BESS: BDP Estimation Based Slow Start Algorithm for MPTCP in mmWave-LTE Networks

Yanbing Liu, Xiaowei Qin, Ting Zhu, Xiaohui Chen, Guo Wei
Department of Electronic Engineering and Information Science,
University of Science and Technology of China, Hefei 230027, China
Email: viper@mail.ustc.edu.cn, {qinxw, zhuting, cxh, wei}@ustc.edu.cn

***Abstract*—In recent years, millimeter wave (mmWave) communications, considered as a compelling technology to achieve throughput improvement target of the fifth generation (5G) mobile networks, is quickly attracting attention in both scientific and industrial communities. To enhance reliability of mmWave link, mmWave-LTE networks with MPTCP could be used as an effective solution due to its path diversity. However, the poor performance of MPTCP is found in mmWave-LTE networks based on previous studies. In this paper, we identify the reason of the preceding problem which is the congestion window overgrowth in Slow Start phase, and propose a BDP Estimation Based Slow Start (BESS) algorithm for MPTCP to address this problem. Our simulations show that with the control of BESS, MPTCP can sufficiently use the bandwidth of mmWave networks, and improve networks robustness when mmWave link is switched from LOS to NLOS condition.**

***Index Terms*—Multipath TCP (MPTCP), mmWave, Slow Start, BDP**

I. Introduction

Nowadays, most smart devices are equipped with multiple network interfaces. Both cellular and WiFi interfaces have been standard features of smart phones; Servers shipped with redundant Ethernet are also common in modern data centers. To provide improved throughput and better fairness, researchers proposed Multipath TCP (MPTCP) [1] which allows a single connection to transmit packets on several paths between hosts concurrently.

Besides traditional Radio Access Technologies (RATs), the combination of MPTCP and some new emerging wireless technologies is gaining extensive attention. In 5G mobile networks, millimeter wave (mmWave) communications have been considered to play an important role to offer huge bandwidth and magnitude greater throughput [2]. However, some characteristics of mmWave including high propagation loss, directivity and sensitivity to blockage pose significant challenges to maintain reliable and efficient transmission on mmWave links [3]. Using the multi-connectivity feature of MPTCP, mmWave-LTE hybrid networks can theoretically be an ideal end-to-end solution to improve the transmission robustness. When blockage or user mobility deteriorate performance of mmWave link, the traffic will be moved to the stable LTE link in a short time and high-speed transmission will be continued automatically once mmWave link recovers.

However, some severe issues of MPTCP performance in mmWave-LTE networks are observed in previous studies [4] [5]. In many cases, the total throughput of MPTCP connection is below that of single path TCP connection running on the mmWave link, and this phenomenon suggests that the congestion control of MPTCP cannot satisfy the goal of improve throughput proposed in RFC6356 [6].

In this paper, we perform a further study on this problem, and after analyze related experimental data, our conclusion is that the Slow Start scheme is the main reason leading to the throughput degradation. The present Slow Start mechanism will induce serious congestion window ($cwnd$) overshoot. Large number of out-of-order packets backlogged in receive buffer caused by excessive send rate and huge path heterogeneity lead to sharp throughput declines and poor bandwidth utilization on MPTCP subflows. Therefore, Slow Start scheme of MPTCP needs to be improved to achieve better performance in mmWave-LTE networks.

The main contributions of our work are the following:

- We analyze the performance problem of MPTCP in mmWave-LTE networks, and Slow Start scheme is identified as the major cause of this problem according to our experiments. Severe $cwnd$ overshooting due to current Slow Start results in massive out-of-order packets which lead to poor throughput performance of MPTCP in mmWave-LTE networks.
- We propose a BDP Estimate Based Slow Start (BESS) algorithm to solve the mentioned MPTCP performance problem in mmWave-LTE networks. For subflows in Slow Start phase, we will estimate the total Bandwidth Delay Product (BDP) of all MPTCP subflows and compare it with total inflight packets to determine whether to exit Slow Start and move to Congestion Avoidance. Additionally, in order to maintain fairness to single path TCP, the idea of coupled $ssthresh$ is absorbed and improved to be more reasonable.
- Our simulation results show that BESS can achieve significant throughput gain. In mmWave-LTE networks, BESS can improve throughput by up to 100% compared to current regular Slow Start. When compared with other proposed MPTCP Slow Start algorithm, BESS still have a distinct advantage in the terms of throughput performance.

The remainder of this paper is organized as follows. In Section II, we analyze the performance problem of MPTCP

978-1-5386-6358-5/18/$31.00 ©2018 IEEE

in mmWave-LTE networks and the influence of Slow Start on this problem. Section III presents the design details of BESS. We give the evaluation results of BESS in Section IV. Section V concludes the paper.

II. Problem Statement

Some investigations have been carried out to explore the interactions between MPTCP and mmWave-LTE networks in previous research. According to RFC6356, three design goals are set for MPTCP: improve throughput, do no harm and balance congestion. However, in [4], researchers found that MPTCP with coupled congestion control algorithm (BALIA [7], OLIA [8], etc.) fails to meet the first design goal of MPTCP. In many situations, MPTCPs throughput is lower than that of single path TCP and network bandwidth is wasted severely.

After analyze related experimental data, we observe that the Slow-Start scheme is the key reason leading to the throughput degradation. In current MPTCP, $cwnd$ control are uncoupled in Slow Start phase and the same operations as single path TCP are performed on each subflow. When a subflow receives an ACK, its $cwnd$ is increased by the number of previously unacknowledged bytes acknowledged in the incoming ACK, and the window size is doubled in every RTT until it reaches $ssthresh$. However, the initial value of $ssthresh$ is set to infinity in TCP stack according to RFC5681 [9], to ensure that the available network capacity probed by the newly established connection won't be limited by the initial $ssthresh$. Consequently, the exponential growth of $cwnd$ is kept until packet losses occur and when slow start stage ends, MPTCP subflows may seriously overshoot their $cwnd$ beyond path capacity.

More significantly, acute packet reordering problem is aggravated as a joint result of $cwnd$ overshooting and the path heterogeneity in mmWave-LTE networks. The Head-of-Line blocking effect [10] [11] and receive window reduction caused by these out-of-order packets result in dramatic throughput degradation and poor overall utilization. Figure 1 shows receive window and the free space of receive buffer which is shared by all subflows in a MPTCP connection versus time, and Fig. 2 presents the throughput degradation of a MPTCP connection on mmWave and LTE links. As shown in figures, after Slow Start ends, huge amounts of out-of-order packets flood into receive buffer in a matter of seconds, and subsequently, a sharp drop of throughput can be observed on each subflow. The transmission rate will be limited to a low level during the entire recovery process.

III. BESS Algorithm

In this section, we present the BESS algorithm to address the above problem. In our algorithm, the total BDP estimation of MPTCP subflows paths which are in Slow Start stage will be estimated, and the comparison of BDP and the total packets number inflight is the key criterion to avoid $cwnd$ overgrowth.

To estimate the total BDP of MPTCP Slow Start subflows, bandwidth and delay are estimated respectively as a first step. Various researchers have proposed solutions on bandwidth

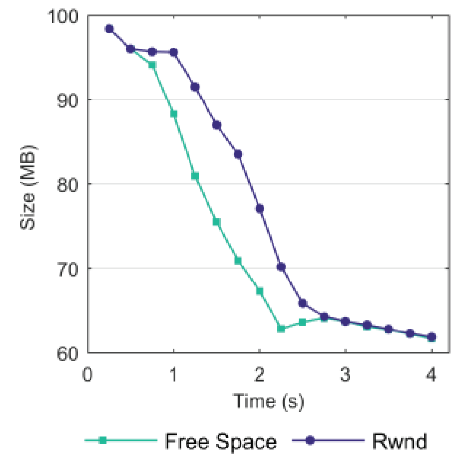

Fig. 1. Receive window and receive buffers free space of MPTCP in mmWave-LTE networks.

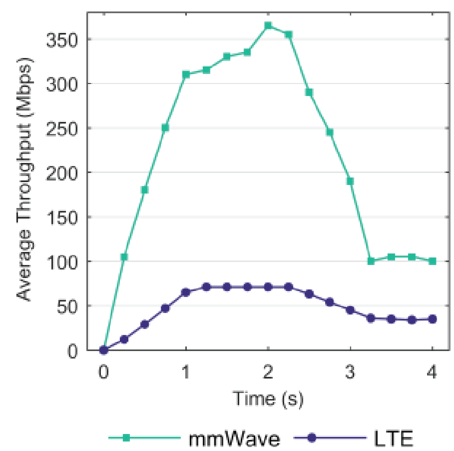

Fig. 2. Dramatic throughput degradation of MPTCP in mmWave-LTE networks.

estimation. In [12], researchers advance using packet train dispersion to calculates a bandwidth measurement. When N back-to-back packets of size L are sent to the receiver, these packets can be referred to a packet train and the train length is $\Delta(N) = \sum_{k=1}^{N-1} \delta^k$, where δ^k is the dispersion between packet k and $k+1$. In our design, for measurement reasons, corresponding ACK train replaces the packet train of which length is $\Lambda(N) = \sum_{k=1}^{N-1} \lambda^k$, and similarly λ^k is the dispersion between an ACK k and $k+1$.

Using the ACK train, the bandwidth $B(N)$ can be measured by:

$$B(N) = \frac{(N-1)L}{\Lambda(N)} \quad (1)$$

Therefore, in each RTT round, the bandwidth B_r of subflow

r can be calculated by:

$$B_r = \frac{PacketsNum_r}{\Lambda_r} \quad (2)$$

$PacketsNum_r$ is the number of packets sent in current RTT round, and Λ_r is the ACK train length.

In single path TCP transmission, BDP is simply the product of bandwidth and the minimum forward delay of the link. However, there are significant differences among subflows forward delay while calculation of the total BDP needs a unified representative forward delay value. Considering that the performance of MPTCP is largely dependent on the largest forward delay because of the presence of data reordering received over subflows having different delays, we choose the maximum value of subflows minimum forward delay to calculate the total BDP:

$$BDP_{total} = max_C(MinForwardDelay_r) \sum_C B_r \quad (3)$$

$$C = \{r | \text{subflow } r \text{ is in Slow Start stage}\} \quad (4)$$

After we get the estimation of total BDP of all Slow-Start subflows, we compare it with the total inflight packets number on these subflows. When the inflight packets approach the total BDP, this subflow will exit from Slow Start and switch to Congestion Avoidance.

Except realizing the throughput goal, MPTCP is also responsible for ensuring fairness to regular TCP. MPTCP should not take up more capacity than single path TCP. In [13], the concept of coupled $ssthresh$ was proposed to limit the total transmission rate of MPTCP in Slow Start to an $expected$ value, which is defined as the quotient of $conssthresh$ and the minimum value $baseRTT$ of all subflows RTT. From an overall perspective, fairness problem can be solved effectively through this method. However, we think that the direct comparison of total transmission rate of MPTCP and the $expected$ throughput is inappropriate. The $expected$ throughput is designed to be the representative value of imaginary single path TCP throughput when Slow Start phase ends, while the total throughput of MPTCP also contains some subflows variation in other stages such as Congestion Avoidance. Therefore, some improvements are needed to make algorithm more reasonable and precise.

Our solution is as follows: When a subflow exit Slow Start, the current throughput of this subflow will be recorded as $OccupiedRate_r$, and this value will be reset when this subflow moves to Slow Start. The upper bound of the total throughput of other Slow Start subflows is calculated by:

$$BoundRate = \frac{conssthresh}{baseRTT} \sum_s OccupiedRate_s \quad (5)$$

On each ACK, if the total rate of subflows which are in Slow Start approaches the $BoundRate$, the subflow will exit Slow Start and stop the exponential growth of cwnd. Algorithm 1 shows the pseudo-code of BESS.

Algorithm 1 BDP Estimate Based Slow Start (BESS) algorithm

Initialize:
$sspermit_r = 1$ $conssthresh = ssthresh$
$mindelay_r = 0$ $occupiedrate_r = 0$
At the start of each RTT round:
if $sspermit_r$ **and** $cwnd_r \leq ssthresh_r$ **then**
  $trainstart_r = now$
end if
On each ACK:
$mindelay_r = min(mindelay_r, RTT_r)$
if $sspermit_r$ **and** $cwnd_r \leq ssthresh_r$ **then**
  $totalBDP = \sum_{sspermit_s=1} \frac{packetsnum_s}{now-trainstart_s} max\left(\frac{mindelay_i}{2}\right)$
  if $\sum_{sspermit_s=1} inflight_s \geq totalBDP$ **then**
    $sspermit_r = 0$
    $occupiedrate_r = \frac{cwnd_r}{RTT_r}$
  end if
  $totalssrate = 0$
  for each subflow s **do**
    if $sspermit_s$ **and** $cwnd_s \leq ssthresh_s$ **then**
      $totalssrate+ = \frac{cwnd_s}{RTT_s}$
    end if
  end for
  if $totalssrate > \frac{conssthresh}{baseRTT} - \sum_s occupiedrate_s$ **then**
    $sspermit_r = 0$
    $occupiedrate_r = \frac{cwnd_r}{RTT_r}$
  end if
end if
if $!sspermit_r$ **and** $cwnd_r \leq ssthresh_r$ **then**
  $ssthresh_r = cwnd_r$
end if
if $sspermit_r$ **and** $cwnd_r > ssthresh_r$ **then**
  $sspermit_r = 0$
  $occupiedrate_r = \frac{cwnd_r}{RTT_r}$
end if
Timeout:
$sspermit_r = 1$ $occupiedrate_r = 0$ $mindelay_r = 0$

IV. Evaluation

In this section, we implemented BESS algorithm and a series of simulation experiments are performed to evaluate our novel algorithm in mmWave-LTE networks.

A. Simulation Setup

The simulations are based on a combination of MPTCP implementation in the Linux kernel [14] and NYU ns-3 mmWave module [15] with the help of Direct Code Execution (DCE) library [16]. A typical MPTCP-enabled mmWave-LTE network scenario shown in Fig. 3 is established and configured to carry out experiments. The bandwidth of mmWave and LTE links are fixed to 500Mbps and 80Mbps respectively, and the setup of other main parameters is same as a typical setup in [17].

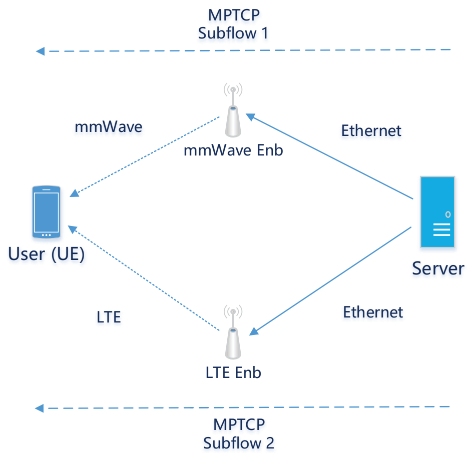

Fig. 3. MmWave-LTE network scenario which enables MPTCP used in the experiments.

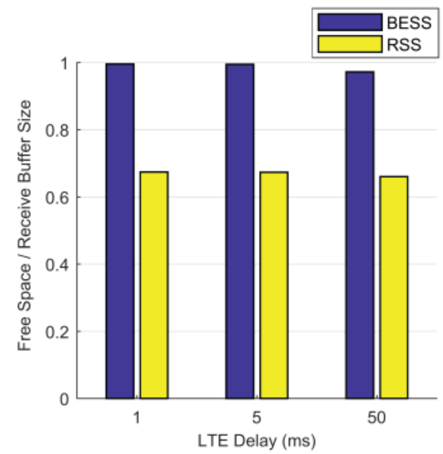

Fig. 5. Average throughput under different UE-eNB distance.

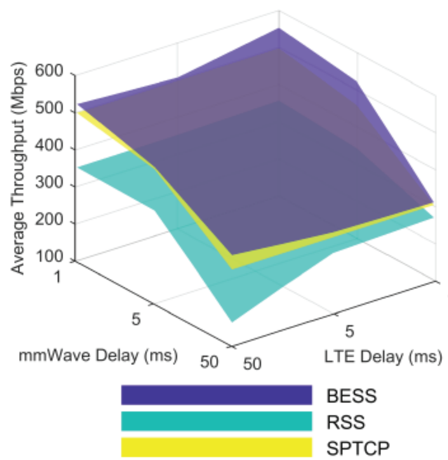

Fig. 4. Average throughput under different link delay.

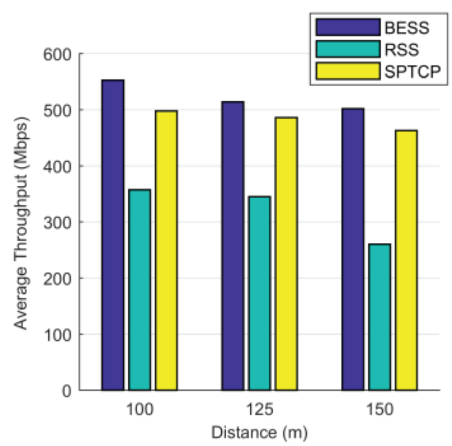

Fig. 6. Average throughput under different UE-eNB distance.

B. *Performance Comparison*

In our experiments, the performance of BESS is compared with that of regular Slow Start (RSS) and single path TCP (SPTCP). BESS and RSS are plugged into MPTCP with the BALIA congestion control algorithm, and SPTCP is with TCP CUBIC [18]. Kinds of factors including path delay, UE-eNB distance and LOS-NLOS transition are varied to make a comprehensive comparison, and some analyses are followed on these factors and their influence on system performance.

a) Path Delay: At first, an important issue that influences MPTCP performance is delay of subflows. We varied the delay of mmWave and LTE links from 1 to 50 ms. Figure 4 shows the impact of delay on performance of the whole system. When the mmWave delay of mmWave link is low (less than 5ms), the throughput remains stable and bandwidth are fully utilized with BESS or SPTCP, and the overall throughput of BESS is higher in most cases because MPTCP with BESS can make full use of bandwidth of mmWave link and LTE link simultaneously. However, the throughput of MPTCP with RSS can only reach about 350Mbps and a great deal of network resources are wasted as we observed in Section II. When the delay rises to 50ms, the throughput of three algorithms falls slightly. On the other hand, due to the huge difference between mmWave and LTE bandwidth, the simple increase of LTE links delay has no remarkable impact on total throughput in our observation.

Furthermore, we analyze the influence of the difference between delay of links from the perspective of out-of-order packets in receive buffer. MmWave links delay is fixed to 1ms while the delay of LTE link is varied from 1ms to 50ms. Figure 5 shows the percentage of free space in receive buffer. A first observation is that the BESS has addressed the out-of-order packets problem due to *cwnd* overgrowth and the backlog in receive buffer is exceedingly mild. Besides, when the delay difference increases, the number of out-of-order packets will rise accordingly, but only very slightly.

b) UE-eNB Distance: It has been shown that the increase of distance between the user equipment (UE) and the evolved

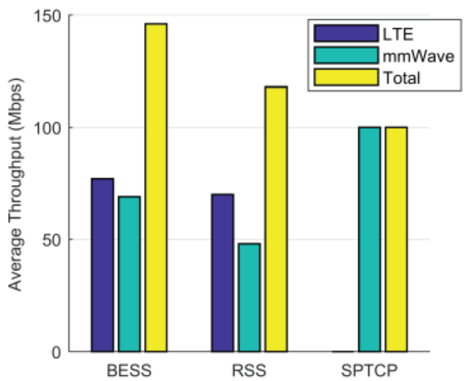

Fig. 7. Average throughput with LOS-NLOS transition.

node base (eNB) will degrade the throughput of mmWave networks [4]. The UE-eNB distance in the experiments is varied from 100m to 150m. Similarly, as seen in fig. 6, the throughput of BESS is still the highest when distance changes. Besides, the throughput of three algorithms experience a slight dip when UE-eNB distance is increased.

c) LOS-NLOS transition: Another significant factor that affects MPTCP and TCP in mmWave networks is LOS-NLOS transition. The non-line-of-sight (NLOS) channel suffers from higher attenuation than the line-of-sight (LOS) channel [19], and the throughput be substantially impacted when mmWave link is switched from LOS to NLOS. We perform related experiments to study algorithms performance with LOS-NLOS transition. In 0 7s, UE is in LOS condition and in 7 26s, UE is blocked by a building and switched to NLOS condition. Figure 7 shows the LTE throughput, the mmWave throughput and the total throughput under the control of three algorithms. From fig. 7, BESS has the obvious advantage with respect to average throughput, and the main reason is that the throughput on LTE link maintains stable after UE moves to NLOS condition while that on mmWave link is limited to approximately 15Mbps. Therefore, MPTCP with proper $cwnd$ control mechanism can provide more reliable and efficient transmission than single path TCP when LOS-NLOS transition occurs.

V. Conclusion

In this paper, we first present an in-depth analysis on the performance issues of MPTCP in mmWave-LTE networks. To address this problem, we propose a BDP Estimate Based Slow Start (BESS) algorithm for MPTCP which estimate total BDP of all Slow Start subflows and decide whether to exit Slow Start according to the comparison of BDP and inflight packets, and meanwhile TCP fairness is guaranteed to some extent. Our simulation results show that BESS can improve throughput of MPTCP in mmWave-LTE networks effectively and MPTCP with BESS allows a more efficient response to performance degradation due to LOS-NLOS transitions.

Acknowledgment

This work was supported by National Science and Technology Major Project of China MIIT (Grant No.2017ZX03001011-004), the Fundamental Research Funds for the Central Universities.

References

[1] A. Ford, C. Raiciu, M. Handley, and O. Bonaventure, “Tcp extensions for multipath operation with multiple addresses,” Tech. Rep., 2013.
[2] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, “Five disruptive technology directions for 5g,” *IEEE Communications Magazine*, vol. 52, no. 2, pp. 74–80, 2014.
[3] S. Rangan, T. S. Rappaport, and E. Erkip, “Millimeter-wave cellular wireless networks: Potentials and challenges,” *Proceedings of the IEEE*, vol. 102, no. 3, pp. 366–385, 2014.
[4] M. Polese, R. Jana, and M. Zorzi, “Tcp and mp-tcp in 5g mmwave networks,” *IEEE Internet Computing*, vol. 21, no. 5, pp. 12–19, 2017.
[5] ——, “Tcp in 5g mmwave networks: Link level retransmissions and mp-tcp,” in *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, May 2017, pp. 343–348.
[6] C. Raiciu, M. Handley, and D. Wischik, “Coupled congestion control for multipath transport protocols,” Tech. Rep., 2011.
[7] Q. Peng, A. Walid, J. Hwang, and S. H. Low, “Multipath tcp: Analysis, design, and implementation,” *IEEE/ACM Transactions on Networking*, vol. 24, no. 1, pp. 596–609, 2016.
[8] R. Khalili, N. Gast, M. Popovic, and J.-Y. Le Boudec, “Mptcp is not pareto-optimal: performance issues and a possible solution,” *IEEE/ACM Transactions on Networking (ToN)*, vol. 21, no. 5, pp. 1651–1665, 2013.
[9] M. Allman, V. Paxson, and E. Blanton, “Tcp congestion control,” Tech. Rep., 2009.
[10] M. Scharf and S. Kiesel, “Head-of-line blocking in tcp and sctp: Analysis and measurements,” in *Global Communications Conference (GLOBECOM), 2006 IEEE*. IEEE, 2006.
[11] S. Ferlin, T. Dreibholz, and Ö. Alay, “Multi-path transport over heterogeneous wireless networks: Does it really pay off?” in *Global Communications Conference (GLOBECOM), 2014 IEEE*. IEEE, 2014, pp. 4807–4813.
[12] C. Dovrolis, P. Ramanathan, and D. Moore, “Packet-dispersion techniques and a capacity-estimation methodology,” *IEEE/ACM Transactions On Networking*, vol. 12, no. 6, pp. 963–977, 2004.
[13] Y. Wang, K. Xue, H. Yue, J. Han, Q. Xu, and P. Hong, “Coupled slow-start: Improving the efficiency and friendliness of mptcp’s slow-start,” in *GLOBECOM 2017-2017 IEEE Global Communications Conference*. IEEE, 2017, pp. 1–6.
[14] C. Paasch, S. Barré *et al.*, “Multipath tcp in the linux kernel,” *Available from {www. multipath-tcp. org.}*, 2013.
[15] M. Mezzavilla, M. Zhang, M. Polese, R. Ford, S. Dutta, S. Rangan, and M. Zorzi, “End-to-end simulation of 5g mmwave networks,” *IEEE Communications Surveys & Tutorials*, 2018.
[16] H. Tazaki, F. Uarbani, E. Mancini, M. Lacage, D. Camara, T. Turletti, and W. Dabbous, “Direct code execution: Revisiting library os architecture for reproducible network experiments,” in *Proceedings of the ninth ACM conference on Emerging networking experiments and technologies*. ACM, 2013, pp. 217–228.
[17] M. Polese, M. Giordani, M. Mezzavilla, S. Rangan, and M. Zorzi, “Improved handover through dual connectivity in 5g mmwave mobile networks,” *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 9, pp. 2069–2084, 2017.
[18] S. Ha, I. Rhee, and L. Xu, “Cubic: a new tcp-friendly high-speed tcp variant,” *ACM SIGOPS operating systems review*, vol. 42, no. 5, pp. 64–74, 2008.
[19] S. Geng, J. Kivinen, X. Zhao, and P. Vainikainen, “Millimeter-wave propagation channel characterization for short-range wireless communications,” *IEEE Transactions on Vehicular Technology*, vol. 58, no. 1, pp. 3–13, 2009.