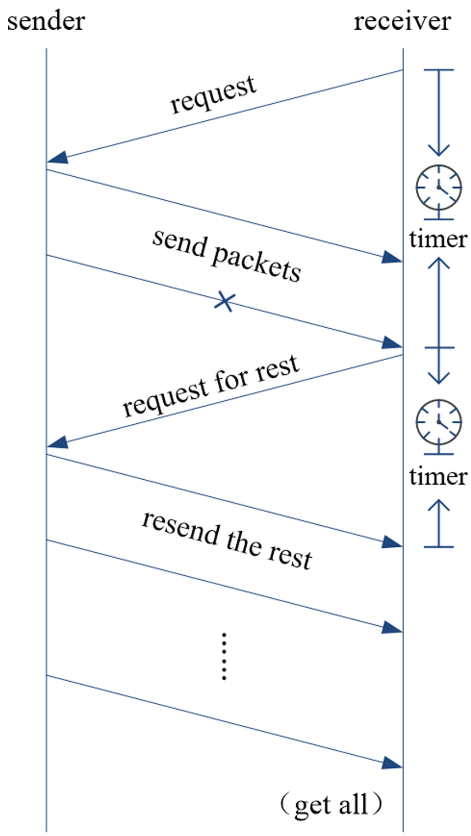

Fig. 2. Timer on receiver.

3 Simulation and Analysis

In the simulation, we arbitrarily set the transport period is 1 time slot, and set that the machine would cost 0.01 time slot to transmit one packet to channel. The timeout period is 3 round trip periods, namely 6 transport periods.

The simulation algorithm of ACK model is listed in Fig. 3.

```
Step1    While (NumPackets not null)
Step2      Record the number of packets sent
Step3        Time plus 0.01 * number of packets
Step4        simulation returns the number of packets successfully sent
Step5        Time plus packet transmission time
Step6        If (all packages are not lost)
Step7            If (lose rate simulation feedback whether the ACK was
                    successfully transmitted)
Step8                Time record feedback time
Step9                Record lost packets, just resend lost packets next time
Step10           else
Step11               Time record wait 3 times roundtrip time time
Step12           End If
Step13       Else
Step14           Time record all retransmission time
Step15       End If
Step16       Update the lose rate based on the number of users
Step17   End while
```

```
Step1    While (NumPackets not null)
Step2      Record the number of packets sent
Step3        Time plus 0.01 * number of packets
Step4        simulation returns the number of packets successfully sent
Step5        Time plus packet transmission time
Step6        If(get the feedback)
Step7            According to the feedback to get the lost packages
Step8            Resend the lost packages and time record wait 3 times
         roundtrip time
Step9        Else
Step10           Resend all the packages and time record wait 3 times
         roundtrip time
Step11       End If
Step12       Update the lose rate based on the number of users
Step13   End while
```

Fig. 3. Timer on sender.

The simulation algorithm of NACK model is listed in Fig. 4.

```
Step1    Do
Step2      Record the number of packets sent
Step3      Time plus 0.01 * number of packets
Step4      Record the number of lost packets
Step5      If (Packets are missing)
Step6        Record the Time to wait for feedback packet
Step7        While (feedback NACK is not received)
Step8            Record resend NACK time
Step9        End while
Step10       Record the wait time after the last successful NACK
Step11     End If
Step12     Update the lose rate based on the number of users
Step13   End while(all packets are received)
```

```
Step1    Do
Step2      Record the number of packets sent
Step3      Time plus 0.01 * number of packets
Step4      Record the number of lost packets
Step5      If (Packets are missing)
Step6        Record the Time to wait for feedback packet
Step7        While (feedback NACK is not received)
Step8            Resend the NACK and time 3 times roundtrip time plus
         sending NACK's time
Step9        End while
Step10       Record the wait time after the last successful NACK
Step11     End If
Step12     Update the lose rate based on the number of users
Step13   End while(all packets are received)
```

Fig. 4. Timer on receiver.

The simulation results are showed in Fig. 5 and Fig. 6.

Because the lose rate would increase in crowed channel, the NACK model that reduces non-effective duplicate retransmission obtain more stable performance.

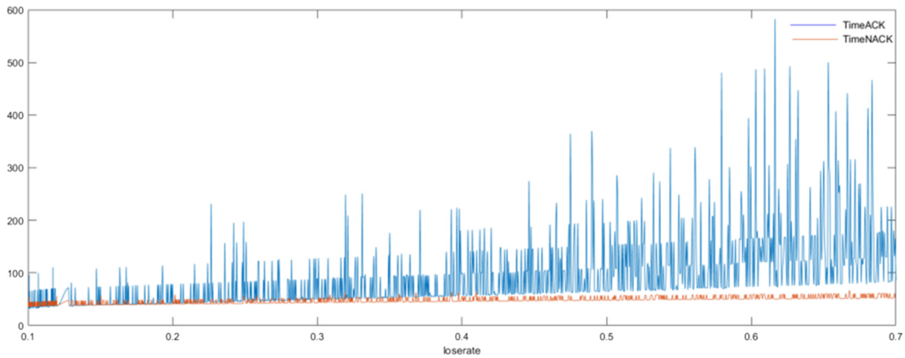

Fig. 5. Cost time for send 100 packets. (Horizontal axis is initial lose rate, the lose rate would update in the process; vertical axis is cost time).

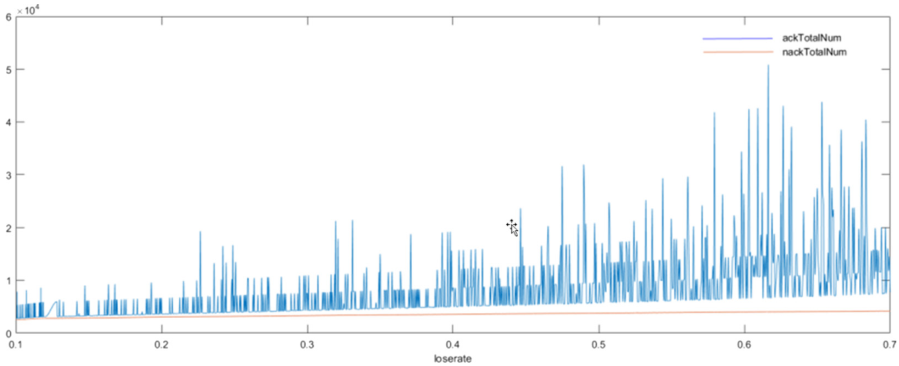

Fig. 6. Amount of sending for send 100 packets. (Horizontal axis is initial lose rate, the lose rate would update in the process; vertical axis is cost time).

4 Conclusion

In this paper, we combine the design principles and advantages of TCP and UDP protocols to design the UDP based retransmission mechanism. The timer is deployed in receiver; the retransmission process is triggered by receiver. We compare two retransmission models. The first approach sets timer on sender side like TCP, and the second one sets timer on receiver side. The simulation results show that the second model could reduce duplicate retransmission and obtain a stable performance.

Acknowledgements. This work is partly supported by Jiangsu technology project of Housing and Urban-Rural Development (No.2018ZD265) and Jiangsu major natural science research project of College and University (No. 19KJA470002).

References

1. Zhang, K., Chen, L., An, Y., Cui, P.: A QoE test system for vehicular voice cloud services. Mob. Netw. Appl. (2019). https://doi.org/10.1007/s11036-019-01415-3
2. Wang, F., Jiang, D., Qi, S.: An adaptive routing algorithm for integrated information networks. China Commun. **7**(1), 196–207 (2019)
3. Huo, L., Jiang, D., Lv, Z., Singh, S.: An intelligent optimization-based traffic information acquirement approach to software-defined networking. Comput. Intell. **36**(1), 151–171 (2019)
4. Chen, L., Jiang, D., Bao, R., Xiong, J., Liu, F., Bei, L.: MIMO scheduling effectiveness analysis for bursty data service from view of QoE. Chin. J. Electron. **26**(5), 1079–1085 (2017)
5. Jiang, D., Wang, Y., Lv, Z., et al.: Big data analysis-based network behavior insight of cellular networks for industry 4.0 applications. IEEE Trans. Ind. Inform. **16**(2), 1310–1320 (2020)
6. Bao, R., Chen, L., Cui, P.: User behavior and user experience analysis for social network services. Wireless Netw. (2019). https://doi.org/10.1007/s11276-019-02233-x,(
7. Jiang, D., Huo, L., Song, H.: Rethinking behaviors and activities of base stations in mobile cellular networks based on big data analysis. IEEE Trans. Netw. Sci. Eng. **1**(1), 1–12 (2018)
8. Chen, L., et al.: A lightweight end-side user experience data collection system for quality evaluation of multimedia communications. IEEE Access **6**(1), 15408–15419 (2018)
9. Chen, L., Zhang, L.: Spectral efficiency analysis for massive MIMO system under QoS constraint: an effective capacity perspective. Mob. Netw. Appl. (2020). https://doi.org/10.1007/s11036-019-01414-4
10. Wang, F., Jiang, D., Qi, S., Qiao, C., Shi, L.: A dynamic resource scheduling scheme in edge computing satellite networks. Mob. Netw. Appl. (2020). https://doi.org/10.1007/s11036-019-01421-5
11. Jiang, D., Huo, L., Lv, Z., et al.: A joint multi-criteria utility-based network selection approach for vehicle-to-infrastructure networking. IEEE Trans. Intell. Transp. Syst. **19**(10), 3305–3319 (2018)
12. Jiang, D., Zhang, P., Lv, Z., et al.: Energy-efficient multi-constraint routing algorithm with load balancing for smart city applications. IEEE Internet Things J. **3**(6), 1437–1447 (2016)
13. Lee, Y., Kim, Y., Park, S.: A machine learning approach that meets axiomatic properties in probabilistic analysis of LTE spectral efficiency. In: 2019 International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Korea (South), pp. 1451–1453 (2019)
14. Ji, H., Sun, C., Shieh, W.: Spectral efficiency comparison between analog and digital RoF for mobile fronthaul transmission link. J. Lightwave Technol. **38**(20), 5617–5623 (2020)
15. Hayati, M., Kalbkhani, H., Shayesteh, M.G.: Relay selection for spectral-efficient network-coded multi-source D2D communications. In: 2019 27th Iranian Conference on Electrical Engineering (ICEE), Yazd, Iran, pp. 1377–1381 (2019)
16. You, L., Xiong, J., Zappone, A., Wang, W., Gao, X.: Spectral efficiency and energy efficiency tradeoff in massive MIMO downlink transmission with statistical CSIT. IEEE Trans. Signal Process. **68**, 2645–2659 (2020)
17. Jiang, D., Li, W., Lv, H.: An energy-efficient cooperative multicast routing in multi-hop wireless networks for smart medical applications. Neurocomputing **220**(2017), 160–169 (2017)
18. Jiang, D., Wang, Y., Lv, Z., et al.: Intelligent optimization-based reliable energy-efficient networking in cloud services for IIoT networks. IEEE J. Sel. Areas Commun. **38**(5), 928–941 (2019)
19. Wiatr, P., Chen, J., Monti, P., Wosinska, L.: Energy efficiency versus reliability performance in optical backbone networks [invited]. IEEE/OSA J. Opt. Commun. Netw. **7**(3), A482–A491 (2015)

20. Tan, J., Xiao, S., Han, S., Liang, Y., Leung, V.C.M.: QoS-aware user association and resource allocation in LAA-LTE/WiFi coexistence systems. IEEE Trans. Wireless Commun. **18**(4), 2415–2430 (2019)
21. Wang, Y., Tang, X., Wang, T.: A unified QoS and security provisioning framework for wiretap cognitive radio networks: a statistical queueing analysis approach. IEEE Trans. Wireless Commun. **18**(3), 1548–1565 (2019)
22. Hassan, M.Z., Hossain, M.J., Cheng, J., Leung, V.C.M.: Hybrid RF/FSO backhaul networks with statistical-QoS-aware buffer-aided relaying. IEEE Trans. Wireless Commun. **19**(3), 1464–1483 (2020)
23. Zhang, Z., Wang, R., Yu, F.R., Fu, F., Yan, Q.: QoS Aware transcoding for live streaming in edge-clouds aided HetNets: an enhanced actor-critic approach. IEEE Trans. Veh. Technol. **68**(11), 11295–11308 (2019)
24. Jiang, D., Wang, W., Shi, L., et al.: A compressive sensing-based approach to end-to-end network traffic reconstruction. IEEE Trans. Netw. Sci. Eng. **5**(3), 1–2 (2018)
25. Qi, S., Jiang, D., Huo, L.: A prediction approach to end-to-end traffic in space information networks. Mob. Netw. Appl. (2019). https://doi.org/10.1007/s11036-019-01424-2
26. Jiang, D., Huo, L., Li, Y.: Fine-granularity inference and estimations to network traffic for SDN. PLoS ONE **13**(5), 1–23 (2018)
27. Barakabitze, A.A., et al.: QoE management of multimedia streaming services in future networks: a tutorial and survey. IEEE Commun. Surv. Tutor. **22**(1), 526–565 (2020)
28. Orsolic, I., Skorin-Kapov, L.: A framework for in-network QoE monitoring of encrypted video streaming. IEEE Access **8**, 74691–74706 (2020)
29. Song, E., et al.: Threshold-oblivious on-line web QoE assessment using neural network-based regression model. IET Commun. **14**(12), 2018–2026 (2020)
30. Seufert, M., Wassermann, S., Casas, P.: Considering user behavior in the quality of experience cycle: towards proactive QoE-aware traffic management. IEEE Commun. Lett. **23**(7), 1145–1148 (2019)
31. Wang, Y., Jiang, D., Huo, L., Zhao, Y.: A new traffic prediction algorithm to software defined networking. Mob. Netw. Appl. (2019). https://doi.org/10.1007/s11036-019-01423-3
32. Huo, L., Jiang, D., Qi, S., Song, H., Miao, L.: An AI-based adaptive cognitive modeling and measurement method of network traffic for EIS. Mob. Netw. Appl. (2019). https://doi.org/10.1007/s11036-019-01419-z
33. Huo, L., Jiang, D., Zhu, X., et al.: An SDN-based fine-grained measurement and modeling approach to vehicular communication network traffic. Int. J. Commun. Syst. 1–12 (2019)
34. Polese, M., Chiariotti, F., Bonetto, E., Rigotto, F., Zanella, A., Zorzi, M.: A survey on recent advances in transport layer protocols. IEEE Commun. Surv. Tutor. **21**(4), 3584–3608 (2019)
35. Morelli, A., Provosty, M., Fronteddu, R., Suri, N.: Performance evaluation of transport protocols in tactical network environments. In: MILCOM 2019 - 2019 IEEE Military Communications Conference (MILCOM), Norfolk, VA, USA, pp. 30–36 (2019)
36. Dwivedi, U., Agarwal, A., Roy, A., Chakraborty, A., Mukherjee, A.: A survey on wireless sensor network protocols for transport layer. In: 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, pp. 390–394 (2018)
37. Assefi, M., Wittie, M., Knight, A.: Impact of network performance on cloud speech recognition. In: International Conference on Computer Communication and Networks, pp. 1–6. IEEE (2015)
38. Gomez, C., Crowcroft, J.: Multimodal retransmission timer for LPWAN. IEEE Internet Things J. **7**(6), 4827–4838 (2020)

39. Albashear, A.M.M., Ali, H.A., Ali, A.M.: Detection of man-in-the-middle attacks by using the TCP retransmission timeout: key compromise impersonation attack as study case. In: 2018 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE), Khartoum, pp. 1–8 (2018)
40. Kishimoto, S., Osada, S., Tarutani, Y., Fukushima, Y., Yokohira, T.: A TCP Incast avoidance method based on retransmission requests from a client. In: 2019 International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Korea (South), pp. 153–158 (2019)