Performance Enhancement of Multipath TCP in Mobile Ad Hoc Networks

Tongguang Zhang*, Shuai Zhao, Bingfei Ren, Yulong Shi, Bo Cheng, *Member, IEEE*, Junliang Chen
State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, P. R. China
{ztguang, zhaoshuaiby, rbf, shiyulong2015, chengbo, chjl}@bupt.edu.cn

***Abstract*—In some special circumstances, e.g. tsunamis, floods, battlefields, earthquakes, etc., communication infrastructures are damaged or non-existent, as well as unmanned aerial vehicle (UAV) cluster. For the communication between people or UAVs, UAVs or mobile smart devices (MSDs) can be used to construct Mobile Ad Hoc Networks (MANETs), and Multipath TCP (MPTCP) can be used to simultaneously transmit in one TCP connection via multiple interfaces of MSDs. However the original MPTCP subpaths creating algorithm can establish multiple subpaths between two adjacent nodes, thus cannot achieve true concurrent data transmission. To solve this issue, we research and improve both the algorithm of adding routing table entries and the algorithm of establishing subpaths to offer more efficient use of multiple subpaths and better network traffic load balancing. The main works are as follows: (1) improve multi-hop routing protocol; (2) run MPTCP on UAVs or MSDs; (3) improve MPTCP subpaths establishment algorithm. The results show that our algorithms have better performance than the original MPTCP in achieving higher data throughput.**

I. Introduction

Natural disasters, e.g. earthquakes, floods, tsunamis, etc., often cause breakdown or interruption of communication infrastructures. To improve the efficiency of search-and-rescue, it is necessary to provide communications among people or MSDs. Smartphones are more powerful and can be used to construct MANETs. In addition, UAV cluster is a research hotspot and can be also used to carry out reconnaissance and surveillance. However, due to MSDs move in and out of network coverage areas and the topology is constantly changing, there are the issues exhibited by MANETs, that is, frequent disconnections and failed transmissions. Hence, we use Multipath TCP (MPTCP) [1][2] to solve the issues aiming to offer more efficient use of multiple subpaths and better network traffic load balancing. The first contribution of this paper is to improve multi-hop routing protocol (MHRP) that adds routing table entries according to the number of next hops and network interface number. The second is to port the Linux Kernel MultiPath TCP (version 0.92) [3] to Android. The third contribution is to improve the establishment algorithm for MPTCP subpaths. To our best knowledge, it is the first time to simultaneously research and improve both the algorithm of adding routing table entries and the MPTCP subpaths creating algorithm.

II. Implementation of MPTCP in MANETs

Fig. 1 shows a scenario of MPTCP in MANETs (MiM). We assume each mobile node (M1-M5) is equipped with four network interfaces (e0,e1,e2,e3). IP addresses allocated for network interfaces belong to different network segments. In expression 1. Y denotes the mobile nodes (M1-M5), X denotes the interfaces (e0,e1,e2,e3) in nodes (M1-M5).

$$112.26.X.Y/24,\ X\in[0,3]\ \ Y\in[1,5] \tag{1}$$

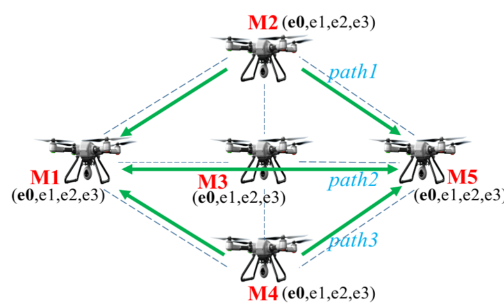

Fig. 1. The application scenario of MPTCP in MANETs.

A. *Improving Multi-hop Routing Protocol*

In this paper, OSPF MANET Designated Routers (MDR) is used as MHRP. In Fig. 1, MDR can only generate routing information for e0, however, MPTCP needs to build subpaths using the interfaces (e0-e3). Therefore, it is need to improve MHRP to support for the establishment of MPTCP subpaths, the improved method is as follow: generate routing table entries for e0, meanwhile, only one routing table entry is generated for each destination address (e1-e3) respectively. Thus, in M1, there are 6 routing entries to M5 generated by the improved MDR, as represented in equation 2 and 3, where E_0RE represents the routing entry for e0 in M1, E_XRE represents the routing entry for e1,e2,e3 in M1.

$$E_0RE = \text{nexthop via } 112.26.0.Y \text{ eth0 to } 112.26.0.5,\quad Y\in[2,4] \tag{2}$$

$$E_XRE = \text{nexthop via } 112.26.0.Y \text{ eth0 to } 112.26.X.5,\quad X\in[1,3]\ \ Y=X+1 \tag{3}$$

Due to space constraints, the detailed algorithm description is put in Algorithm-1.txt [4]. The test result is shown in Fig. 2, the left shows the routing table of M1, which includes 6 routing entries to M5, the right shows the routing table of M5.

* Tongguang Zhang is corresponding author.

This work was supported in part by the National Natural Science Foundation of China under Grant 61501048, China Postdoctoral Science Foundation funded project under Grant 2016-T90067, 2015M570060.

978-1-5090-6501-1/17/$31.00 ©2017 IEEE

B. MPTCP on Android

It is the key for the implementation of MiM to run MPTCP on UAVs or MSDs, in this paper, we assume the operating system on UAV or MSD is Android. We port MPTCP to Android-x86-nougat (Android 7.1.1, kernel 4.4.62) based on the MPTCP implementation in the Linux kernel (MP-TCP). The process of porting MPTCP to Android includes two steps: First, identify the files needed to be copied and modified by comparing the directory structures and the individual files between MP-TCP and Android kernel source tree; Second, verify the MPTCP function correctness in Android.

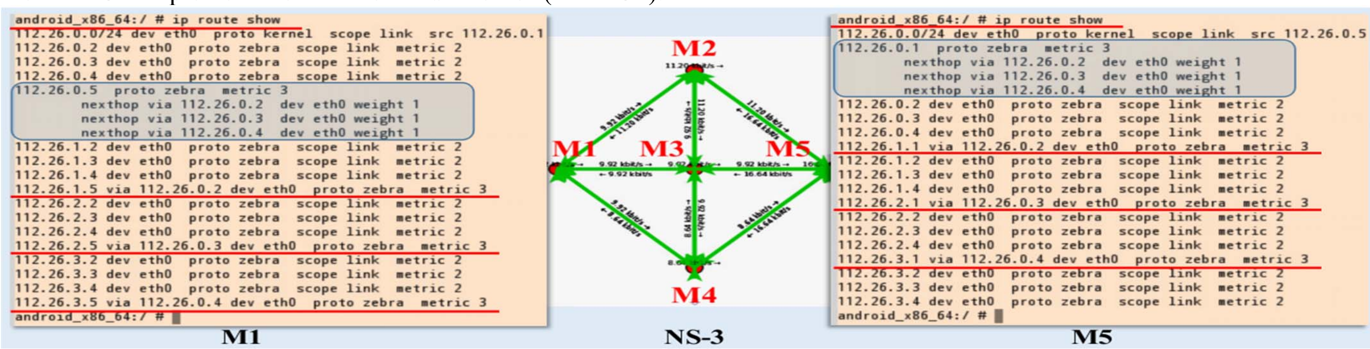

Fig. 2. The test result of improving multi-hop routing protocol.

C. Improving Subpaths Establishment Algorithm

We assume M1 acts as server, M5 acts as client, the original MPTCP uses M5:(e0,e1,e2,e3) and M1:(e0,e1,e2,e3) to establish subpaths, such as the MPTCP subpaths of M5 to M1, as represented in expression 4.

src: 112.26.X.5, dst: 112.26.Y.1, X,Y∈[0, 3] (4)

src: 112.26.X.5, dst: 112.26.X.1, X∈[0, 3] (5)

Why not be subpaths like expression 5? The root reason resides in the improvement to routing protocol in section II.A. For better network traffic load balancing, it is necessary to establish subpaths represented in expression 5. Hence, it is essential to improve the subpaths establishment algorithm.

Table. 1 shows the subpaths established by the improved MPTCP. NoNH is the number of next hops. NoNH=3 of master-subpath improves its connectivity reliability. The interfaces (M1-M5:e0) play three roles: 1. used by quagga to generate multi-hop routing entries; 2. used to establish master subpath; 3. forward packets as mid-nodes. The interfaces M1-M5:(e1,e2,e3) are used to establish slave subpaths.

Table. 1: The established subpaths.

subpath	src-IP	dst-IP	middle nodes	NoNH
master-subpath	M5:e0	M1:e0	M2, M3, M4	3
slave-subpath1	M5:e1	M1:e1	M2	1
slave-subpath2	M5:e2	M1:e2	M3	1
slave-subpath3	M5:e3	M1:e3	M4	1

III. Testing and Evaluation

The test environment is an IBM Server with 32 CPU-cores, 64GB mem, and Fedora 24 installed. We create five VirtualBox instances for Android, each of them has four network interfaces. Network topology is generated by NS-3 script [4]. We run command ***nc*** in M1 to make it as a file server, run command ***nc*** in M5 to download file *bash* from M1. The test results show that MPTCP is successfully running in MANETs. Note that the topology is simplified by letting NS-3 nodes remain quiescent without affecting the function test of MiM.

We also carry out 30 times experiments to compare data transfer performances of TCP-MDR (TCP with Multi-hop Routing), MPTCP-MDR (original MPTCP with Multi-hop Routing), Improved-MPTCP-MDR (improved MPTCP with Multi-hop Routing). As shown in Fig. 3, Improved-MPTCP-MDR is better than TCP-MDR, however, MPTCP-MDR is worse than TCP-MDR. The reason is that original MPTCP can not distinguish which subpaths belong to one pair of nodes and which subpaths belong to multiple pairs of nodes.

All modified source code files and MPTCP demo (mp4 file) can be got from [4].

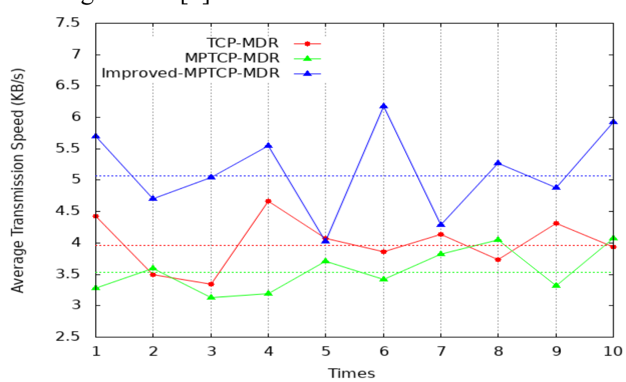

Fig. 3. The evaluation of improved MPTCP.

IV. Conclusion

In this paper, our fundamental contributions are to research and improve the algorithms of both multi-hop routing and establishing subpaths in MANETs, and to port MPTCP [3] to Android. The third contribution presented in section I is the main difference of our improved MPTCP and current MPTCP. The test results show that our algorithms can offer more efficient use of multiple subpaths and better network traffic load balancing.

References

[1] A. Ford, C. Raiciu, M. Handley, and O. Bonaventure, "TCP Extensions for Multipath Operation with Multiple Addresses," RFC 6824, 2013.

[2] MPTCP working group. [Online]. Available: https://datatracker.ietf.org/wg/mptcp/documents/

[3] C. Paasch, S. Barré, et al., "Multipath TCP in the Linux Kernel," [Online]. Available: http://www.multipath-tcp.org

[4] https://github.com/ztguang/MiM