MPTCP-meLearning: A Multi-Expert Learning-Based MPTCP Extension to Enhance Multipathing Robustness against Network Attacks

Yuanlong CAO^{†a)}, Ruiwen JI[†], Lejun JI[†], Nonmembers, Xun SHAO^{††b)}, Member, Gang LEI[†], and Hao WANG[†], Nonmembers

SUMMARY With multiple network interfaces are being widely equipped in modern mobile devices, the Multipath TCP (MPTCP) is increasingly becoming the preferred transport technique since it can use multiple network interfaces simultaneously to spread the data across multiple network paths for throughput improvement. However, the MPTCP performance can be seriously affected by the use of a poor-performing path in multipath transmission, especially in the presence of network attacks, in which an MPTCP path would abruptly and frequently become underperforming caused by attacks. In this paper, we propose a multi-expert Learning-based MPTCP variant, called MPTCP-meLearning, to enhance MPTCP performance robustness against network attacks. MPTCP-meLearning introduces a new kind of predictor to possibly achieve better quality prediction accuracy for each of multiple paths, by leveraging a group of representative formula-based predictors. MPTCP-meLearning includes a novel mechanism to intelligently manage multiple paths in order to possibly mitigate the out-of-order reception and receive buffer blocking problems. Experimental results demonstrate that MPTCP-meLearning can achieve better transmission performance and quality of service than the baseline MPTCP scheme.

key words: multipath TCP, out-of-order data reception, throughput prediction, multipath management

1. Introduction

In recent years, wireless networks and technologies have experienced a period of rapid growth. The large-scale distribution of wireless network infrastructures provides a mobile Internet user with ubiquitous and instant network connectivity [1], [2]. Moreover, driven by the rapid advances in various wireless access technologies, modern mobile devices (i.e., smartphones, tablets, and other handheld computers) embedded with multiple wireless network interfaces and configured with several IP addresses (a.k.a. multihoming) are becoming more and more common [3]–[7]. With the multihoming feature, these mobile devices can access and download data from the Internet by using multiple network links simultaneously. For example, the Apple's iPhone

Fig. 1 Multipath TCP communication: a simple scenario.

series (i.e., iPhone and iPad) [8] enable “multipath transfer” feature, which combines both the Wi-Fi interface and the cellular interface to optimize network connection. Besides, the Huawei's flagship smartphones (i.e., Mate 40 and Mate 40 Pro) [9] and the Samsung Galaxy lineups (i.e., S9 and S9 Plus) [10] can offer the multipath connectivity feature for the downloading of large files, by simultaneously using both the Wi-Fi network link and the cellular network link. With the increasingly widespread application of multi-homed mobile devices, the Multipath Transmission Control Protocol (MPTCP) [11], [12] recently has attracted extensive attention and become a hot topic in the research field of Internet communication technologies.

The MPTCP is an emerging transport technique towards enabling the concurrent use of multiple network interfaces to spread the data across several independent end-to-end paths [13]. Figure 1 presents a basic scenario of MPTCP communication, in which a multi-homed mobile device is receiving the data from a remote server by simultaneously using the Wi-Fi link, the cellular link, and the Bluetooth link, supported by the MPTCP technology. In this case, the multi-homed mobile device can aggregate the bandwidth of the three network links to increase their throughput performance, enhance the resilience of connectivity, and max-

Manuscript received February 9, 2021.

Manuscript revised May 7, 2021.

Manuscript publicized July 8, 2021.

[†]The authors are with School of Software, Jiangxi Normal University, Nanchang 330022, China.

^{††}The author is with School of Regional Innovation and Social Design Engineering, Kitami Institute of Technology, Kitami-shi, 090-0801 Japan.

a) E-mail: ylcao@jxnu.edu.cn (Corresponding author)

b) E-mail: x-shao@ieee.org (Corresponding author)

DOI: 10.1587/transinf.2021NGP0009

Fig. 2 MPTCP overview

imimize resource usage. In addition to the promising multipath transmission feature, the backwards compatibility with the current Internet applications is another important feature that makes a great contribution to the success of the MPTCP. As an extension of Transmission Control Protocol (TCP), MPTCP inherits the standard socket application programming interfaces (APIs) from the TCP. An MPTCP connection actually consists of a set of TCP connections (known as subflows) [14], [15], and each TCP connection (subflow) independently performs the transmission of data over an individual path [16], as shown in Fig. 2. Correspondingly, MPTCP can be backwards compatible with TCP. With its attractive features of multipathing and backward compatibility, the MPTCP is regarded to be the core transport technique of future Internet [17]–[20].

Although applying the promising MPTCP technology towards effective concurrent multipath transmission introduces numerous potential advantages [21]–[24], the out-of-order (OFO) data arrival in MPTCP is a tricky and inevitable problem because asymmetric MPTCP transmission paths are likely to have different quality of service (QoS)-related transmission parameters (e.g., available bandwidth, packet loss, propagation delay and so on) [25]–[27]. Especially when MPTCP suffers from malicious network attacks, the OFO data arrival phenomenon can be problematic since an attacked path within the MPTCP connection generally has a large transmission quality difference compared with other MPTCP paths [28]–[32]. The most related researches demonstrate that predicting the expected transmission quality of each path, and then preventing the usage of a poor-quality path in the multipath transmission is an effective approach to alleviate the OFO data arrival problem in MPTCP [33].

The most widely way to predict transmission quality for each MPTCP path is to model the corresponding TCP connection behavior mathematically and build a formula-based throughput predictor according to the TCP connection's own characteristics. However, a single way to use a certain kind of formula-based approach as throughput predictor for an MPTCP path (a TCP connection) that is undergoing network attacks can make the prediction accuracy low. This is because network attacks can induce serious situations and intensive challenges such as abrupt and frequent transmission fluctuations and packet losses that have significant side-effects on the performance of transmissions.

In MPTCP, although each of paths independently performs the transmission of data over an individual path, these paths actually affect each other because of cross-path interactions between them. For instance, if a path experiences transmission interruption caused by some kind of malicious network attack, the other paths within the MPTCP connection are likewise disturbed in transmissions. Various forms of malicious network attacks, such as the Cross-Path Inference Attacks [34], the Man-in-the-Middle attacks [31], [35], the random/selective cyber attacks [36], the low-rate distributed denial-of-service (LDDoS) attacks [29], the connection hijack attack [32], and the traffic diversion attack [32], have been reported to exploit the vulnerabilities (e.g., cross-path interactions between paths) in MPTCP to cause performance degradation of MPTCP. Nevertheless, there is no exact specification that can be used for preventing the use of attacked paths in MPTCP [37], [38].

In this paper, we propose MPTCP-*meLearning*, a multi-expert Learning-based MPTCP extension to enhance MPTCP performance robustness against network attacks. The design objectives of MPTCP-*meLearning* are (i) to possibly achieve better transmission quality prediction accuracy for each MPTCP path, even when MPTCP experiences an unexpected transmission situation (e.g., a network attack situation); and (ii) to adaptively choose a subset of high-performing paths for data transmission while renouncing the use of a relative low-performing path in multipath transmission if necessary. We evaluate the proposed MPTCP-*meLearning* scheme by simulations. The results demonstrate that MPTCP-*meLearning* can achieve better transmission performance and QoS than the baseline MPTCP in the presence of network attacks. Most notably, MPTCP-*meLearning* makes valuable and constructive contributions against the literature in this area, in the following two main aspects:

- i) It presents a new kind of predictor to possibly achieve better quality prediction accuracy for each of multiple paths, even when the path experiences bursty losses or abrupt fluctuations caused by some kinds of network attacks, by leveraging a group of representative formula-based predictors.
- ii) It designs a novel mechanism to intelligently manage multiple paths in MPTCP in order to possibly mitigate the OFO reception and receive buffer blocking problems caused by the large transmission quality differences among the asymmetric paths.

The remainder of the paper is organized as follows. Section 2 introduces the detailed designs of the *meLearning* Station and *meLearning* Scheduler. Section 3 compares and analysis the performance of the baseline MPTCP and MPTCP-*meLearning*. Section 4 concludes the paper and gives our future work.

2. MPTCP-*meLearning* Detail Design

Figure 3 presents the architecture of the proposed MPTCP-

Fig. 3 The architecture of MPTCP-meLearning

Table 1 Basic notations.

Notations	Descriptions
F_i	A formula-based TCP throughput prediction model
$\bar{F}_i$	The output generated by the i^{th} prediction model
W	The congestion window size
W_{max}	The maximum congestion window size
MSS	Maximum segment size
RTT	Round-trip time
RTO	Retransmission Timeout
L	The packet loss rate measured on d_i
$\bar{Q}$	The transmission quality of an MPTCP path

meLearning solution, which involves an MPTCP sender, an MPTCP receiver, and n network paths. As a variant of MPTCP, the MPTCP-*meLearning* extends the baseline MPTCP at the sender side, by using a new Multi-Expert Learning Station (hereafter referred to as *meLearning* Station) and a novel Multi-Expert Learning-based Scheduler (hereafter referred to as *meLearning* Scheduler). While at the MPTCP-*meLearning* receiver side, all the modules are directly inherited from the modules of the baseline MPTCP receiver, which means the MPTCP-*meLearning* receiver just accomplishes the same operations as the baseline MPTCP receiver does; for example, it performs packet reordering and acknowledgement feedback if any MPTCP segment arrived. The following are the detailed designs of the *meLearning* Station and *meLearning* Scheduler. For convenience, we briefly list the basic notations, as shown in Table 1. These notations are helpful to read and understand the MPTCP-*meLearning* Detail Design section.

2.1 *meLearning* Station

The *meLearning* Station is proposed to possibly achieve better throughput prediction accuracy, by aggregating a group of representative formula-based predictors to predict the ex-

Fig. 4 The architecture of *meLearning* Station

pected throughput for each MPTCP path, even MPTCP experiences an unexpected transmission situation (e.g., a network attack situation). Figure 4 illustrates the architecture of the proposed *meLearning* Station. As the figure shows, the *meLearning* Station contains an Expert System (dubbed “e-System”) which consists of m formula-based predictors, and each predictor represents a formula-based TCP throughput prediction model. Assume that path d_i is one of transmission paths in an MPTCP connection, and $(Q_1, Q_2, \dots, Q_n)$ are the n network parameters of d_i (e.g., end-to-end delay, packet loss and other QoS-related network parameters), a formula-based TCP throughput prediction model (denoted as F_i) used in the *meLearning* Station can be mathematically represented by a function of n network parameters of d_i , which can be expressed by

$$F_i(Q_1, Q_2, \dots, Q_n), \quad s.t. 1 \leq i \leq m. \quad (1)$$

At present, there are so many formula-based throughput prediction models in the research field of TCP. In practice, all those solid prediction models can be aggregated to build up an expert system in the *meLearning* Station. In this paper, the three most popular TCP throughput prediction models: the Veno model [39], the Mathis model [40], and the Padhye model [41], are included in the e-System and acted as built-in experts to predict each TCP connection (each MPTCP path). This is because that to the best of our knowledge, most of the current prediction models are derived from one of the three well-known TCP throughput

prediction models. In the *meLearning* Station, each of the three models performs the throughput prediction for the path d_i independently, by leveraging the n network parameters obtained on d_i .

$$\overline{F}_i = F_i(Q_1, Q_2, \dots, Q_n), \quad (2)$$

where $\overline{F}_i$ ($1 \leq i \leq m$) are the output generated by the i^{th} prediction model.

Let *const* be a constant, and let W , W_{max} , MSS , RTT , RTO , and L be the congestion window size, the maximum congestion window size, maximum segment size, round-trip time, Retransmission Timeout and packet loss rate measured on d_i , the three early TCP throughput prediction models: the Venio model [39], the Mathis model [40], and the Padhye model [41], can be expressed by using the following Eqs. (3), (4) and (5), respectively.

$$F_1 = W \times \frac{1}{RTT}, \quad (3)$$

$$F_2 = \frac{MSS}{RTT} \times \frac{const}{\sqrt{L}}, \quad (4)$$

$$F_3 = \min \left(\frac{W_{max}}{RTT}, \frac{1}{RTT \sqrt{\frac{2bL}{3} + RTO \min(1, 3 \sqrt{\frac{3bL}{8}}) L(1+32L^2)}} \right). \quad (5)$$

When all the above three models complete the throughput calculation for the MPTCP path d_i , the *meLearning* Station then synthesizes the prediction results using the weighted sum of the output of each prediction model as:

$$\begin{cases} Q(F_1, F_2, \dots, F_m) = \frac{1}{m} \times \sum_{i=1}^m \alpha_i \overline{F}_i; \\ \overline{Q} = Q(F_1, F_2, \dots, F_m), \end{cases} \quad (6)$$

where $m=1,2,3$ is the index of prediction models, $\alpha_i \in [0,1]$ are the variable weights and each one with a fairness value $\frac{1}{2}$ by default, $Q(F_1, F_2, \dots, F_m)$ is a simple additive weighting function, and $\overline{Q}$ is the final synthesized output of the *meLearning* Station. The path with the largest $\overline{Q}$ value has the highest expected throughput.

By using the combination of the three classic prediction models as an expert team, and by smoothing the output of each prediction model, the *meLearning* Station can possibly acquire a more stable and accurate prediction result for each MPTCP path even in presence of bursty losses and abrupt fluctuations caused by some kinds of network attacks. In this regard, the *meLearning* Station is actually a new kind of predictor that has a great potential to identify and distinguish the transmission quality of each path within the MPTCP connection. Specifically, the *meLearning* Station will periodically (per one RTT) compute the transmission quality for each MPTCP path, and an MPTCP path with the largest $\overline{Q}$ value has the highest best quality, and vice versa.

2.2 *meLearning* Scheduler

The *meLearning* Scheduler is an extension to the baseline

MPTCP scheduler with the addition of multipath management operations. More specifically, the *meLearning* Scheduler is responsible for spreading data across multiple independent transmission paths in an MPTCP connection, like the baseline MPTCP scheduler does. Besides, it is designed to intelligently manage multiple paths to possibly mitigate the OFO data arrival problem in MPTCP, by fully exploiting the advantages of the *meLearning* Station. This is vitally important for MPTCP because if an MPTCP receiver needs to maintain a large number of OFO data in its finite buffer space for reordering, like it was previously mentioned, the MPTCP is more likely to experience serious receive buffer blocking, which causes severe performance degradation of multipath transmission.

Let's assume that there are κ paths ($d_1, d_2, \dots, d_\kappa$) in an MPTCP connection, and let d_{group} , d_{group}^{active} , and $d_{group}^{deactive}$ be a group with all the κ paths, a group with the active paths, and a group with the deactivated paths in MPTCP, respectively, in which ($d_{group}^{active}, d_{group}^{deactive} \in d_{group}$) and ($d_{group}^{active} \cap d_{group}^{deactive} = \phi$). To possibly reduce OFO data reception and avoid the receiver buffer blocking in MPTCP, the *meLearning* Scheduler discriminates the transmission quality for each of the κ paths according to the output of the *meLearning* Station, and thereby adaptively manages these paths in multipath transmission, by following the strategies below:

- i) The *meLearning* Scheduler only uses the paths within the group d_{group}^{active} to transmit the MPTCP data chunks;
- ii) The *meLearning* Scheduler can renounce the use of a poor-performing path for data transmission if a serious receive buffer blocking occurs;
- iii) The *meLearning* Scheduler also can reactivate a path within the group $d_{group}^{deactive}$ for data transmission conditionally.

The design objectives of the above strategies used for the *meLearning* Scheduler are to choose a group of suitable transmission paths for effective multipath transmission while preventing the usage of a poor-performing path (e.g., a path that is under attack) in multipath transmission if necessary. The multipath management operations, as well as the data scheduling operations of the *meLearning* Scheduler, each step of which is described in more detail, are as follows:

- i) Concurrently distributing the MPTCP data chunks across the paths within the group d_{group}^{active} , and periodically (per one RTT) sending probe data to the paths within the group $d_{group}^{deactive}$;
- ii) Sorting all the paths within the group d_{group}^{active} in rank order of worst to best according to their own $\overline{Q}$ values, supported by the *meLearning* Station;
- iii) Deactivating the path with the lowest value of $\overline{Q}$ for data transmission when a severe receive buffer blocking occurs, and then moving this path from the group d_{group}^{active} to the group $d_{group}^{deactive}$;
- iv) If any of the paths within the group $d_{group}^{deactive}$ has the $\overline{Q}$ value greater than or equal to a specific threshold value,

Table 2 Symbols presented in the Algorithm 1.

Notations	Descriptions
$\bar{Q}$	The transmission quality of an MPTCP path
d_{group}	A group with all the available paths in MPTCP
$d_{\text{group}}^{\text{active}}$	A group with the active paths
$d_{\text{group}}^{\text{deactive}}$	A group with the deactivated paths
$d_{\text{group}[\ell]}^{\text{active}}$	The ℓ^{th} path within the group $d_{\text{group}}^{\text{active}}$
$d_{\text{group}[\gamma]}^{\text{deactive}}$	The γ^{th} path within the group $d_{\text{group}}^{\text{deactive}}$
$\text{count}(d_{\text{group}}^{\text{active}})$	The number of paths in the group $d_{\text{group}}^{\text{active}}$
$\text{count}(d_{\text{group}}^{\text{deactive}})$	The number of paths in the group $d_{\text{group}}^{\text{deactive}}$

moving this path from the group $d_{\text{group}}^{\text{deactive}}$ to the group $d_{\text{group}}^{\text{active}}$, and then reactivate this path for data delivery.

For the above step iv), we denote by d_j the j^{th} path in $d_{\text{group}}^{\text{deactive}}$, by $\bar{Q}_{d_j}$ the transmission quality of the path d_j , and by $\text{count}(d_{\text{group}}^{\text{active}})$ the number of paths in $d_{\text{group}}^{\text{active}}$. To simplify the calculation procedure, if the quality of path d_j , $\bar{Q}_{d_j}$, is not lower than the cumulative average quality of all the paths within the group $d_{\text{group}}^{\text{active}}$, this path can be reactivated by the *meLearning* Scheduler for the transmission of MPTCP data chunks. can be expressed by using the following mathematic model,

$$\bar{Q}_{d_j} - \frac{1}{\text{count}(d_{\text{group}}^{\text{active}})} \times \sum_{\omega=1}^{\text{count}(d_{\text{group}}^{\text{active}})} \bar{Q}_{d_\omega} \geq \mathbb{C}, \quad (7)$$

where $\mathbb{C}$ is a constant and its value can be changed with the actual situation. Here, the value of $\mathbb{C}$ is simply set to zero. Such a way to reactivate a path for data distribution can maximize the utilization of network resources while possibly ensuring all the paths in multipath transmission have little difference in terms of transmission quality.

The main pseudo-code of the MPTCP-*meLearning* with the operations enabled by the *meLearning* Station and *meLearning* Scheduler is presented in the Algorithm 1. For convenience, we present a list of important symbols in the Table 2 that are helpful for the reading and understanding of the algorithm.

3. Performance Evaluation

3.1 Simulation Topology

The performance evaluation has been carried out on the network simulator version 2.35 [42], which is known simply as NS-2 and includes the baseline MPTCP module [43]. The simulation topology is presented in Fig. 5, which includes an MPTCP sender and an MPTCP receiver. The two MPTCP endpoints are connected by three asymmetric and independent paths, which are path #1 that is configured with

Algorithm 1 The operations in MPTCP-*meLearning*

Initialization:

$\mathbb{C}=0$;

$d_{\text{group}}^{\text{active}}=d_{\text{group}}$;

$d_{\text{group}}^{\text{deactive}}=\phi$;

1: schedule MPTCP data chunks across the paths in $d_{\text{group}}^{\text{active}}$;

2: **for** ($\ell=1, \ell \leq \text{count}(d_{\text{group}}^{\text{active}}), \ell++$) **do**

3: compute the value of $\bar{Q}$ for $d_{\text{group}[\ell]}^{\text{active}}$ by Eqs. (1)-(6);

4: **if** receive buffer blocking occurs **then**

5: sort $d_{\text{group}[\ell]}^{\text{active}}$ in ascending order of $\bar{Q}$;

6: mark $d_{\text{group}[0]}^{\text{active}}$ as the path with the lowest value of $\bar{Q}$;

7: move $d_{\text{group}[0]}^{\text{active}}$ from $d_{\text{group}}^{\text{active}}$ to $d_{\text{group}}^{\text{deactive}}$;

8: **end if**

9: **end for**

10: **for** ($\gamma=1, \gamma \leq \text{count}(d_{\text{group}}^{\text{deactive}}), \gamma++$) **do**

11: compute the value of $\bar{Q}$ for $d_{\text{group}[\gamma]}^{\text{deactive}}$ by Eqs. (1)-(6);

12: **if** $\bar{Q}_{d_j} - \frac{1}{\text{count}(d_{\text{group}}^{\text{active}})} \times \sum_{\omega=1}^{\text{count}(d_{\text{group}}^{\text{active}})} \bar{Q}_{d_\omega} \geq \mathbb{C}$ **then**

13: move $d_{\text{group}[\gamma]}^{\text{deactive}}$ from $d_{\text{group}}^{\text{deactive}}$ to $d_{\text{group}}^{\text{active}}$;

14: **end if**

15: **end for**

10Mbps bandwidth and 10-20 ms propagation delay, path #2 that is configured with the same network parameters as path #1, and path #3 that is configured with 10Mbps bandwidth and 20-30 ms propagation delay.

In order to reflect the burst frame loss and high frequency in the data link layer in this experiment, the *Unified Loss* model (used to represent the distributed loss caused by wireless noise interference) and the two-state *Markov Loss* model (used to represent low-frequency continuous loss caused by wireless signal fading) are added to the access links of both path #1 and #2. We use the default queue management algorithm Droptail in NS-2. Table 2 presents other important network parameters used to configure the three paths.

Furthermore, considering that the complex behavior of the Internet cross-traffic is very important for measuring the performance of a transport protocol, each of the two paths (#1 and #2) is attached to one Variable Bit Rate (VBR) traffic generator to send VBR cross-traffic over the two paths. The selected VBR cross-flow data packet is like [29], in which 1500 bytes account for 46%, 1300 bytes account for 1.7%, 628 bytes account for 2.1%, 576 bytes account for 1.2%, and 44 bytes account for 49%. The VBR cross-traffic occupies 25% of path #1's access link bandwidth, and 50% of path #2's access link bandwidth, respectively.

Fig. 5 Simulation topology

Table 3 Path parameter configuration used in the simulation.

Network Parameters	Path #1, #2	Path #3
Access link bandwidth	10Mbps	10Mbps
Access link delay	10-20ms	20-30ms
Access link queue type	Droptail	Droptail
Core network bandwidth	100Mbps	100Mbps
Core network delay	30ms	75ms
Uniform loss rate	1-3%	0%
Markov loss rate	1%	0%

In addition, to simulate a case of an MPTCP path with network attacks, the access link of path #3 is attached with ten Constant Bit Rate (CBR) traffic generators in order to generate the low-rate distributed denial of service (LDDoS) attack traffic, with considering that an LDDoS attack usually uses User Datagram Protocol (UDP) with the CBR traffic. All the ten CBR generators begin their attack at 0.1^{th} second and end at 99.9^{th} second of simulation time. The packet size of CBR is set to 1444 bytes. The sending rate of CBR packets is 1 mbps. They inject the LDDoS traffic with the same characteristics as described in our previous work [29, Eq. (1)]. The simulation time is 100 seconds.

3.2 Simulation Results

1) *Data sending and receiving times.* Comparison results of data transmission time using the baseline MPTCP and MPTCP-meLearning are shown in Fig. 6 (a), and comparison results of data reception time using the baseline MPTCP and MPTCP-meLearning are shown in Fig. 6 (b). We can intuitively understand from the two figures that the performance of data transmission and reception of the baseline MPTCP is better than that of MPTCP-meLearning in the first few seconds of simulation time. This is because that the baseline MPTCP uses all the three paths for data transmission. In MPTCP-meLearning, however, compared

Fig. 6 The comparison of data sending and receiving times

with paths #1 and #2, the quality of path #3 is different from them, so that path #3 will not be used in multipath transmission. However, after the first few seconds, the performance of MPTCP-meLearning reached a higher level than that of the baseline MPTCP. This is because, in the baseline MPTCP, path #3 is used for multipath transmission, but its performance is poor and even prone to failure, which blocks the receiver buffer and interrupts paths #1 and #2. However, in the multipath transmission of MPTCP-meLearning, only the stable paths #1 and #2 are used for the transmission of data.

2) *Out-of-order DSN.* The Out-of-Order Data Sequence Number (OFO DSN) represents the offset between the DSNs of two consecutively received MPTCP chunks, therefore, it can be calculated by the difference between the DSN of the current MPTCP chunk and that of the latest received MPTCP chunk. Nowadays, the OFO DSN is considered as a fairly good performance metric for a multipath protocol. Figure 7 shows the situation of OFO DSN under two different situations of

Fig. 7 The comparison of out-of-sequence DSN

using the baseline MPTCP and MPTCP-*meLearning*. Because the scheduler of the baseline MPTCP will allocate corresponding traffic data on all available paths, and ignoring many paths that are easy to fail will lead to rearrangement of more disordered data. Therefore, we can know from the figure that not only the baseline MPTCP produces more OFO DSNs than MPTCP-*meLearning*, but also a large amount of redundant data will be rearranged if the baseline MPTCP is used. Compared with the baseline MPTCP, MPTCP-*meLearning* can guarantee the arrival of non-disorderly traffic data because it only uses a set of suitable paths for multipath transmission. In the more quantitative results, the peak value of OFO DSN when using the baseline MPTCP is up to 3.8×10^6 , while using MPTCP-*meLearning*, it is only approximately 5.1×10^4 .

- 3) *Throughput Performance*. As shown in Fig. 8, the throughput results under the baseline MPTCP usage and MPTCP-*meLearning* usage are compared. In multipath transmission, in order to achieve a high level of throughput, MPTCP-*meLearning* will avoid using paths that are prone to failure. Besides, it can reduce the probability of blocking the receiver buffer, and it can also reduce the probability of transmission interruption of other high-performance paths. For the above reasons, MPTCP-*meLearning* can distribute traffic data on the subset of stable paths. However, if the baseline MPTCP is used, besides easily interrupted paths, data transmission will be hindered, and disordered traffic data will also be received. This undoubtedly weakens the overall throughput performance. In order to illustrate the throughput comparison better, we calculate the cumulative average throughput can be computed by averaging the total average throughput values in a total of 100 seconds of simulation time. The cumulative average throughput of the baseline MPTCP and MPTCP-*meLearning* is shown in the sub-graph in Fig. 8. It can be clearly seen that compared with the baseline MPTCP, the cumulative average throughput of

Fig. 8 The comparison of throughput

Fig. 9 The comparison of end-to-end delay

MPTCP-*meLearning* is 37.67% higher.

- 4) *End-to-end delay*. Figure 9 compares the end-to-end delay performance with the baseline MPTCP and MPTCP-*meLearning*. As stated above, in order to ensure that the MPTCP segments arrive at the receiver in an accurate order and avoid unnecessary, redundant retransmission in the interrupted (or fault-prone) path in the transmission process, the characteristics of MPTCP-*meLearning* to avoid using poor performance paths and/or fault-prone paths reduce the delay (mainly data packet rearrangement delay and data packet transmission delay). Therefore, when the total simulation time is 100 seconds when comparing the cumulative average delay of these two different transmission methods, it can be found that the cumulative average delay of MPTCP-*meLearning* is about 1.25% lower than that of the baseline MPTCP.
- 5) *Jitter comparison*. As a suitable measure to evaluate and explore the time performance of multipath transmission protocol, jitter actually refers to the variation of packet delay with heavy end-to-end traffic data transmission time. In the process of low-performance transmission, the amount of jitter may be more. As shown in Fig. 10, after comparing the jitter performance when

Fig. 10 The comparison of jitter

using the baseline MPTCP and MPTCP-*meLearning*, it can be found that the path management behavior of MPTCP-*meLearning* (i.e., the traditional “full MPTCP” mode to the “intelligent MPTCP” mode) will affect the jitter performance. Its jitter performance is occasionally lower than the baseline MPTCP, but these are acceptable. Because MPTCP-*meLearning* can adaptively select a reasonable path for transmission, the scheduler of the baseline MPTCP will receive data out of sequence and have redundant retransmission. In contrast, MPTCP-*meLearning* can achieve stable performance, while the service quality and efficiency of the baseline MPTCP are lower.

4. Conclusions and Future Works

In this paper, we present a multi-expert learning-based MPTCP variant, dubbed as MPTCP-*meLearning*, to enhance multipathing robustness against network attacks. In contrast to MPTCP, MPTCP-*meLearning* includes two additional blocks, which are the Multi-Expert Learning Station (abbreviated as *meLearning* Station) that is dedicated to possibly achieving better quality prediction accuracy for each of multiple paths in MPTCP, even when an MPTCP path experiences bursty losses or abrupt fluctuations caused by some kinds of network attacks, by leveraging a group of representative formula-based predictors, and the Multi-Expert Learning-based Scheduler (abbreviated as *meLearning* Scheduler) that is devoted to intelligently managing multiple paths in MPTCP in order to possibly mitigate the OFO reception and receive buffer blocking problems caused by the large transmission quality differences among the asymmetric paths. The simulation results reveal that MPTCP-*meLearning* can achieve better performance than the baseline MPTCP in the presence of network attacks.

Acknowledgments

This work was supported by the National Natural Science

Foundation of China (NSFC) under Grant No. 61962026, by the Natural Science Foundation of Jiangxi Province under Grant No. 20192ACBL21031, by the ROIS NII Open Collaborative Research 2021 under Grant no. 21AF03, and by the Cooperative Research Project Program of the Research Institute of Electrical Communication, Tohoku University.

References

- [1] G. Cerar, H. Yetgin, M. Mohorčič, and C. Fortuna, “Machine Learning for Wireless Link Quality Estimation: A Survey,” *IEEE Communications Surveys & Tutorials*, vol.23, no.2, pp.696–728, Jan. 2021. Early Access.
- [2] M.R. Palash and K. Chen, “MPWiFi: Synergizing MPTCP Based Simultaneous Multipath Access and WiFi Network Performance,” *IEEE Transactions on Mobile Computing*, vol.19, no.1, pp.142–158, 2020.
- [3] Z. Liu, G. Cheung, J. Chakareski, and Y. Ji, “Multiple Description Coding and Recovery of Free Viewpoint Video for Wireless Multi-Path Streaming,” *IEEE Journal of Selected Topics in Signal Processing*, vol.9, no.1, pp.151–164, 2015.
- [4] F. Song, Z. Ai, Y. Zhou, I. You, K.-K.R. Choo, and H. Zhang, “Smart Collaborative Automation for Receive Buffer Control in Multipath Industrial Networks,” *IEEE Transactions on Industrial Informatics*, vol.16, no.2, pp.1385–1394, 2020.
- [5] Z. Liu, M. Dong, H. Zhou, X. Wang, Y. Ji, and Y. Tanaka, “Device-to-device assisted video frame recovery for picocell edge users in heterogeneous networks,” *Proc. 2016 IEEE International Conference on Communications*, May 2016.
- [6] Y. Cao, D. Yu, L. Zeng, Q. Liu, F. Wu, X. Gui, and M. Huang, “Towards Efficient Parallel Multipathing: A Receiver-Centric Cross-Layer Solution to Aid Multipath TCP,” *Proc. IEEE ICPADS*, pp.1–8, Dec. 2019.
- [7] Y. Cao, L. Zeng, Q. Liu, G. Lei, M. Huang, and H. Wang, “Receiver-Assisted Partial-Reliable Multimedia Multipathing Over Multi-Homed Wireless Networks,” *IEEE Access*, vol.7, pp.177675–177689, 2019.
- [8] <https://support.apple.com/lv-lv/HT201373>, accessed Dec. 2020.
- [9] <https://support.huawei.com/enterprise/en/doc/EDOC1000162776/e03f3b43/multipath-connectivity>, accessed Jan. 2021.
- [10] <https://www.samsung.com/uk/support/mobile-devices/what-is-the-download-booster-and-how-do-i-enable-it-on-my-samsung-galaxy-alpha/>, accessed Dec. 2020.
- [11] A. Ford, C. Raiciu, M. Handley, and O. Bonaventure, “TCP Extensions for Multipath Operation With Multiple Addresses,” *IETF RFC 6824*, 2013.
- [12] A. Ford, C. Raiciu, M. Handley, O. Bonaventure, and C. Paasch, “TCP Extensions for Multipath Operation with Multiple Addresses,” *IETF RFC 8684*, 2020.
- [13] P. Ignaciuk and M. Morawski, “Discrete-Time Sliding-Mode Controllers for MPTCP Networks,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp.1–11, 2020. Early Access Article.
- [14] Y. Zhang, H. Mekky, Z.-L. Zhang, F. Hao, S. Mukherjee, and T.V. Lakshman, “SAMPO: Online subflow association for multipath TCP with partial flow records,” *Proc. IEEE INFOCOM*, pp.1–9, April 2016.
- [15] K. Xue, J. Han, H. Zhang, K. Chen, and P. Hong, “Migrating Unfairness Among Subflows in MPTCP With Network Coding for Wired-Wireless Networks,” *IEEE Transactions on Vehicular Technology*, vol.66, no.1, pp.798–809, 2017.
- [16] H. Sinky, B. Hamdaoui, and M. Guizani, “Seamless Handoffs in Wireless HetNets: Transport-Layer Challenges and Multi-Path TCP Solutions with Cross-Layer Awareness,” *IEEE Network*, vol.33, no.2, pp.195–201, 2019.
- [17] L. Li, K. Xu, T. Li, K. Zheng, C. Peng, D. Wang, X. Wang, M.

- Shen, and R. Mijumbi, "A measurement study on multi-path TCP with multiple cellular carriers on high speed rails," *Proc. ACM SIGCOMM*, pp.161–175, 2018.
- [18] S.K. Saha, S. Aggarwal, R. Pathak, D. Koutsonikolas, and J. Widmer, "MuSher: An Agile Multipath-TCP Scheduler for Dual-Band 802.11ad/ac Wireless LANs," *Proc. ACM MobiCom*, pp.1–16, 2019.
- [19] B.Y.L. Kimura, D.C.S.F. Lima, L.A. Villas, and A.A.F. Loureiro, "Interpath Contention in MultiPath TCP Disjoint Paths," *IEEE/ACM Transactions on Networking*, vol.27, no.4, pp.1387–1400, 2019.
- [20] J. Zhao, J. Liu, H. Wang, C. Xu, W. Gong, and C. Xu, "Measurement, Analysis, and Enhancement of Multipath TCP Energy Efficiency for Datacenters," *IEEE/ACM Transactions on Networking*, vol.28, no.1, pp.57–70, 2020.
- [21] M. Fukuyama, N. Yamai, S. Ohzahata, and N. Kitagawa, "Throughput Improvement of MPTCP by Selective Bicasting with Cross-Layer Control in Wireless Environment," *Proc. IEEE COMPSAC*, pp.204–209, 2018.
- [22] J. Wu, B. Cheng, M. Wang, and J. Chen, "Quality-Aware Energy Optimization in Wireless Video Communication With Multipath TCP," *IEEE/ACM Transactions on Networking*, vol.25, no.5, pp.2701–2718, 2017.
- [23] J. Wu, R. Tan, and M. Wang, "Energy-Efficient Multipath TCP for Quality-Guaranteed Video over Heterogeneous Wireless Networks," *IEEE Transactions on Multimedia*, vol.21, no.6, pp.1593–1608, 2019.
- [24] Y. Cui, L. Wang, X. Wang, H. Wang, and Y. Wang, "FMTCP: A Fountain Code-Based Multipath Transmission Control Protocol," *IEEE/ACM Transactions on Networking*, vol.23, no.2, pp.465–478, 2015.
- [25] B.Y.L. Kimura, D.C.C.F. Lima, and A.A.F. Loureiro, "Packet Scheduling in Multipath TCP: Fundamentals, Lessons, and Opportunities," *IEEE Systems Journal*, vol.15, no.1, pp.1445–1457, 2020. Early Access Article.
- [26] K. Xue, J. Han, D. Ni, W. Wei, Y. Cai, Q. Xu, and P. Hong, "DPSAF: Forward Prediction Based Dynamic Packet Scheduling and Adjusting With Feedback for Multipath TCP in Lossy Heterogeneous Networks," *IEEE Transactions on Vehicular Technology*, vol.67, no.2, pp.1521–1534, 2018.
- [27] B.Y.L. Kimura, D.C.S.F. Lima, and A.A.F. Loureiro, "Alternative Scheduling Decisions for Multipath TCP," *IEEE Communications Letters*, vol.21, no.11, pp.2412–2415, 2017.
- [28] M. Jadin, G. Tihon, O. Pereira, and O. Bonaventure, "Securing multipath TCP: Design & implementation," *Proc. IEEE INFOCOM*, pp.1–9, May 2017.
- [29] Y. Cao, F. Song, Q. Liu, M. Huang, H. Wang, and I. You, "A LDDoS-Aware Energy-Efficient Multipathing Scheme for Mobile Cloud Computing Systems," *IEEE Access*, vol.5, pp.21862–21872, 2017.
- [30] G. Noh, H. Park, H. Roh, and W. Lee, "Secure and Lightweight Subflow Establishment of Multipath-TCP," *IEEE Access*, vol.7, pp.177438–177448, 2019.
- [31] H.-D.-D. Nguyen, C.-D. Phung, S. Secci, B. Felix, and M. Nogueira, "Can MPTCP secure Internet communications from man-in-the-middle attacks?" *Proc. 13th International Conference on Network and Service Management*, Nov. 2017.
- [32] A. Munir, Z. Qian, Z. Shafiq, A. Liu, and F. Le, "Multipath TCP traffic diversion attacks and countermeasures," *Proc. IEEE ICNP*, Oct. 2017.
- [33] B.-H. Oh and J. Lee, "Feedback-Based Path Failure Detection and Buffer Blocking Protection for MPTCP," *IEEE/ACM Transactions on Networking*, vol.24, no.6, pp.3450–3461, 2016.
- [34] M.Z. Shafiq, F. Le, M. Srivatsa, and A.X. Liu, "Cross-path inference attacks on multipath TCP," *Proc. ACM Workshop on Hot Topics in Networks (HotNets)*, pp.1–7, Nov. 2013.
- [35] C.-D. Phung, B.F. Silva, M. Nogueira, and S. Secci, "MPTCP robustness against large-scale man-in-the-middle attacks," *Computer Networks*, vol.164, pp.106896.1–106896.14, Dec. 2019.
- [36] Y. Cao, J. Chen, Q. Liu, G. Lei, H. Wang, and I. You, "Can Multipath TCP Be Robust to Cyber Attacks with Incomplete Information?" *IEEE Access*, vol.8, pp.165872–165883, 2020.
- [37] Y. Cao, M. Collotta, S. Xu, L. Huang, X. Tao, and Z. Zhou, "Towards Adaptive Multipath Managing: A Lightweight Path Management Mechanism to Aid Multi-homed Mobile Computing Devices," *Applied Sciences*, vol.10, pp.1–18, 2020.
- [38] Y.-S. Lim, Y.-C. Chen, E.M. Nahum, D. Towsley, and K.-W. Lee, "Cross-layer path management in multi-path transport protocol for mobile devices," *Proc. IEEE INFOCOM*, pp.1815–1823, 2014.
- [39] C.P. Fu and S.C. Liew, "TCP Veno: TCP Enhancement for Transmission over Wireless Access Networks," *IEEE Journal on Selected Areas in Communications*, vol.21, no.2, pp.216–228, 2003.
- [40] M. Mathis, J. Semke, J. Mahdavi, and T. Ott, "The Macroscopic Behavior of the TCP Congestion Avoidance Algorithm," *ACM SIGCOMM Computer Communication Review*, vol.27, no.3, pp.67–82, 1997.
- [41] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: A Simple Model and Its Empirical Validation," *ACM SIGCOMM Computer Communication Review*, vol.28, no.4, pp.303–314, 1998.
- [42] UC Berkeley, LBL, USC/ISI and Xerox Parc, NS-2 documentation and software, version 2.35.
- [43] Google Code Project, Multipath-TCP: Implement multipath TCP on NS-2, <http://code.google.com/p/multipath-tcp/>, accessed: April 2020.

Yuanlong Cao received the B.S. degree in the Computer Science and Technology from Nanchang University, China, in 2006, the M.S. degree in the Software Engineering from the Beijing University of Posts and Telecommunications (BUPT), in 2008, and the Ph.D. degree from the Institute of Network Technology, BUPT, in 2014. He was an Intern/Software Engineer with BEA TTC, IBM CDL, and DT Research, Beijing, from 2007 to 2011. He is currently an Associate Professor with the School of Software, Jiangxi Normal University, China. His research interests include multimedia communications and the next-generation Internet technology. He has served as the Technical Reviewer for several journals, including the *IEEE Transactions on Industrial Informatics*, *IEEE Transactions on Cognitive Communications and Networking*, *IEEE Access*, *Computer Communications*, *Journal of Network and Computer Applications*.

Ruiwen Ji received the B.S. degree in the School of Economics and Management from the Beijing University of Posts and Telecommunications, China, in 2020. She is currently pursuing the M.S. degree in the Management Science and Engineering at Jiangxi Normal University, with Dr. Yuanlong Cao as her advisor. Her research interests include multimedia networking, the Internet technology, and information management and systems.

Lejun Ji received the B.S. degree in the School of Computer Science and Technology from the Liaocheng University, China, in 2020. She is currently pursuing the M.S. degree in the Management Science and Engineering at Jiangxi Normal University. Her research interests include multimedia networking, the Internet technology, and information management and systems.

Xun Shao received his Ph.D. in information science from the Graduate School of Information Science and Technology, Osaka University, Japan, in 2013. From 2013 to 2017, he was a researcher with the National Institute of Information and Communications Technology (NICT) in Japan. Currently, he is an Assistant Professor at the School of Regional Innovation and Social Design Engineering, Kitami Institute of Technology, Japan. His research interests include distributed systems and networking. He

is a member of the IEEE and IEICE.

Gang Lei is currently an Associate Professor and the Vice Chairman of the School of Software, Jiangxi Normal University, China, where he is also serving as the Associate Director of the Academic Committee. His research interests include the big data technology, computer network management, and information systems.

Hao Wang received the B.S. degree and the M.S. degree in the Department of Computer Science from the Harbin Institute of Technology, China, in 1984 and 1998, respectively. He received the Ph.D. degree in the Department of Computer Science from the Peking University, in 2006. He is currently a Professor with the School of Software, Jiangxi Normal University, China. His major research interests include computer network congestion control and computer network management. He was elected as

the University Young and Middle-aged Academic Leader in Jiangxi Province and the Distinguished Teacher of Jiangxi Province.