An Experimental Study on Multipath TCP Congestion Control with Heterogeneous Radio Access Technologies

Monika Prakash§, Atef Abdrabou*§, *Member, IEEE*, and Weihua Zhuang†, *Fellow, IEEE*
Email:{monika.p@uaeu.ac.ae, atef.abdrabou@uaeu.ac.ae, wzhuang@uwaterloo.ca}

***Abstract*—In the near future, a large volume of the data traversing wireless networks will not only be requested and/or reported by humans but also by machines (e.g., Internet-of-things and machine-to-machine applications). This mandates the availability of enormous wireless bandwidth and an end-to-end reliable information transfer. Currently, many wireless devices are equipped with two wireless interfaces and covered by heterogeneous radio access technologies (RATs). Thus, the usage of a transport layer designed for multi-homed devices such as multipath transmission control protocol (MPTCP) is inevitable. This paper experimentally investigate the performance of three congestion control algorithms, which can be used by MPTCP, namely, Cubic, linked-increases algorithm (LIA), and opportunistic LIA (OLIA). The testbed comprises real (not simulated) LTE and WiFi networks that are used to connect dual-homed wireless nodes to one another. The paper comparatively studies the throughput performance of the three algorithms under varying factors including the receiver buffer size, the number of parallel connections, the data volume, and the flow lifetime. Our key findings reveal that although Cubic is not designed with multipath in mind, it outperforms the multipath-based LIA and OLIA, whenever LTE per-node capacity is higher than its WiFi counterpart. Also, in a reversed situation (WiFi per-node capacity is higher) Cubic outperforms OLIA and LIA for short-lived flows.**

***Index Terms*—Multipath TCP, congestion control, heterogeneous, radio access technologies.**

I. Introduction

Over the past two decades, wireless communication systems have witnessed phenomenal progress. The advancements in the wireless communication industry have lead to a tremendous increase in the number of smart personal wireless devices. Moreover, a multitude of machine-to-machine (M2M) and Internet-of-Things (IoT) applications rely heavily on wireless networks such as e-healthcare, smart cities, intelligent transport system, to name a few. This is anticipated to result in an explosive growth of bandwidth-hungry applications. The evolving fifth generation (5G) systems aim at satisfying the requirements of these applications by providing, among other features, high data rates, low network latency, and ubiquitous connectivity.

One way of realizing these features is by exploiting the overlapped coverage of multiple RATs and the increasing penetration of multihomed wireless devices [1]–[4]. This mandates the efficient utilization of the resources available in co-located heterogeneous wireless networks (e.g., LTE and WiFi) while providing end-to-end reliable data transfer. However, the legacy Internet transport layer protocol, TCP (transmission control protocol) is not designed to simultaneously transfer data over multiple network interfaces of multihomed nodes (e.g., no resource scheduling or fault tolerance capability).

Therefore, the Internet engineering task force (IETF) has recently introduced the multipath TCP (MPTCP) as an extension to legacy TCP for multipath connections [5]. MPTCP provides the ability to distribute the application layer traffic over multiple network interfaces of a device. Since the MPTCP protocol is built on the legacy TCP, which is mainly designed for wireline networks, its performance over wireless networks is expected to be affected by wireless channel impairments. This is due to the fact that all the available congestion control mechanisms are triggered by events that include lost data segments or a lost acknowledgment, which are very likely to happen over a wireless link without a real congestion condition.

Thus, the objective of this research is to experimentally investigate the behavior of MPTCP in a multiple-RAT environment. We are particularly interested in examining the throughput performance of three congestion control algorithms when used by MPTCP over heterogeneous wireless networks. Using comprehensive experimentation, our study compares between the throughput of LIA [6], OLIA [7], and Cubic [8] algorithms under varying performance-affecting factors. Different from Cubic, LIA and OLIA are designed with multipath in mind. The employed testbed is configured in three different setups that mimic possible connectivity scenarios including M2M and IoT communications. The testbed utilizes real (not simulated) LTE evolved packet core (EPC) with a real eNodeB in addition to real WiFi networks and Ethernet-based backbone.

The paper makes the following contributions. First, a comprehensive experimental study for the MPTCP throughput performance of Cubic, LIA, and OLIA congestion control algorithms is provided for three different lab setups (network configurations). The effect of varying factors, such as receiver buffer size, number of parallel connections, transmitted data volume, and flow lifetime, on the achievable throughput of a dual-homed wireless node is investigated for a number of nodes concurrently sending data distributed over LTE and

*Corresponding Author.
§M. Prakash and A. Abdrabou are with the Department of Electrical Engineering, UAE University, Al-Ain, PO 15551, UAE.
†W. Zhuang is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, N2L 3G1, Canada.

WiFi interfaces. This includes examining how each congestion control algorithm assigns a data traffic share to each network interface. Second, the overall MPTCP throughput is comparatively studied for the three algorithms with different WiFi standards representing a wide range of WiFi transmission rates available in practice. Third, the overall throughput gain of MPTCP over single-homed connections is quantified under the aforementioned factors.

The rest of the paper is organized as follows. Section II introduces the most relevant research works in the literature. Section III provides a basic background about Cubic, LIA, and OLIA algorithms. The system model is described in Section IV. Section V presents the details of the experimental setups including the hardware and software tools used. The experimental study is provided in Section VI. Finally, Section VII concludes the paper.

II. Related Works

The MPTCP architecture is introduced in RFC 6182 [5]. In 2013, it has been published as an experimental standard in RFC 6824 [9]. Over the past years, there has been a wealth of research works related to the design, implementation, and performance of MPTCP. For instance, the design and implementation of MPTCP have been addressed in [10]–[12] and the references therein.

On the other hand, the performance of MPTCP is investigated in several research works. The authors in [13] study the impact of the variation of delay, bandwidth, and frame error rate among subflows on the MPTCP throughput performance using the ns-3 simulator. In [14], a study on the impact of the choice of the primary path on MPTCP throughput is performed using the Mininet emulator. The impact of path characteristics (round trip delay and bandwidth) in addition to scheduling policies on MPTCP throughput performance are investigated in [15] using the ns-3 simulator.

The authors in [16] studies the bandwidth utilization and file transmission time of MPTCP with multihomed nodes that have wired (Ethernet) and wireless (WiFi) interfaces using a testbed. Alheid et al. in [17] investigate the impact of different packet reordering solutions on MPTCP aggregate throughput and path utilization using the ns-3 simulator. In [18], the MPTCP performance is explored in terms of file download time, round trip delay (RTT), out-of-order delay, and segment loss rate with three congestion control algorithms (OLIA, LIA, and Reno). However, study is performed using only one dual-interface sending node (a laptop) with 3G/4G and WiFi interfaces. The measurements are recorded based on data traffic transferred through public 3G/4G networks to a remote server via the Internet, which involve unspecified amount of background traffic concurrently transmitted through the 3G/4G networks.

To the best of our knowledge, no other research work in the literature experimentally investigates the MPTCP throughput with a focus on congestion control using a testbed that contains a number of dual-homed wireless nodes concurrently sending data through the uplink of a full-fledged EPC with a real (not simulated) eNodeB and a real WiFi network. The study comparatively evaluates the performance of Cubic, LIA, and OLIA algorithms under throughput-influencing factors and a network loading condition in a heterogeneous wireless network lab setting that emulates different real-world connectivity scenarios including IoT and M2M communications.

III. Background Information on Congestion Control Algorithms

This section introduces some background information about the congestion control algorithms studied in this paper. LIA and OLIA are congestion control algorithms designed with multipath in mind, whereas Cubic is designed to be a single path congestion control algorithm. When Cubic is used in a multipath scenario, the MPTCP default scheduler distributes the traffic on subflows according to RTT, however, the congestion control runs separately on each individual subflow (i.e, without considering the status of other subflows). In the sequel, a brief description of the operation of the three algorithms is offered.

A. Cubic

Cubic is the current default congestion control algorithm in Linux. Cubic replaced the linear congestion window growth function of the standard TCP [19] by a cubic function in order to improve the scalability of TCP over fast and long distance networks [8]. The congestion window grows with time starting from the last congestion event as given in

$$W(t) = C(t-K)^3 + W_{max} \tag{1}$$

where C is a Cubic parameter, t is the elapsed time from the last window reduction, and K is the time period that the above function takes to increase to W_{max} when there is no loss events. Also, it is important to note that Cubic's window growth is independent of RTT.

B. LIA

LIA couples the additive increase functions of the subflows. For each ACK received on subflow i, LIA increases its congestion window by the following:

$$min(\alpha L \frac{MSS_i}{W_{sum}}, L\frac{MSS_i}{W_i}) \tag{2}$$

where

$$\alpha = W_{sum} \frac{max(\frac{W_i}{RTT_i^2})}{\left(\sum\limits_{i=1}^{N} \frac{W_i}{RTT_i}\right)^2}.$$

Here, L is the number of the acknowledged bytes, MSS_i is the maximum segment size of subflow i, W_i is the congestion window of subflow i, N is the number of subflows, and W_{sum} is the sum of the congestion window of all subflows.

LIA uses an unmodified standard TCP behavior in case of a packet loss [6]. The algorithm is designed to increase throughput and friendliness with other coexistent TCP flows.

C. OLIA

Similar to LIA, OLIA couples the additive increasing function of the congestion window size of the MPTCP subflows, whereas it uses an unmodified standard TCP behavior in case a segment loss event occurs. When a subflow i receives an ACK, its congestion window is increased by the following amount

$$\left(\frac{\frac{W_i}{RTT_i^2}}{(\sum_{k=1}^{N} \frac{W_k}{RTT_k^2})^2} + \frac{\alpha_i}{W_i} \right) L\ MSS_i$$

where α_i is computed based on the number of transmitted bytes since the last loss.

IV. System Model

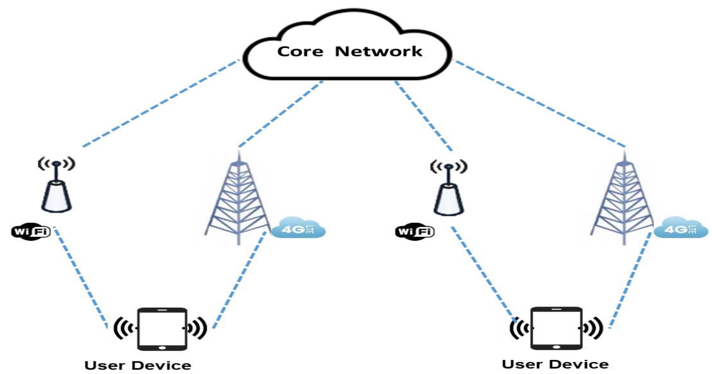
The expected growth in the popularity of M2M/IoT applications mandates a tremendous demand for wireless bandwidth. In fact, a significant number of these applications depend on sensing/monitoring some phenomena over a period of time then sending the recorded measurements to some information management entity for displaying or processing. Other applications rely on sharing the data of certain detected events and/or alarms, over some time duration, between sensory nodes for analysis and/or taking actions. Indeed, both types of applications require reliable and high throughput data transfer. In addition, the originated traffic from these applications is mainly uplink-oriented.

In the system model, we address the aforementioned types of applications, where a number of stationary dual-homed wireless nodes concurrently share some data volume with stationary dual-homed receiver nodes. As the sending nodes are assumed to be covered by different radio access technologies, the MPTCP is used as a reliable transport layer to send the data segments distributed over two wireless networks. The system model covers three different configurations that reflect common practical scenarios.

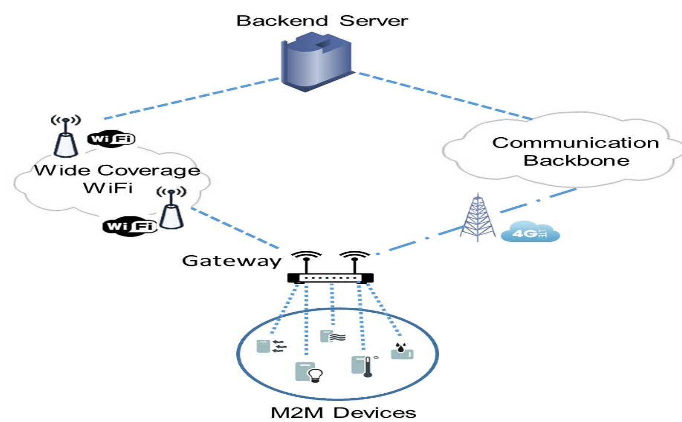
The Fig. 1 shows the three configurations. The first configuration targets a scenario where both the sending and receiving nodes are covered by two common radio access technologies such as WiFi and LTE.

The second configuration addresses a scenario where the sending nodes are covered by WiFi and LTE networks, whereas the receiving nodes are also covered by a WiFi network and another wired or wireless technology such as DSL or WiMAX, respectively. This scenario can represent the communication between capillary M2M gateways (sending nodes) and their corresponding management servers (receiving nodes) through the backbone of a carrier LTE network and a WiFi network with a wide coverage.

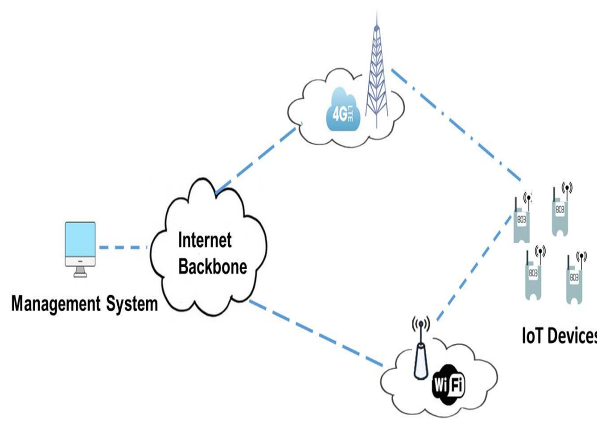
The third configuration portrays a typical IoT architecture where the sending nodes are connected to the Internet through the overlapped coverage of WiFi and LTE networks while the managing/monitoring nodes receive the data through a single network (e.g., WiFi). Patient health status tracking represents a typical application for this configuration.



(a) First configuration.



(b) Second configuration.



(c) Third configuration.

Fig. 1: System model.

V. Experimental Setup

The section introduces the details of the experimental setup including the testbed configuration, the hardware, and the software tools, which are utilized to obtain the experimental results.

A. Testbed Description

The testbed uses real (not simulated) networks of different radio access technologies. One is a single-channel contention-based network following the WiFi IEEE 802.11 medium access control (MAC) protocol. The other is a contention-free single-cell LTE network with a real eNodeB.

The experimental study is performed using three different setups of the same testbed in order to mimic the different scenarios described in Section IV. In all setups, 10 dual-homed wireless nodes are used to create 5 sender-receiver pairs, which are located in an indoor area covering around 1000 sq ft. Each experiment is repeated at least 50 times for accurate statistics.

As depicted in the Fig. 2, in Setup I, each transceiver of a dual-homed node is connected to a different infrastructure based wireless network. The WiFi interfaces of the sender nodes are connected to WiFi Router 1, whereas the interfaces of receiver nodes are connected to WiFi Router 2. The LTE interfaces of the sender and receiver nodes are connected to the real eNodeB (i.e., they share the capacity offered by the eNodeB). This configuration makes the subflows of the WiFi and LTE networks independent of each other, and hence the subflows of the same MPTCP connection do not share a common bottleneck.

In Setup II, the transceivers of the dual-homed sender nodes are connected to a WiFi network (via Router 1) and eNodeB of the LTE network similar to Setup I. Also, as in Setup I, one wireless interface of a dual-homed receiver node is connected to different a WiFi network via Router 2 as shown in Fig. 3. However, the second interface of a receiver node is connected to another WiFi network via Router 3. This WiFi network emulates the connectivity of the other interface to the cellular backbone network, which can be done via a wired (e.g., DSL) or another wireless technology (e.g., WiMax). Since the second interface of a receiver node is not cellular, the transmitted segments have to pass through the LTE core network (i.e., the evolved packet core) through the mobility management entity (MME), the serving gateway (SGW), and the packet data network gateway (PGW). Again, in this setup, there is no common bottleneck link that the subflows of one MPTCP connection have to occupy.

The Fig. 4 shows that the major difference between Setup 3 and the other two setups is the existence of a bottleneck link between the subflows of the same MPTCP connection. In this setup, the interfaces of the sender nodes are connected the same way as the other two setups. However, the two interfaces of a receiver node are connected to the same WiFi network via Router 2 as revealed in Fig. 4. This forms a bottleneck network as all the subflows have to share its capacity.

B. Hardware and Software Tools

The section describes the main testbed hardware and the software tools used to perform the experimental study.

1) *The LTE Network:* A real evolved packet core (EPC) [20] is used in the testbed. The EPC realizes the MME, the SGW, the PGW, and the home subscriber server (HSS) according to LTE Release 14 standard [21]. The EPC can be connected to any external network via PGW through a Gigabit Ethernet interface. A real eNode B is connected to the EPC. It is implemented using a software-defined radio and supports frequency division duplex (FDD) mode with a bandwidth up to 20 MHz. FDD Band 7 (2.6 GHz) is used in all experiments. A command line interface is available to monitor the operation of the EPC and the eNodeB.

The computing nodes connect to the LTE network using Huawei E3276 FDD USB adapters. The adapters implement LTE Release 8 [21], which allows peak uplink and downlink speeds of 50 Mbps and 150 Mbps, respectively.

2) *The WiFi Networks:* In all the three setups, real WiFi networks are used. They are implemented using three WiFi routers (dual-band ASUS AC1900), which can support IEEE 802.11a/g/n/ac standards over either the 2.4 GHz or 5 GHz according to the selected standard. The routers are configured as access points as revealed in Figures 2-4. The computing nodes connect to the WiFi networks using USB WiFi adapters (ASUS Dual-band AC56), which support the same standards as the WiFi routers.

3) *The Computing Nodes:* The computing nodes are single board computers (Toradex Robin Z530L), which are equipped with an Intel Atom Z530 processor running at 1.6 GHz clock speed with 1 GBytes of memory. Ubuntu Linux server version runs on all the nodes. We use the default Linux kernel implementation of MPTCP, which provides the flexibility of selecting the congestion control algorithm to be used. The default packet scheduler of MPTCP is employed. It distributes the data traffic over different subflows based on the RTT of the subflow.

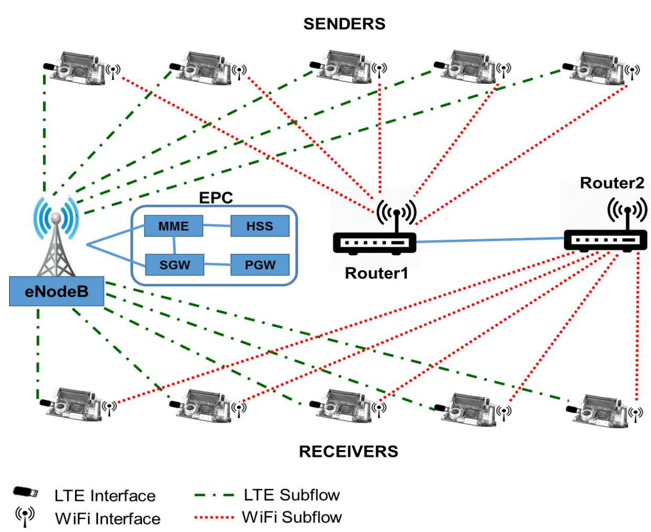


Fig. 2: Experimental Setup I.

C. Software Tools

Several software tools are employed in order to automate the execution of lab experiments and record the results as in

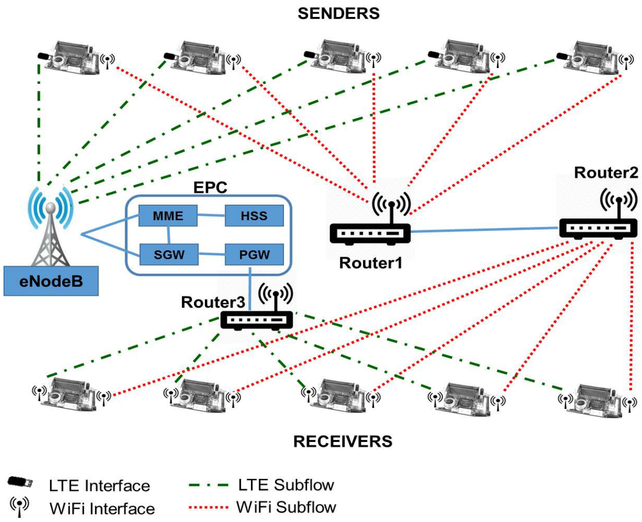


Fig. 3: Experimental Setup II.

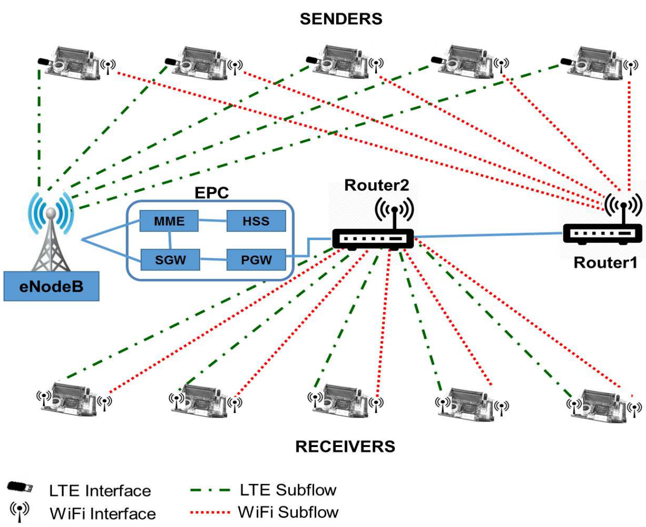


Fig. 4: Experimental Setup III.

the following. Parallel secure shell (PSSH) is used to issue the data transfer commands simultaneously to all the computing nodes. Data traffic generation is performed using the *iperf* tool. It allows the tuning of various TCP-related parameters such as number of parallel connections, receiver buffer size, and data volume size. Nicstat tool is used to measure the throughput of each wireless interface. TCP Probe is used in order to analyze the variation of the congestion window size and RTT.

VI. Experimental Study

In this section, the results of our extensive experimentation are introduced. Our study focuses on three different performance aspects of the MPCTP protocol. First, we investigate the assignment of data traffic shares of WiFi and LTE interfaces among the three congestion control algorithms (Cubic, LIA, and OLIA) under different performance-controlling parameters. Second, the overall MPTCP performance of the three algorithms is studied for commonly-used WiFi standards using different frequency bands. Third, the performance gain of the MPTCP with dual-homed devices compared with single-homed ones (WiFi or LTE) is introduced.

The performance of the studied congestion control algorithms is compared in terms of the data transfer throughput, which is measured by the time required to transfer the data from the sender to the receiver application. Note that all the experimental results are obtained with applying no rate limitation to any of the WiFi adapters, routers, or to the LTE eNodeB.

A. MPTCP performance comparison in different setups over different parameters

The section investigates how each congestion control algorithm handles the distribution of data traffic between the WiFi and LTE interfaces. The section particularly addresses the following parameters:

- The receiver buffer size, which represents the maximum congestion window size. We study this parameter as it controls the amount of data being transmitted over both networks by the data senders.
- The data volume, which is represented here by the size of the file being transmitted. In practice, it can also represent the amount of data that is exchanged between devices or machines in a data session.
- The number of parallel MPTCP connections that carry the test data volume. Studying this parameters allows investigating how each congestion control algorithm behaves in case an aggregated amount of data is sent over a number of connections [22]. It also helps in understanding how the different algorithms handle short-lived data connections that carries similar amount of data and all started within a short period of time. A situation that can readily happen if multiple machines/devices are sending a relatively small amount of data almost simultaneously to a gateway or a UE that simultaneously transmits different data streams.

It is worth noting that the study of these parameters is experimentally performed over different setups with two different WiFi standards, namely, IEEE 802.11g at 2.4 GHz and IEEE 802.11a/n at 5 GHz. Different WiFi standards achieve different transmission rates in the lab environment, and hence, affect the data rate difference between the LTE and WiFi interfaces.

1) Number of Parallel Connections: Fig. 5 shows the throughput share of the WiFi and LTE interfaces for Cubic, LIA, and OLIA algorithms with different number of parallel connections for Setup I and II. Setup III results are omitted as they yield no different conclusion. In this experiment, the receiver buffer size is set at 1000 KB and the overall data volume transferred over all connections is 15 MB.

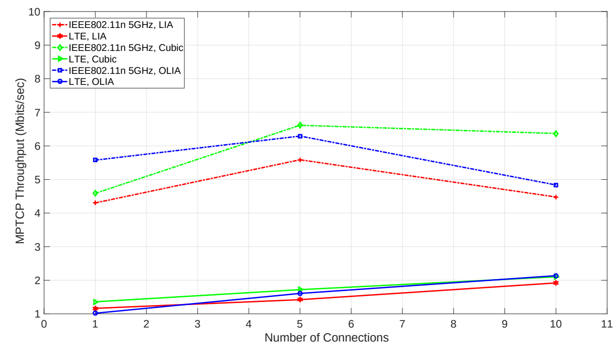
Fig. 5(a) reveals the results obtained from Setup I. We generally observe a better WiFi performance compared with LTE, which is mainly attributed to relaying the data over two cellular links of the same cell. The figure also clearly shows an increase of the throughput share for both WiFi and LTE interfaces as the number of connections is increased from 1 to 5. The reason for this increase is that any ACK timeout or 3-Duplicate-ACK event causes a reset of the congestion window of its own connection. Thus, the case of sending the overall amount of data (e.g., aggregated traffic) over one single-connection may suffer from a congestion window reset that affects the transfer of the whole amount of data compared with 5 connections. However, we observe that increasing the number of connections beyond 5 negatively affected the WiFi throughput share for LIA and OLIA algorithms in contrary with LTE. We found the reason of this trait is the higher RTT values of the WiFi data in the 10-connection case. Since all WiFi connections share the same interface queue, this makes the segments from any connection waits for a longer time as the number of connections grow. This does not lead to a significant increase of the RTT of the LTE interface as already small amount of data is pushed over this interface due to its low capacity.

Apparently, OLIA performs slightly better than LIA since its congestion window increasing function, unlike LIA, does not depend only on RTT but also on packet loss. Cubic performs the best as its congestion window's growth function depends mainly on packet loss events not RTT.

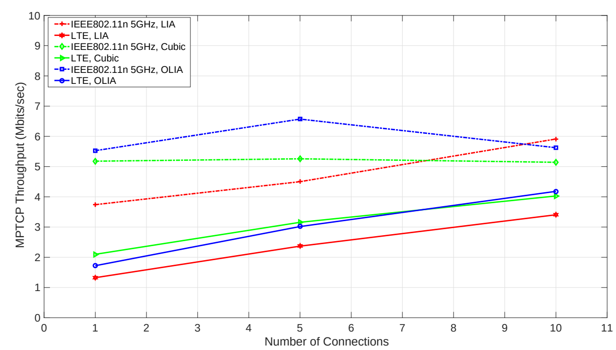
Fig. 5(b) shows the MPTCP throughput share of LTE and WiFi interfaces for Setup II. The figure exhibits a similar increasing trend for all algorithms in case of LTE as in Fig. 5(a), where the LTE link capacity here is higher since no relaying is done in this setup. However, the senders' WiFi network gets less traffic load since more traffic is pushed on the LTE network with increasing the number of connections. This generally decreases the WiFi link RTT and leads to an increasing trend of the throughput for LIA algorithm. Similar trend was expected for OLIA, however, for the 10-connection case, we found that the RTT for LTE and WiFi were close in a way such that the MPTCP scheduler reduced the amount of data to be sent over WiFi while it pushed more over LTE (as it generally suffers less loss events). It is readily seen that Cubic does not push more traffic on WiFi irrespective of the number of connections since the lighter load on WiFi (compared with Fig. 5(a)) reduces the segment loss due to packet collisions, and hence increasing the number of connections does not lead to a significant effect.

Different Setup II results are displayed in Fig. 5(c), which shows the LTE and WiFi throughput shares in case the WiFi link (IEEE 802.11g) has less capacity compared with the LTE link. This case can happen in practice in outdoor scenarios where the performance of IEEE 802.11n has been shown to approach the IEEE 802.11g [23]. The figure reveals a low throughput with single connection case but an increasing throughput trend for all algorithms for the LTE interface. This is consistent with the previous results and reflects a practical situation where the operator LTE network is loaded, which leads to congested routers/gateways. This significantly affects congestion window size due to frequent resetting. Since the WiFi network capacity is lower than the LTE network, all algorithms push less data over the WiFi interfaces with increasing the number of connections from 1 to 5 since both the RTT and segment loss are higher. A further slight decrease of WiFi throughput is observed OLIA for the 10-connection case as the growth function takes into account both RTT and packet loss events. It is found that increasing the number of connections more than 10 leads to sending the majority of the traffic over the primary interface (WiFi), which yields results similar to the ones shown in Fig. 10.

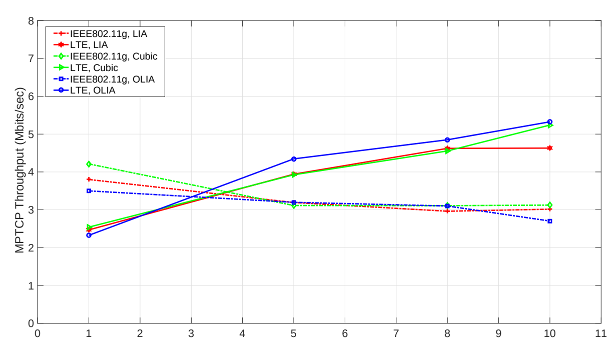
In short, our observations reveal the following: (i) Sending a data volume (e.g., aggregate) over a number of parallel MPTCP connections or if there are mulitple short-lived flows running simultaneously, the overall MPTCP throughput generally increases for Cubic, LIA, and OLIA algorithms, (ii) An exception happens if the WiFi available network capacity is remarkably higher than the available LTE network capacity (e.g., around three times). In this case, the overall throughput performance of LIA and OLIA starts to degrade if the number of connections is significantly increased.



(a) Setup I.



(b) Setup II (IEEE802.11n 5GHz).



(c) Setup II (IEEE802.11g).

Fig. 5: MPTCP throughput with number of parallel connections.

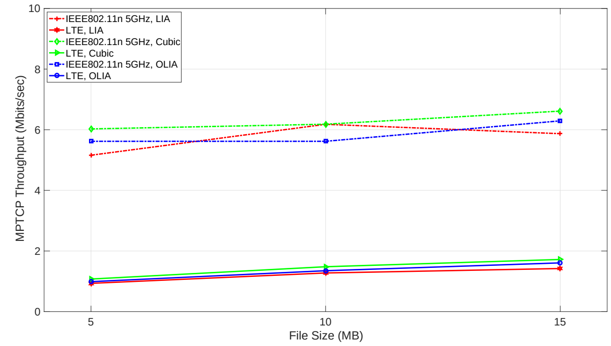
2) Data Volume: Fig. 6 shows the throughput performance of MPTCP with different file sizes for different congestion control algorithms for Setup I and Setup II. Setup III results are omitted as they yield no different conclusion. The receiver buffer size is set at 1000 KB.

Fig. 6(a) reveals the throughput results of Setup I with the number of connections is set to 5. The figure clearly shows that the LTE link throughput increases with the increasing the file size for all algorithms. This is mainly due to the fact that more data is available to fill the calculated congestion window since the WiFi subflow is established first. For WiFi links, Cubic shows a slight increase from 5 to 10 MB followed by a larger increase at 15 MB. It is worth noting that the connection establishment overhead has a slight effect on the throughput for short-lived connection. For this reason, OLIA also shows a bigger increase at 15 MB as the overhead becomes negligible. However, LIA is affected mainly by the increased average RTT as the data volume increases since the connections established first will stay for longer competing with the other connections. Fig. 6(b) depicts the throughput results of Setup II with the number of connections is set to 10 mainly to increase the LTE link throughput. The figure shows that both LIA and OLIA throughput increases for both LTE and WiFi as the data volume is increased from 10 to 15 MB. The reason is that more data becomes available to fill their congestion windows (for WiFi subflows), which is apparently bigger than Cubic's congestion window due to low average RTT. For a larger data volume, we observe a slight throughput increase for WiFi for OLIA and Cubic driven mainly by the decrease of the connection establishment overhead. However, it is found that the amount of OLIA throughput increase over WiFi for 20 MB has accompanied a similar decrease for LTE as the amount of data left for LTE diminishes. The throughput decrease for LIA in case of WiFi as it is sensitive to RTT increase.

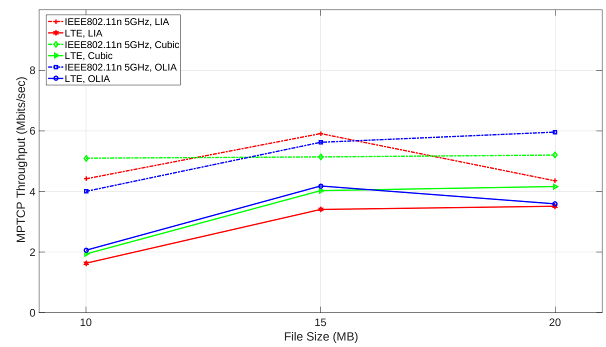
Fig. 6(c) shows the throughput results of Setup II with a lower capacity WiFi network (IEEE 802.11g). All the algorithms send slightly more traffic over LTE and also WiFi (except OLIA) as the file size increased from 10 to 15 MB. While OLIA and Cubic shows a normal fixed throughput with increasing the data volume, LIA increases the amount of traffic over LTE while decreases the WiFi share due to a better average RTT for LTE links.

In summary, a fixed throughput trend with different data volumes is achieved for MPTCP subflows of relatively long connections for Cubic and OLIA algorithms, whereas for LIA it is heavily affected by RTT. Short-lived connections do not sufficiently exploit the congestion window size for all MPTCP subflows. Thus, the excessive increase in the number of parallel connections leads to a decrease in the overall throughput as most of the traffic will be pushed on the primary subflow (i.e., the first subflow to be established) before the other subflow starts.

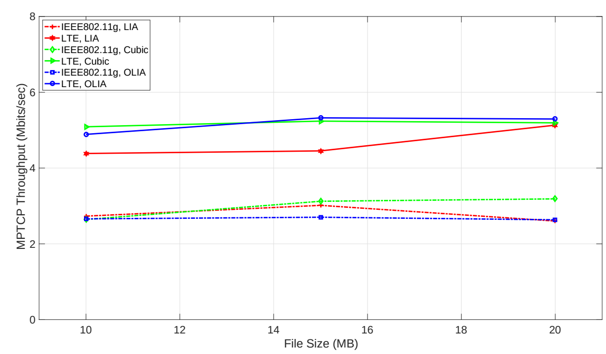
3) Receiver Buffer Size: The section addresses the effect of varying the receiver buffer size on the three congestion control algorithms understudy for all the experimental setups considered. The receiver buffer size is the limit of the congestion window size. Thus, the more the receiver buffer size increases, the more the congestion window size will be allowed to grow,



(a) Setup I.



(b) Setup II (IEEE802.11n 5GHz).



(c) Setup II (IEEE802.11g).

Fig. 6: MPTCP Throughput performance with file size.

and hence increases the amount of data sent over the network for a certain time interval. The experiment is performed by sending a 15 MB of data across 10 parallel MPTCP connections, which results in relatively short connections.

Fig. 7(a) shows the the throughput share of WiFi and LTE links for different values of the receiver buffer size for Setup I. For OLIA and Cubia algorithms, the throughput grows larger for WiFi as the buffer size is increased from 640 KB to 800 KB since this allows the algorithms to reach a congestion window size bigger than 640 KB. However, a further increase is not effective due to loss events attributed to packet collisions. On the contrary, LIA shows a decreasing trend since increasing the size of the congestion window loads the network and leads to a longer RTT. Since the LTE links of this setup already have limited capacity, the growth of the congestion window is limited, and hence no significant change is noticed in the throughput for all algorithms.

Fig. 7(b) reveals a different behavior for Cubic and OLIA for LTE links in Setup II as the available capacity of the LTE network is higher than Setup I. The throughput for Cubic apparently grows with increasing the buffer size above 640 KB then saturates as the buffer size becomes no longer a limiting factor for the size of the congestion window. For OLIA, increasing the buffer size beyond 800 KB leads the algorithm to slightly decrease the share of the LTE due to the lack of resource blocks, whereas it increases the share of WiFi. It is observed that the WiFi throughput for Cubic is slightly decreased for 1200 KB due to packet collisions. The behavior of LIA for LTE and WiFi links is anticipated as increasing the congestion window size results in a larger network traffic load and a longer RTT.

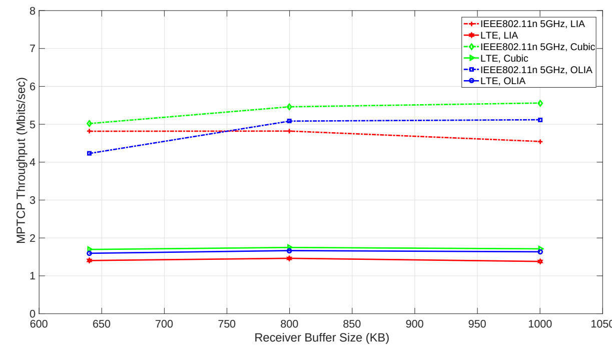
Fig. 7(c) shows a similar behavior for all the algorithms for the LTE links for Setup II and IEEE 802.11a WiFi. As in Fig. 7(b), there is a slight peak value at one of the buffer sizes followed by a decrease in the throughput due to the lack of resources, which causes segment loss or delay. Evidently, the lack of WiFi capacity causes segment loss events to happen, which negatively affects the WiFi throughput of OLIA and Cubic as the buffer size increases. On the contrary, LIA performs the best at 1000 KB since it is not affected by loss events.

Fig. 7(d) shows the results for varying the buffer size for Setup III, where a bottleneck exists for all subflows. The figure reveals that Cubic behaves the same way as in Figures 7(b) and 7(c), where a throughput peak is observed at 800 KB for both LTE and WiFi interfaces since the congestion window size of each subflow is calculated separately. However, the coupling between subflows is apparent with OLIA and LIA. As the buffer size increases, it reaches a peak for WiFi at 800 KB. A further increase leads to a decrease in the WiFi throughput share for OLIA due to packet collisions but the effect on LIA becomes significant beyond 1000 KB due to the increase in RTT. For LTE, the opposite is noticed for both LIA and OLIA since no collision exists. Apparently, LIA achieved the highest overall throughput at 1000 KB.

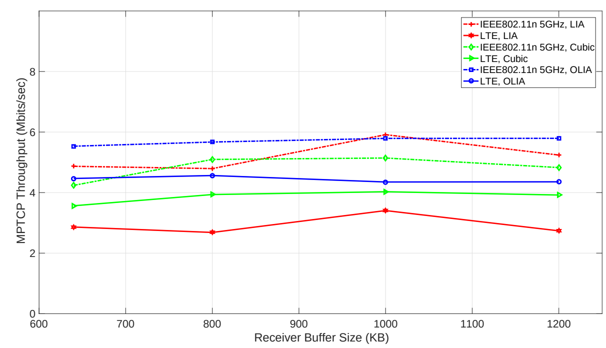
B. MPTCP performance with different WiFi standards

This section offers a comparison of the overall throughput performance of MPTCP for the congestion control algorithms understudy with different WiFi standards (IEEE 802.11a/g/n), different WiFi frequency bands (2.4 GHz and 5 GHz), and different experimental setups.

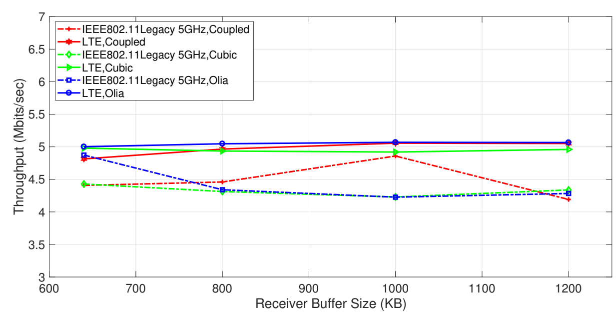
For Setup I, all the experiments are conducted by transferring a 15 MB file with a buffer size of 800 KB over 5 parallel MPTCP connections, which results in a relatively short flows (3 MB per connection). Fig. 8 shows the overall MPTCP throughput for IEEE 802.11a and IEEE 802.11n operating in the 2.4 GHz and 5 GHz bands. Since Setup I represents a case where the LTE network is somewhat loaded, the MPTCP performance for different WiFi standards is affected by the available capacity difference between the LTE and



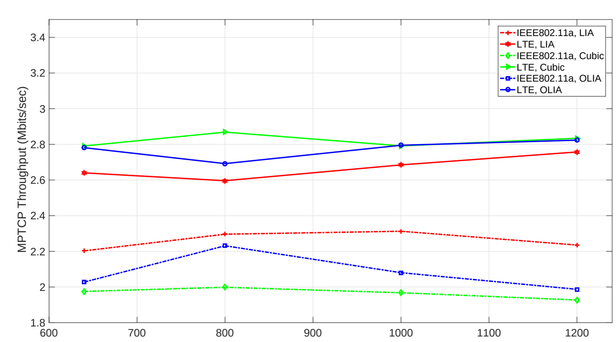
(a) Setup I.



(b) Setup II (IEEE802.11n 5GHz).



(c) Setup II (IEEE802.11a).



(d) Setup III.

Fig. 7: MPTCP Throughput with receiver buffer size.

WiFi interfaces. As the figure shows, Cubic and OLIA exhibit similar performance for IEEE 802.11a, which offers a close per-node data rate to the LTE interface. However, a slightly better performance is observed with LIA as it depends only on RTT. As the WiFi available capacity increases, in case of IEEE 802.11n, LIA and Cubic outperform OLIA, which takes into account RTT and also segment loss events due to interference in the 2.4 GHz band. The decrease in RTT and interference leads to a larger available capacity for IEEE 802.11n (5 GHz band), which, in turn, results in a larger MPTCP throughput for all algorithms compared with the previous cases. However, Cubic shows the best performance followed by OLIA since their congestion window growth function depends on the frequency of segment loss events.

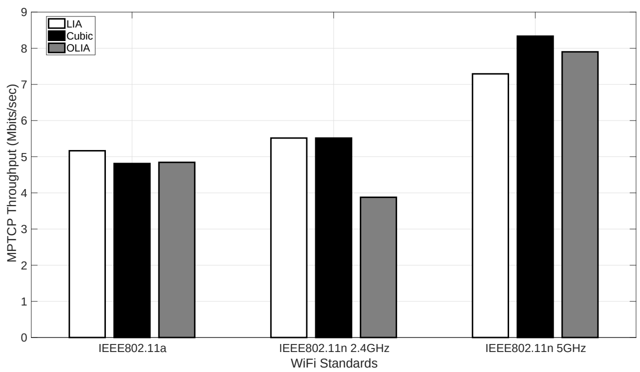


Fig. 8: MPTCP performance with different WiFi standards and congestion control algorithms for Setup I.

Furthermore, Fig. 9 compares the overall MPTCP throughput achieved by Setup II and Setup III for different congestion control algorithms and different WiFi standards. It can be readily seen from the figure that the effect of the bottleneck link in Setup III negatively impacts the MPTCP throughput. It shows also that both OLIA and LIA perform better than Cubic when a bottleneck link exists as their congestion window growth functions rely on the status of all subflows in contrast to Cubic. The figure also reveals that Cubic performs better than LIA and OLIA when the available capacity of the LTE interface is significantly greater than the WiFi one such as in the case of Setup II with IEEE 802.11g.

C. Performance of multi-homed and single-homed connections

This section highlights the gain achieved by using MPTCP with dual-homed connections over single-homed connections. The performance of Cubic, LIA, and OLIA over a single interface (LTE or WiFi) is investigated first. Based on this investigation, the MPTCP performance gain is studied for one of the algorithms.

Fig. 10 shows the results of transferring a 15 MB file over 5 parallel MPTCP connections with 1000 KB buffer size using single interface (either LTE or WiFi). The figure reveals that the three algorithms exhibit similar performance with LTE. In case of using only WiFi, Cubic slightly performs better than LIA, which has a little higher throughput than OLIA. This is

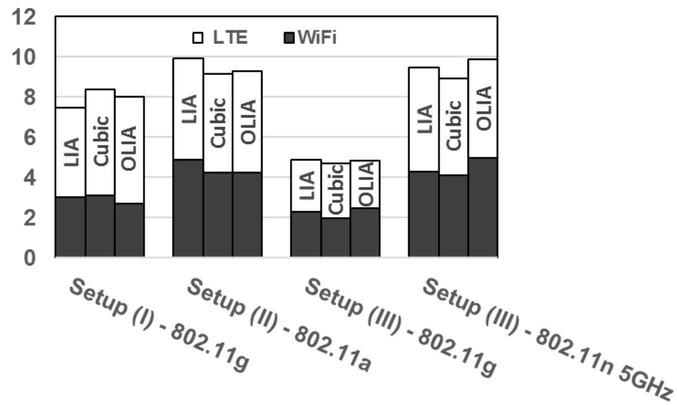


Fig. 9: MPTCP throughput (Mbits/s) with different congestion control algorithms for Setup II and Setup III.

mainly due to packet collisions, which relatively increases the RTT and packet loss events.

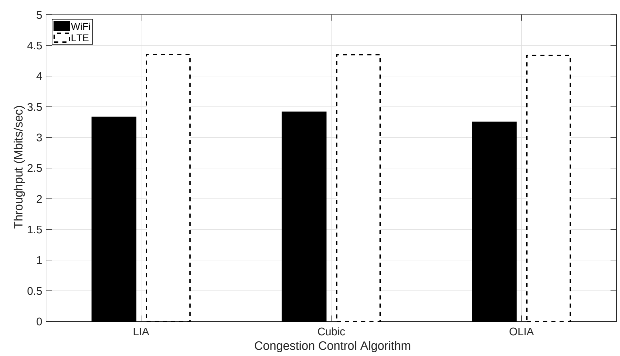


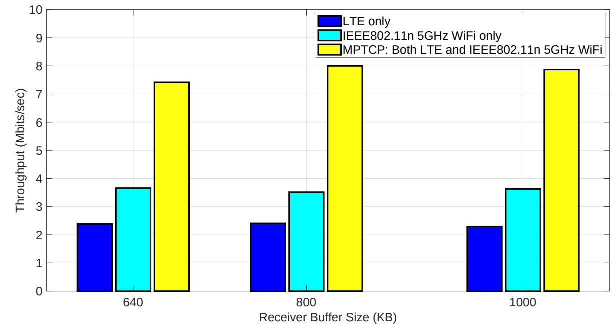
Fig. 10: MPTCP throughput with different congestion control algorithms for single-homed nodes.

Fig. 11 shows again a comparison between the throughput performance of single-homed (WiFi or LTE) nodes and dual-homed ones. The experimental results introduced in this figure are obtained by performing experiments using Setup I with WiFi links operating in the 5 GHz range using IEEE 802.11n. The employed MPTCP algorithm is LIA since it either performs similar or outperforms OLIA over single-homed connections as shown in Fig. 10. Five parallel connections are used for each node in each experiment unless otherwise specified. Three sets of experiments are performed as in the sequel.

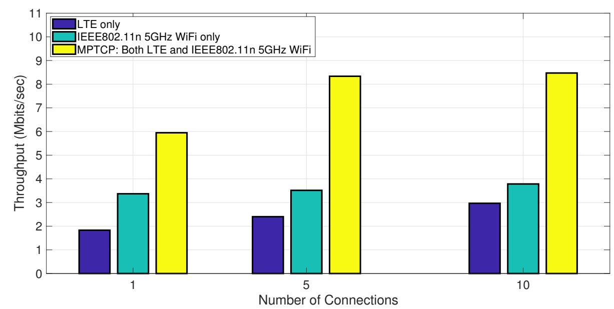
The first set investigates the throughput gain of dual-homed nodes versus single-homed LTE and WiFi nodes with different receiver buffer size values. Similar to the behavior observed in Section VI-A3, WiFi links achieve the highest throughput when the buffer size is 640 KB, as Fig. 11(a) reveals, then the throughput starts to slightly decrease. Similar behavior can be also observed with LTE links but the highest throughput is achieved with 800 KB buffer size. Apparently, the MPTCP throughput performance over dual-interface is significantly higher than the throughput performance of single-homed LTE and WiFi interfaces.

Fig. 11(b) shows that increasing the number of parallel TCP connections leads to increasing the throughput performance of single-homed nodes either they use WiFi or LTE. A similar effect is observed with MPTCP over dual-homed parallel connections as discussed in Section VI-A1. However, the MPTCP throughput performance is notably higher than the throughput of a single interface.

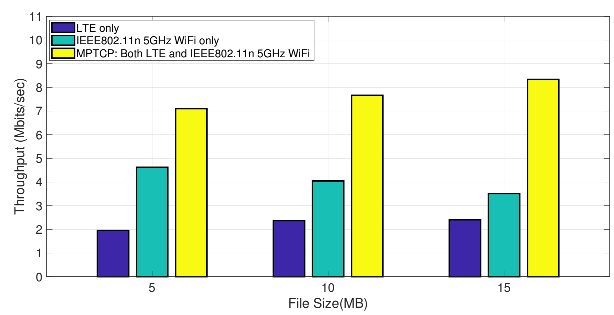
Fig. 11(c) depicts the throughput performance of single-homed and dual-homed nodes with different file sizes. Different from the previous sets of experiments, the figure reveals that WiFi throughput follows a dissimilar trend compared with LTE. Since each file is transferred over five parallel connections, a small file size results in a low amount of data per connection, this leads to less packet collisions as the connections established first finish before the connections subsequently start. Low number of packet collisions leads to a higher throughput for small file sizes. In case of large file sizes, each connection stays for a longer time, which increases the packet collision probability and decreases the throughput. For LTE, increasing the file size from 5 to 10 MB leads to the availability of more data per connection, which leads to a slight increase in the throughput as LTE links do not suffer from packet collisions. As with the other experiment sets, the MPTCP shows a significantly higher throughput performance specially with transmitting large files.



(a) Throughput performance for with receiver buffer size.



(b) Throughput performance with number of parallel connections.



(c) Throughput performance with file size.

Fig. 11: Throughput of single-homed WiFi, single-homed LTE, and multihomed nodes.

VII. Concluding Remarks

The paper gives some insights about the performance of MPTCP congestion control algorithms when operating over two radio access technologies with fundamentally different MAC mechanisms such as WiFi and LTE. The paper experimentally investigates the throughput behavior of LIA and OLIA algorithms, which are designed with mulipath in mind, in addition to the Cubic algorithm when it operates in a multipath setting. The throughput behavior is observed with varying factors, namely, the number of parallel connections, the data volume, and the receiver buffer size.

Different scenarios are experimentally studied via three laboratory setups. The first setup emulates a scenario where the LTE available uplink capacity is significantly less than the WiFi network capacity. It is observed that OLIA outperforms the other algorithms when transferring the data over one single connection. However, Cubic is the best performer with increasing the number of connections or when the data volume per connection decreases (short-lived flows). The second setup emulates a comparable per-node capacity for LTE and WiFi networks. The observations reveal that Cubic performs similar to OLIA if a single connection is used to transfer the data volume. However, when increasing the number of connections or having relatively short-lived flows, OLIA outperforms all the algorithms except for significantly small data sizes, where Cubic performs slightly better than OLIA. The third setup addresses the case where the available LTE capacity per node is higher than WiFi. In this case, Cubic performs better than OLIA and LIA when the flows are short-lived or when the number of connection is relatively high for the same amount of transferred data. In addition, Cubic also outperforms the other two algorithms in case the amount of data sent per connection is large or if a single connection is used.

However, it is noticed that Cubic performs the worst whenever all the subflows have to pass through a bottleneck link. In this case, OLIA is generally performing better than or at least similar to LIA regardless of the sending node interface type (WiFi or LTE) whose capacity is larger. Furthermore, the MPTCP performance gain over single-homed connections is notable. The throughput of single-homed nodes is found to exhibit similar behavior with varying the aforementioned factors except with WiFi, which shows a better throughput when the amount of data per connection is small due to less packet collisions.

VIII. Acknowledgments

This research was made possible by the UAE University UPAR grant number 31N226.

References

[1] A. Gupta and R. K. Jha, "A Survey of 5G Network: Architecture and Emerging Technologies," *IEEE Access*, vol. 3, pp. 1206–1232, 2015.

[2] I. F. Akyildiz, S. Nie, S.-C. Lin, and M. Chandrasekaran, "5G roadmap: 10 key enabling technologies," *Computer Networks*, vol. 106, pp. 17–48, 2016.

[3] M. Peng, Y. Li, Z. Zhao, and C. Wang, "System architecture and key technologies for 5G heterogeneous cloud radio access networks," *IEEE Network*, vol. 29, no. 2, pp. 6–14, 2015.

[4] A. Ravanshid, P. Rost, D. S. Michalopoulos, V. V. Phan, H. Bakker, D. Aziz, S. Tayade, H. D. Schotten, S. Wong, and O. Holland, "Multi-connectivity functional architectures in 5G," in *2016 IEEE International Conference on Communications Workshops (ICC)*, pp. 187–192, May 2016.

[5] J. Iyengar, C. Raiciu, S. Barre, M. J. Handley, and A. Ford, "Architectural Guidelines for Multipath TCP Development." RFC 6182, Mar. 2011.

[6] C. Raiciu, M. J. Handley, and D. Wischik, "Coupled Congestion Control for Multipath Transport Protocols." RFC 6356, Oct. 2011.

[7] R. Khalili, N. G. Gast, M. Popovi, and J.-Y. L. Boudec, "Opportunistic Linked-Increases Congestion Control Algorithm for MPTCP," Internet Draft draft-khalili-mptcp-congestion-control-05, IETF, Individual Submission, July 2014.

[8] S. Ha, I. Rhee, and L. Xu, "CUBIC: A New TCP-friendly High-speed TCP Variant," *SIGOPS Oper. Syst. Rev.*, vol. 42, pp. 64–74, July 2008.

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

[10] D. Wischik, C. Raiciu, A. Greenhalgh, and M. Handley, "Design, Implementation and Evaluation of Congestion Control for Multipath TCP," in *Proceedings of the 8th USENIX Conference on Networked Systems Design and Implementation*, NSDI'11, (Berkeley, CA, USA), pp. 99–112, USENIX Association, 2011.

[11] C. Raiciu, C. Paasch, S. Barre, A. Ford, M. Honda, F. Duchene, O. Bonaventure, and M. Handley, "How Hard Can It Be? Designing and Implementing a Deployable Multipath TCP," in *9th USENIX Symposium on Networked Systems Design and Implementation (NSDI 12)*, (San Jose, CA), pp. 399–412, USENIX Association, 2012.

[12] P. L. Eardley, "Survey of MPTCP Implementations," Internet Draft draft-eardley-mptcp-implementations-survey-02, IETF, Individual Submission, July 2013.

[13] J. Hashimoto and T. Shigeyasu, "Performance evaluation of MPTCP under heterogeneous channel characteristics," in *2015 IEEE 8th International Workshop on Computational Intelligence and Applications (IWCIA)*, pp. 137–141, Nov 2015.

[14] B. Arzani, A. Gurney, S. Cheng, R. Guerin, and B. T. Loo, "Deconstructing MPTCP Performance," in *2014 IEEE 22nd International Conference on Network Protocols*, pp. 269–274, Oct 2014.

[15] B. Arzani, A. Gurney, S. Cheng, R. Guerin, and B. T. Loo, "Impact of Path Characteristics and Scheduling Policies on MPTCP Performance," in *2014 28th International Conference on Advanced Information Networking and Applications Workshops*, pp. 743–748, May 2014.

[16] J. A. Cordero, "Multi-path TCP performance evaluation in dual-homed (wired/wireless) devices," *Journal of Network and Computer Applications*, vol. 70, pp. 131–139, 2016.

[17] A. Alheid, D. Kaleshi, and A. Doufexi, "Performance Evaluation of MPTCP in Indoor Heterogeneous Networks," in *Proceedings of the 2014 First International Conference on Systems Informatics, Modelling and Simulation*, SIMS '14, (Washington, DC, USA), pp. 213–218, IEEE Computer Society, 2014.

[18] Y.-C. Chen, Y.-s. Lim, R. J. Gibbens, E. M. Nahum, R. Khalili, and D. Towsley, "A Measurement-based Study of MultiPath TCP Performance over Wireless Networks," in *Proceedings of the 2013 Conference on Internet Measurement Conference*, IMC '13, (New York, NY, USA), pp. 455–468, ACM, 2013.

[19] E. Blanton, D. V. Paxson, and M. Allman, "TCP Congestion Control." RFC 5681, Sept. 2009.

[20] "LTE Network," in *https://www.amarisoft.com/*.

[21] "3rd Generation Partnership Project (3GPP)," in *http://www.3gpp.org/*.

[22] T. Khalifa, A. Abdrabou, K. Naik, M. Alsabaan, A. Nayak, and N. Goel, "Split- and aggregated-transmission control protocol (sa-tcp) for smart power grid," *IEEE Transactions on Smart Grid*, vol. 5, pp. 381–391, Jan 2014.

[23] Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Further Higher-Speed Physical Layer Extension in the 2.4 GHz Band, *IEEE Std 802.11g/D1.1*, 2001.