Decoupled TCP Extension for VLC Hybrid Network

Yanbing Liu, ⓘ Xiaowei Qin, ⓘ Tianyi Zhang, Ting Zhu, Xiaohui Chen, and Guo Wei

Abstract—In recent years, visible light communication (VLC), considered as a compelling technology to solve the spectrum problem of traditional radio frequency (RF) communication, has attracted attention in scientific and industrial communities. However, for a variety of reasons, the design of the VLC uplink scheme has become a significantly challenging problem. In this paper, in order to address the preceding problem, we propose a novel decoupled TCP extension protocol for a VLC hybrid network. Different from regular TCP protocol, decoupling operation to TCP transmission in our new protocol can break through TCP's limitation to a unidirectional link and allow users to fully utilize network resources in a VLC hybrid network consisting of a VLC downlink and a complementary uplink. Furthermore, today most hosts are equipped with several interfaces, so we combine decoupled TCP (DETCP) with multipath TCP (MPTCP) and extend DETCP to a multipath communication situation. Using several links simultaneously ensures more efficient and reliable data transmission in a VLC network. Based on our implementation of the Linux kernel, our experimental results show that our protocol can effectively accomplish the decouple work and achieve high throughput. In addition, several factors that influence the performance are analyzed based on our measurement results.

Index Terms—Decoupled TCP; Multipath decoupled TCP; Transmission control protocol; Visible light communication.

I. INTRODUCTION

Visible light communication (VLC) [1] using LED devices is an effective technology to alleviate spectrum crunch problems [2]. The susceptibility of LED enables VLC to transmit information with amplitude modulation of LED sources, typically called intensity modulation/direct detection (IM/DD) [3]. Several advantages are offered by VLC over traditional RF, including wide unlicensed spectrum, high spatial reuse, and high energy efficiency. Because of these advantages, VLC is considered a promising access option for fifth-generation wireless systems (5G) [4], and interplay between VLC and other access technology has become a research hotspot [5–7]. At present, several modulation schemes have been employed in IM/DD VLC systems, including on–off keying [8], carrier-less

amplitude and phase modulation (CAP) [9,10], and orthogonal frequency division multiplexing (OFDM) [11–14]; further, it has been demonstrated that a 3 Gb/s VLC system based on a single LED can be achieved in the OFDM modulation mode [15].

However, a challenging design task for a VLC system is to provide an efficient uplink. In general, energy-limited properties determine if mobile devices are unsuitable to be equipped with energy-intensive light sources. Consequently, it would be impractical to provide an VLC uplink for portable devices. Moreover, a VLC uplink exacts a highly directional optical beam to achieve an efficient and reliable transmission. This could result in dramatic decline of transmission rates due to a device's movement and rotation. Considering these problems, VLC is not an ideal choice for an uplink channel.

Over the past years, various researchers have proposed solutions such as RF [16], infrared (IR) [17,18], and reflective transceivers [19] to address VLC uplink problems at the physical layer. A problem common to all these solutions is that, because of the asymmetric model adopted by these approaches, some issues of upper-layer protocol compatibility exist. Considering the limitation of the above solutions, researchers implemented a hybrid system comprised of a VLC downlink and an Wi–Fi uplink [5], which enables the typical transmission control protocol (TCP) connection, but there exist some defects in the system. Extra devices and complex routing configuration required in their work may also discourage users from using this VLC/Wi–Fi hybrid system. More significantly, special socket programming is required to support the system. This design breaks compatibility with applications and might hamper a system's widespread usage. In conclusion, a systemic and compatible VLC uplink scheme has not been found.

A transport-layer solution is a novel idea to solve the preceding problem in VLC system design. Because it is insufficient for standard TCP protocol to support data transmission in a VLC hybrid system, we propose decoupled TCP (DETCP), an extension protocol to regular TCP, which achieves decoupling of the TCP's bidirectional transmission. This characteristic means one TCP flow can utilize two links at the same time, i.e., a link serves as the downlink, and a complementary link will play the role of uplink accordingly. With the help of DETCP, users can combine VLC with another physical link arbitrarily to form a complete hybrid bidirectional transmission system regardless of their asymmetry or unidirectionality, and there is no need for any intermediate coordinator devices and no extra requirement on routing

configuration and socket programming. Furthermore, besides the application in a VLC hybrid communication network, a great deal of flexibility and a wide range of applicability brought by DETCP are beneficial to fully exploit existing network resources.

Today, most smart terminals are equipped with multiple interfaces. To address the multihoming problem, the Internet Engineering Task Force (IETF) proposed multipath TCP (MPTCP) [20], which allows a single connection to transmit packets on multiple paths simultaneously. With MPTCP, mutual complementarities of advantages of Wi–Fi and 4G are promoted on mobile phones [21]; in data centers, multiple link bandwidth between two endpoints can be aggregated efficiently [22]. To help the VLC link participate in cooperative transmission in wireless networks and data center networks, we further extend DETCP to multipath cases by combination of DETCP and MPTCP; we call this new protocol MP-DETCP. Under the control and coordination of MP-DETCP, the hybrid bidirectional VLC system can collaborate effectively with other links, which are used by a TCP connection concurrently.

The main contributions of this work are as follows:

- The design and implementation of DETCP decoupling TCP's bidirectional transmission, which allows VLC to freely combine with a complementary uplink.
- The extension of DETCP to enable a unidirectional link in a multipath communication scenario and bring VLC and other links into effective teamwork in a single TCP connection.
- Actual experimentation and analysis on our testbed to evaluate the hybrid VLC network performance with DETCP and MP-DETCP.

In this paper, we describe DETCP and MP-DETCP protocol in detail and test their performance on a real VLC hybrid network. The rest of the paper is structured as follows. In Section II, we discuss the design ideas of DETCP to achieve management and coordination of the entire system. Section III focuses on operation details of DETCP externalizing these ideas. Section IV introduces MP-DETCP's implementation mechanisms for enabling decoupling in multipath conditions. In Section V, we measure and analyze the performance of DETCP and MP-DETCP implementation on a VLC hybrid network. Finally, Section VI concludes the paper and points out future research work.

II. DESIGN IDEAS OF DETCP

TCP is one of the most widely used transport layer protocols that provide reliable transmission service. However, with the development of technologies, some problems related to the combination of TCP and physical links are exposed. The benefit obtained from TCP is limited in situations in which physical links enable efficient and reliable bidirectional transmission. In the case that a VLC system can only provide a downlink leading to a unidirectional network interface, the transmission capability cannot be fully utilized by TCP independently.

A DETCP protocol is designed to solve this problem and realize data transmission on a hybrid system consisting of two independent links. No matter if the hybrid system consists of unidirectional links or bidirectional links or a combination of them, DETCP maintains compatibility with the transfer system. To be more specific, the design goals of DETCP are

1) to decouple the TCP's bidirectional transmission to two unidirectional transmissions running on two independent physical links according to the user's settings;
2) make the protocol usable with no change to existing applications that use the common TCP API.

To achieve these goals, the several differences between DETCP and regular TCP on several aspects must be taken into consideration. The following sections describe the main principles of DETCP.

A. Link Architecture

An Internet socket is used by regular TCP to identify the end-points pair on each host, and a pair of sockets specify a TCP connection uniquely. In this way, a connection is bound to a specific physical link denoted by a socket. However, for unidirectional links such as VLC, an existing TCP link structure cannot enable data transfer in each direction between hosts because a unidirectional link cannot form a transmission circuit.

To solve this problem, DETCP adopts a distinctive link architecture, as shown in Fig. 1. In a new system, two opposite physical links are in charge of a TCP connection and undertake its transfer collectively instead of the original bidirectional link. Consequently, at the connection level, data transfer is still bidirectional, but what we can observe
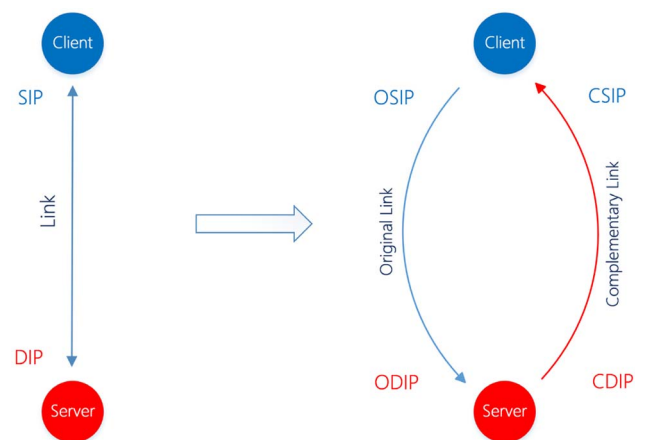

Fig. 1. DETCP link architecture consisting of original link and complementary link. Left half is the link architecture of regular TCP. SIP, source IP address; OSIP, original source IP address; CSIP, complementary source IP address; DIP, destination IP address; ODIP, original destination IP address; CDIP, complementary destination IP address.

at the link layer are two separate one-way transmissions in the opposite direction, as described in the first goal.

B. Connection Identity

In standard TCP protocol, each connection is identified by a four-tuple, including source address, source port, destination address, and destination port, as defined in [23]. The premise behind this control mechanism is that the TCP connection is created on a single link. Now that DETCP needs cooperation of two links, a six-tuple (original source address, complementary source address, source port, original destination address, complementary destination address, destination port) is necessary to specify the DETCP connection. Packet transmission of a DETCP connection will be directed by the six-tuple.

To satisfy the second goal, the socket structure remains unchanged, so we look for other solutions to pass the complementary address into the protocol stack. A kernel configuration file is an obvious and effective way to realize the above function. In the design of DETCP, users can set the complementary address with the configuration file, so that the complete six-tuple is created with the original standard socket pair and complementary addresses. In this way, DETCP guarantees compatibility with the application layer.

C. Connection Initiation and Close

The process of initiating and closing the DETCP connection is quite similar to that of the normal TCP. As a mature and effective method, the handshake mechanism [24] is retained, but there are two main changes between DETCP and TCP in the initiating and closing stage:

1) Because the local complementary address is unknown to the remote host before connection establishment, and the complete address information is necessary for each host to execute decoupled transmission, we add a TCP option to inform the remote host about complementary addresses in the connection initiation process.

2) The trait of decoupling shows up in both the connection initiation stage and connection close stage. It particularly reflects on transmission link handover

according to the link's transmission direction set before. For example, Fig. 2 shows the three-way handshake process directed by the six-tuple.

D. Data Transfer

In order to ensure decoupling in the data transfer process, when the next packet [data or acknowledgement (ACK)] is ready to be sent, the current link's direction must be checked. If its direction is from remote host to local host, the sending work will be accomplished on its complementary link.

In the process of data reception, there is yet another significant problem to be addressed. If the kernel uses an address parsed from the IP header directly to demultiplex arriving packets, the packets may be bound to the wrong DETCP connection. For example, the server sends the data packet to the client with a downlink, and the client will echo back the ACK with an uplink. In this situation, addresses parsed from the ACK packet are the uplink's address pair but, for the server, the connection is bound to the downlink's addresses, and this mismatching leads to misoperation of the incoming packet. We add a new TCP option to handle this problem (further details are in Section III).

III. OPERATIONS OF DETCP

Figure 3 shows the state diagram of DETCP. The left half is the server's overall process, and part of the client is shown in the right half of the figure. We annotate the DETCP's major changes on the figure. In the stage of connection initiation, apart from conventional decoupling to transmission, several settings and options help the connection to establish correctly. After laying the groundwork, packets will be transferred on the appropriate link according to their direction. The following section describes specific operations of the DETCP.

To start with, the necessary preparation work to be performed is the interface direction setup. Our protocol allows users to assign a specific interface a direction ("Out," "In," "Two-Way") through configuration files. When the interface bound to a socket is set as a unidirectional interface, the user needs to prescribe another interface as a complementary
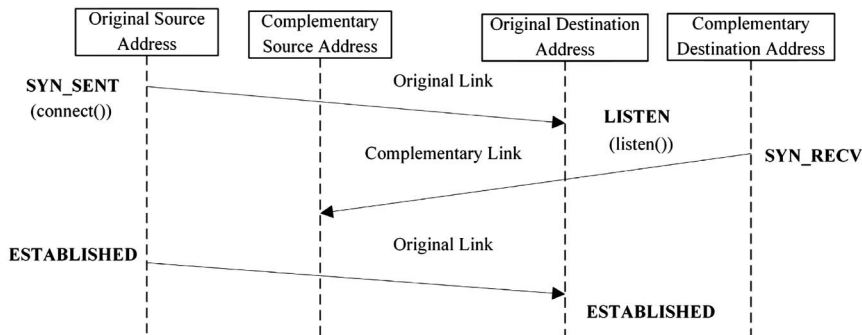

Fig. 2.   Three-way handshake directed by six-tuple in connection initiation phase.
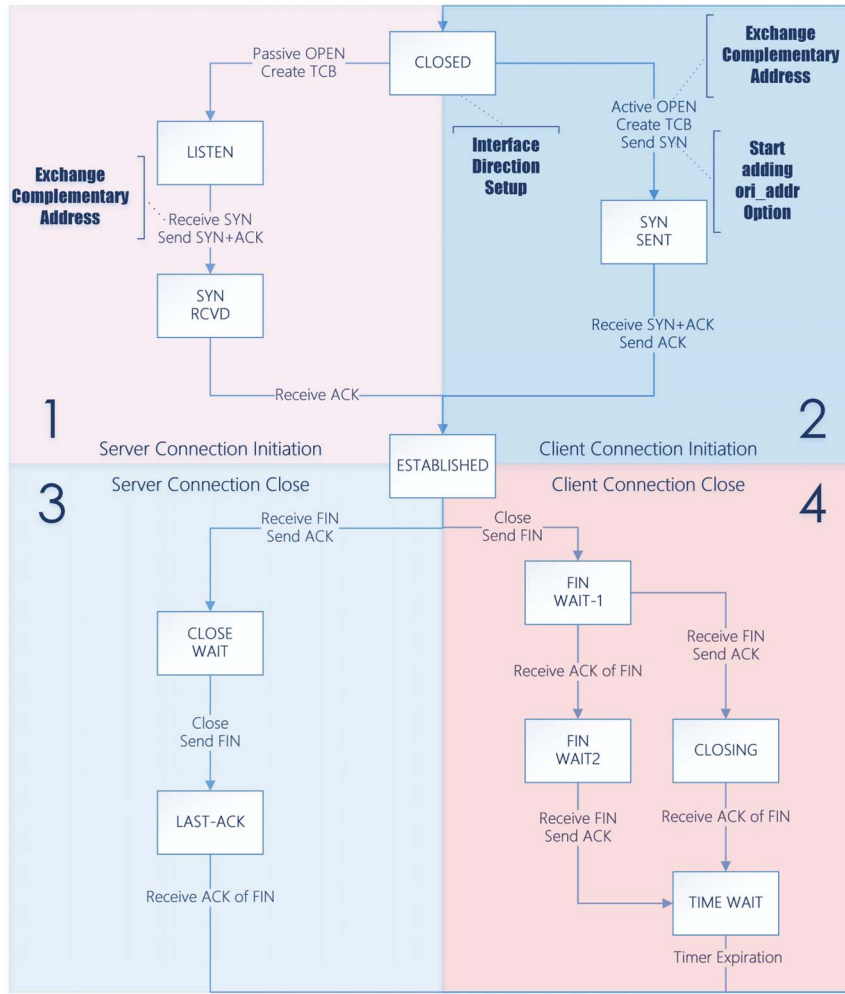
Fig. 3.   State diagram of DETCP. Left half uses the complementary link to send packets; right half uses the original link to send packets.

interface, as mentioned in Section II. All four interfaces uniquely determine a complete connection.

The three-way handshake remains in the connection initiation phase, with alterations to achieve the exchange of information about the complementary link's IP address, as mentioned in Section II. A TCP option CPT_ADDR is designed to accomplish the exchange work. The specific content of every packet is as follows:

1) SYN (client to server): client's complementary IP address.
2) SYN/ACK (server to client): server's complementary IP address.

The principle of decouple runs throughout the whole transmission process, including connection initiation, data transfer, and connection close. Here is an example given to illustrate the decouple rule in these stages.

Suppose the original link is set to uplink (client to server), and the complementary link serves as a downlink (server to client) accordingly. Transmission link behaviors in stages of connection initiation, data transfer, and connection close are described as follows:

1) Connection Initiation (Shown in Fig. 4)
   a) SYN (client to server): the original link.
   b) SYN/ACK (server to client): the complementary link.
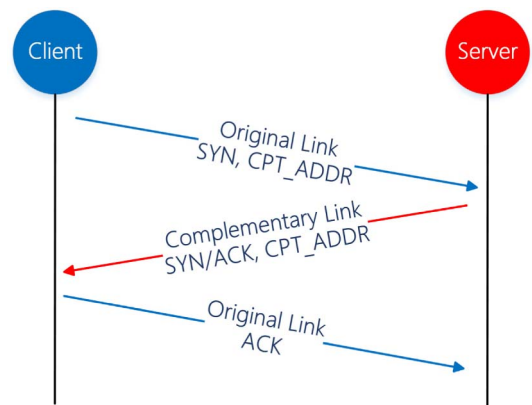   c) ACK (client to server): the original link.

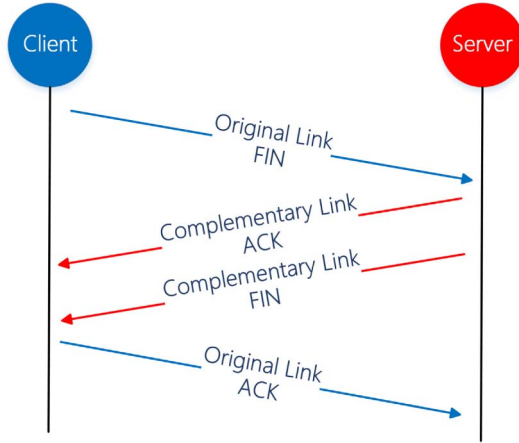Fig. 4.   DETCP connection establishment with three-way handshake.

Fig. 5. DETCP connection close with four-way handshake.

2) Data Transfer
    a) Packets from client to server: the original link.
    b) Packets from server to client: the complementary link.
3) Connection Close (Shown in Fig. 5)
    a) FIN (client to server): the original link.
    b) ACK (server to client): the complementary link.
    c) FIN (server to client): the complementary link.
    d) ACK (client to server): the original link.

In the last part of Section II, we discuss the problem of handling of incoming packets and bring forward a TCP option-based solution. The new TCP option, including the original unchanged address, will be parsed in the process of finding an appropriate connection for an incoming packet, and addresses being parsed replace addresses in the IP header to ensure packets received are bound to the correct link.

IV. MULTIPATH EXTENSION OF DETCP

As technologies evolve, networks are on a trend toward multipath. However, traditional TCP, in essence, is designed to be a single-path protocol and incapable of making use of multiple links concurrently. On mobile devices, when handover from Wi–Fi to 4G occurs, existing TCP connections must be torn down before new connections are established; although there are redundant physical links between two hosts in modern data centers, regular TCP protocol can only choose one link randomly and wastes a great deal of resources in the network.

To enhance throughput and provide better TCP fairness, multipath TCP is proposed, and when standardized by IETF achieves significant performance improvement in multipath networks. According to a set of operations defined in RFC6824 [20], MPTCP builds correspondences between a single TCP connection and multiple links and supports simultaneous transmission of these links, which means it can provide higher aggregate bandwidth of multiple links. Meanwhile, the fairness to regular TCP is also guaranteed. MPTCP has been effectively applied in both mobile devices and data center cases. According to measurements focusing on mobile device connection handover, the result illustrates a smooth and seamless handover with MPTCP [21]; controlled by the congestion control and path selection mechanism, MPTCP has also been proven to give improved throughput and better fairness in data center networks [22].

According to the multihomed development trend of terminals, we design MP-DETCP, including support for a multipath network on the basis of DETCP, and we expect it can facilitate VLC deployment and be used in combination with other links in heterogeneous networks.

A. Design Goals

MP-DETCP's purpose is clear: it allows a single TCP connection to transmit data packets on several available "bidirectional links" between local and remote endpoints, which is the multipath work defined by MPTCP. In VLC heterogeneous networks, the "bidirectional links" fall into two basic categories: the actual bidirectional link and the hybrid transmission link combined from a unidirectional link such as VLC and a complementary link. Therefore, the objective of MP-DETCP is to perform both multipath and decoupling work on heterogeneous networks, which may contain unidirectional and bidirectional links. Figure 6 provides an illustration of MP-DETCP link architecture.

B. Design Overview

The initiation of MP-DETCP connection proceeds in three stages: masterflow establishment, address information exchange, and additional establishment.
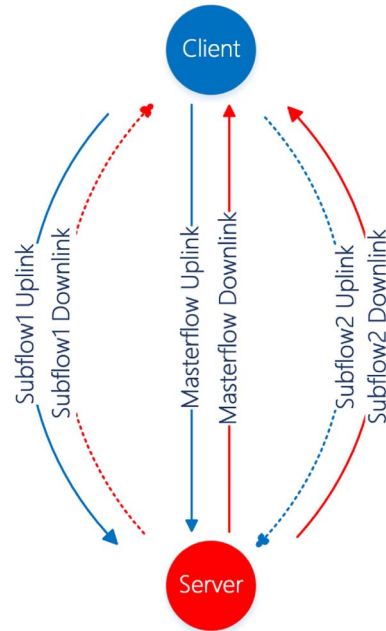

Fig. 6. MP-DETCP link architecture consisting of a masterflow and two subflows. Dotted lines represent that data transmission in this direction is blocked due to the link's unidirectional characteristic.
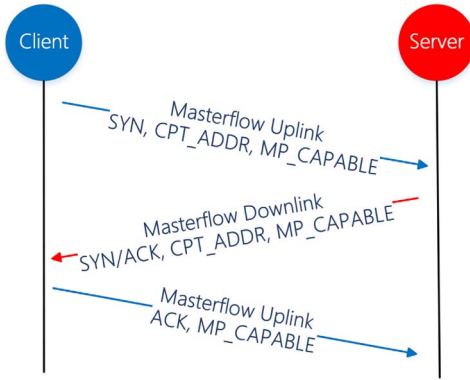
Fig. 7. MP-DETCP connection establishment with three-way handshake.

1) Masterflow Establishment: Before masterflow establishment starts, the setup information in the configuration files would indicate the direction property of the masterflow (either bidirectional or unidirectional). If it is unidirectional, the complementary link's addresses are also needed to be configured, and the two links will jointly carry the masterflow.

The decoupling operation to perform unidirectional masterflow establishment is similar to operation in a DETCP connection. CPT_ADDR option is also required to inform the remote host of a complementary link address, while the MP_CAPABLE option would ensure the support of hosts to MP-DETCP and initialize the multipath management module. The three-way handshake is shown in Fig. 7.

2) Address Information Exchange: With CPT_ADDR, both sides of the conversation have obtained the pertinent information of the masterflow. Next, the information of additional alternative addresses will be exchanged to prepare for the subflow establishment. Packets with the ADD_ADDR option will be sent to inform the remote host of additional addresses and their direction. After the information exchange ends, both sides of the connection will generate two tables containing all address information, as shown in Tables I and II.

3) Additional Establishment: According to address information tables, a client will combine a local address with a remote address to form an address pair, and their directions will confine the matchmaking process and decide the direction of the subflow. The mapping relationship is shown in Table III.

TABLE I
Local Addresses Information

Address ID	IP Address	Direction	Property
1	192.168.1.1	Out	Masterflow Uplink
2	192.168.1.2	In	Masterflow Downlink
3	192.168.1.3	Out	Subflow
4	192.168.1.4	In	Subflow
5	192.168.1.5	Two-Way	Subflow
…	…	…	…

TABLE II
Remote Addresses Information[a]

Address ID	IP Address	Direction	Property
1	192.168.1.6	In	Masterflow Uplink
2	192.168.1.7	Out	Masterflow Downlink
3	192.168.1.8	In	Subflow
4	192.168.1.9	Out	Subflow
5	192.168.1.10	Two-Way	Subflow
…	…	…	…

[a]Direction is interfaces' direction. "Out" refers to the direction from local host to remote host; "In" refers to the direction from remote host to local host; "Two-Way" indicates that the interface is bidirectionality. The property is used to record if this address has been used by master uplink or downlink.

TABLE III
Direction Mapping Relationship

Local Address Direction	Remote Address Direction	Whether to Build Subflow	Subflow Direction
Out	Out	No	/
Out	In	Yes	Out
Out	Two-Way	Yes	Out
In	Out	Yes	In
In	In	No	/
In	Two-Way	Yes	In
Two-Way	Out	Yes	In
Two-Way	In	Yes	Out
Two-Way	Two-Way	Yes	Two-Way

Because there are three different cases ("Out," "In," "Two-Way") in regard to the subflows' directions, establishment of the subflow is divided into two categories:

1) Link's direction is "Out" or "Two-Way" for client (shown in Fig. 8)
   a) SYN (client to server): the subflow link.
   b) SYN/ACK (server to client): the masterflow downlink.
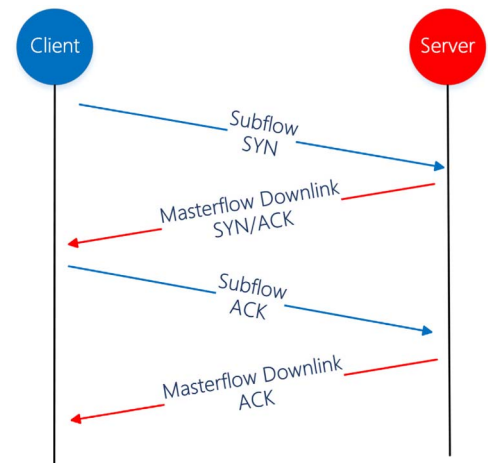   c) ACK (client to server): the subflow link.

Fig. 8. MP-DETCP subflow establishment with four-way handshake when link's direction is "Out" or "Two-Way" for client.
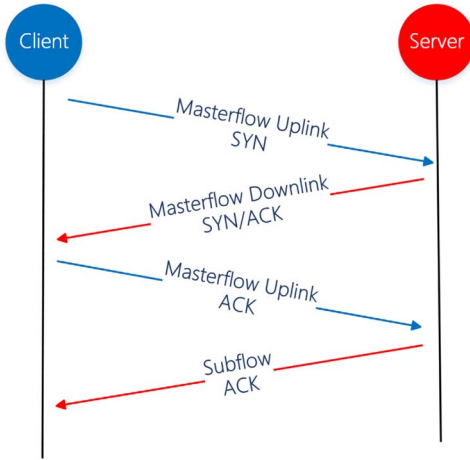
Fig. 9.   MP-DETCP subflow establishment with four-way handshake when link's direction is "In" for client.

   d) ACK (server to client): the masterflow downlink.
2) Link's direction is "In" for client (shown in Fig. 9)
   a) SYN (client to server): the masterflow uplink.
   b) SYN/ACK (server to client): the masterflow downlink.
   c) ACK (client to server): the masterflow uplink.
   d) ACK (server to client): the subflow link.

Through the four-way handshake, connectivity of the subflow link is proved, and the subflow is completely established. However, for unidirectional subflow, a complementary subflow is required before it starts transferring data packets. At present, the pairing mechanism implemented in our kernel is to search for the unpaired opposite link simply by path index. In future work, we will investigate the performance of different link combinations and explore the best combination of various application scenarios. This work will provide guidance about how to design a high-performing, adaptable, and flexible pairing mechanism. After finishing the pairing work, the bidirectional link consisting of two subflows will begin its formal data transmission. The later process including data transfer and connection close is similar to the contents described in Section III, so we will not reiterate it here.

V. EVALUATION

In this section, we implemented DETCP and MP-DETCP in a Linux kernel, and several experiments are performed to evaluate our new protocol in a VLC hybrid network. The network topology is shown in Figs. 11 and 12. DETCP is evaluated on the hybrid system combinations of a VLC and Ethernet link. A 20 M unidirectional VLC link serves as the downlink, and a 100 M Ethernet link is set to the uplink. The capacities of all access links connecting to the router and VLC system are set to 100 Mbps. The settings ensure transmission of the VLC link will not be restricted by access links under normal circumstances. To test MP-DETCP, the first testbed is slightly modified to add another 100 M Ethernet link, so that the system is
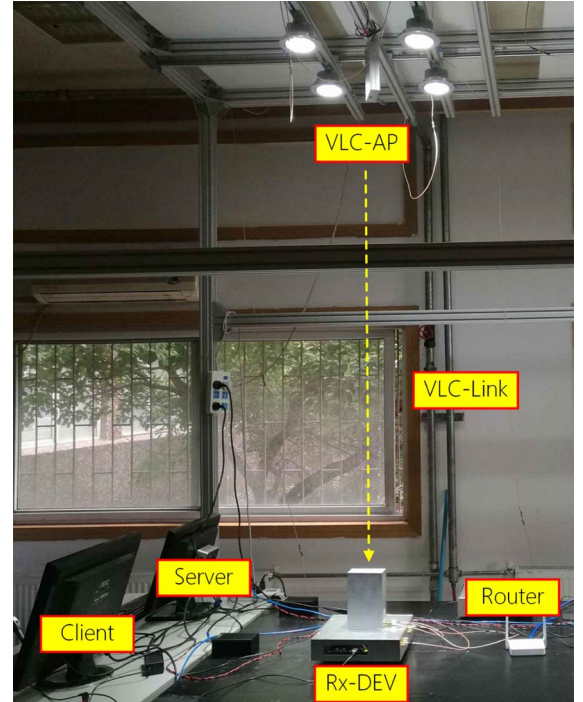

Fig. 10.   Our transmission system in practical experiments.

guaranteed to have two bidirectional transmission links: one is a VLC hybrid link, and the other is an Ethernet link. Figure 10 shows the transmission system in practical experiments.

In this hybrid system, a medium-sized file is transferred from the server to client, so data packet transmission is largely completed on the downlink, and the main function of the uplink is to transfer ACK packets. Therefore, the utilization ratio of the bandwidth is defined as the ratio of downlink throughputs to the downlink bandwidth, which is an important indicator of transmission performance. Packet transmission direction is indicated by the arrows in Figs. 11 and 12.

Our experiments are divided into three parts. The first experiment is used to test the DETCP's availability for combinations of VLC downlink and Ethernet uplink. Then, in the second experiment, we test the robustness through experiments under various harsh conditions and analyze the influence of these factors to the DETCP's performance. Finally, we compare the DETCP and MP-DETCP's performance under different link conditions.
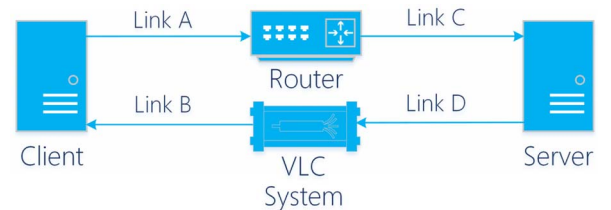

Fig. 11.   Network topology for DETCP experiments. Upper link is the complementary uplink; lower link is the VLC downlink.
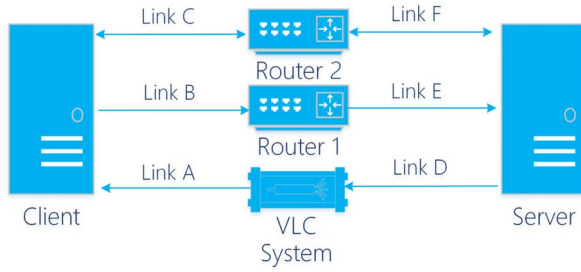
Fig. 12.   Network topology for MP-DETCP experiments. Link A and Link D comprise VLC downlink, and Link B together with Link E comprise the complementary uplink. Besides, Link C and Link F compose the Ethernet subflow.

A. Availability Testing

First, we want to verify whether DETCP can effectively use the available bandwidth in a high-speed hybrid environment. Besides, in order to check the DETCP's tolerance of poor complementary link performance, we configured the client's uplink interface to drop a percentage of the packets randomly.

As shown in Fig. 13, DETCP achieves a highly stable and efficient transmission in a VLC hybrid system without packet loss. In this scenario, the utilization ratio of the bandwidth reaches 95%, and the result demonstrates that DETCP is effective for hybrid system traffic control. When there are packet losses, the goodput of DETCP remains stable with packet loss ratios of 0.5%, 1.0%, or 2.0%, and the utilization ratio of the bandwidth can still reach over 92%. Only when the packet loss ratio reaches over 5%, the performance of DETCP is influenced to a large extent. The measurements indicate that DETCP's tolerance of a complementary link is considerable, and overall system performance will not be appreciably affected by the slight decline in complementary link performance.
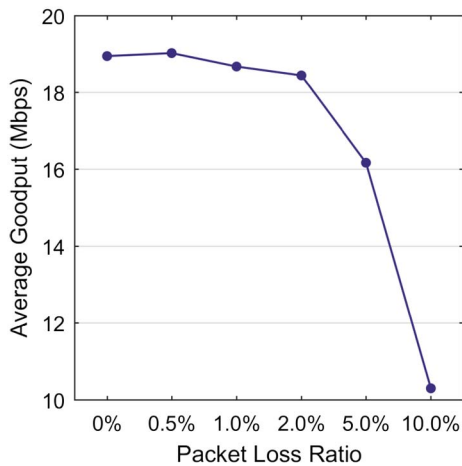
B. Performance Influence Factors

To perform thorough robustness testing on the DETCP of the VLC hybrid system, we set some harsh transmission environments such as packet loss, delay, and packet reorder. Different from settings in the previous section, we configured the downlink's downlink interface to set up. Some analyses are followed on these factors and their influence on system performance.

At first, an issue that influences DETCP performance is packet loss. We configured the interface to randomly drop some packets and varied the loss rate from 0% to 4% on the VLC downlink. As seen in Fig. 14, the average throughput of a connection is affected by the packet losses, and it decreases approximately linearly as the packet loss ratio rises. Another important phenomenon that occurs is that, when packet loss is high, variance of repeated measures enlarges, and the average transmission rate becomes unstable.

Another significant factor that affects DETCP is delay. We insert an additional transmission delay from 0 to 100 ms on the VLC link. Figure 15 shows the average throughput of the whole system. When the delay is low (less than 20 ms), the throughput remains stable, and bandwidth is fully utilized. When the delay rises to 50 ms, the average throughput falls slightly. Further, the average throughput drops by more than 35% when the delay reaches 100 ms, and packet transmission becomes unstable in our observation.

Packet reorder is the last factor that we analyze. Reorder is realized in our experiments with settings as follows: part of the packets will be delayed in transit, while the other packets are transmitted with no delay. Therefore, the number of reorder packets and the delay are controlled through configuration. Figure 16 shows the impact of packet reorder on performance with different percentages of reorder packets. In all cases, there appears downward trends in the average throughput. However, there are also significant differences on the variation amplitude. When the

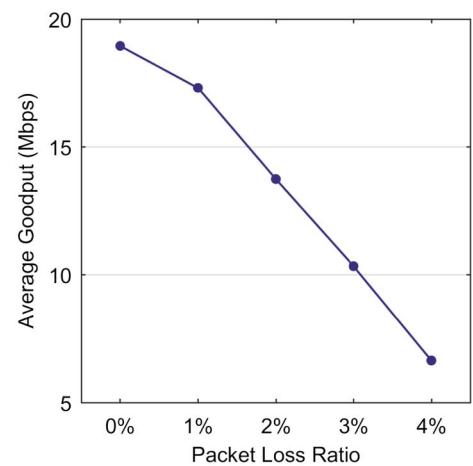Fig. 13.   Average throughputs under different uplink packet loss ratios.

Fig. 14.   Average throughputs under different downlink packet loss ratios.
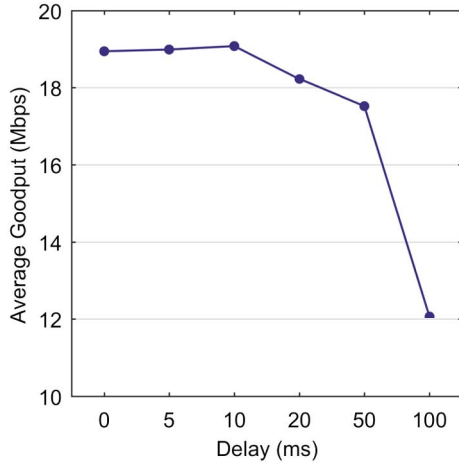
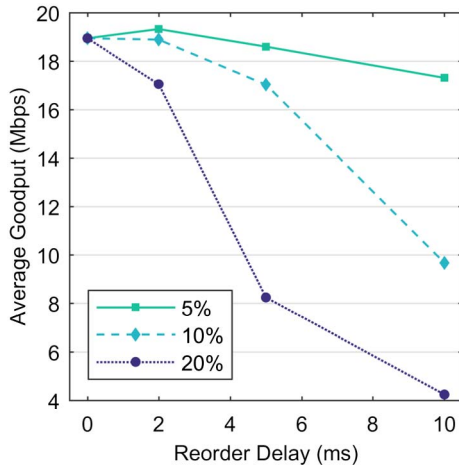Fig. 15.   Average throughputs under different downlink delay.

Fig. 16.   Average throughput comparison under different downlink packet reorder situations.

percentage is 5%, the throughput just decreases slightly when delay reaches 10 ms. But when it comes to 10% or 20%, throughput drops rapidly when transmission delay rises. From Fig. 16, the throughput with 5% reorder packets and 10 ms delay is approximately equal to the throughput with 10% reorder packets and 5 ms delay or with 20% reorder packets and 2 ms delay. Consequently, it is essential for the system to control both the number of reorder packets and the delay, in order to maintain stable and efficient transmission. When the VLC downlink performance deteriorates significantly, for systems in which uplink is bidirectional such as fiber or Wi–Fi, we consider falling back to single-path standard TCP running on the uplink. This might be an effective solution to improve system robustness when the VLC downlink suffers serious performance degradation.

C. Performance Comparison

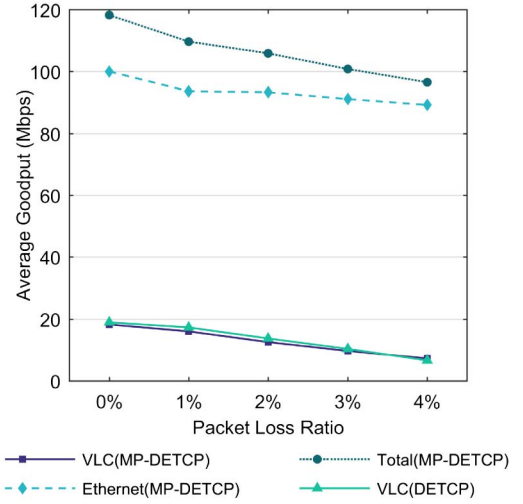One of the main goals of multipath work is to improve throughput. In this experiment, we examine the throughput

Fig. 17.   Average throughput comparison between DETCP and MP-DETCP under different downlink packet loss ratios.

of a MP-DETCP transmission system and compare it with throughput of DETCP in the same link conditions. We record the measurement results when loss ratio of a VLC downlink varies from 0% to 4%. The results, shown in Fig. 17, indicate that, although VLC link's throughput is slightly below that of DETCP, a high-bandwidth utilization is still achieved by MP-DETCP. When data are transmitted without packet loss, the total throughput reaches nearly 115 Mbps, which confirms that the MP-DETCP is capable of providing aggregate bandwidth effectively. As packet loss increases, throughput of the VLC link declines dramatically; meanwhile, throughput of Ethernet link also experiences a slight dip, but the total throughput still maintains a relative high level. It means that, when the VLC link is severely disrupted or blocked, data transmission can still be efficiently completed on the other link.

VI. CONCLUSION AND FUTURE WORK

To solve the problem of transmission of a VLC system, we propose a DETCP protocol to realize decoupling of the bidirectional transmission. On this basis, MP-DETCP is designed to further improve performance of the VLC hybrid system. Our implementation based on the Linux kernel has withstood the tests of availability and robustness in a VLC hybrid network, and the experimental results demonstrate that decoupled transmission is performed efficiently under control of DETCP and MP-DETCP.

Our new protocols provide several research points. The combination of physical links deserves further study to explore potential link pairs with outstanding performance. Besides, path asymmetry of a network may cause TCP performance problems [25], so some modules of TCP such as congestion control [26] can be improved to better accommodate decoupled transmission in a VLC hybrid network. In future work, we plan to investigate these issues to achieve performance enhancement of our protocols.

REFERENCES

[1] Y. Tanaka, T. Komine, S. Haruyama, and M. Nakagawa, "Indoor visible communication utilizing plural white LEDs as lighting," in *12th IEEE Int. Symposium on Personal, Indoor and Mobile Radio Communications*, 2001.

[2] M. Kavehrad, "Optical wireless applications: a solution to ease the wireless airwaves spectrum crunch," *Proc. SPIE*, vol. 8645, p. 86450G, 2013.

[3] J. M. Kahn and J. R. Barry, "Wireless infrared communications," *Proc. IEEE*, vol. 85, pp. 265–298, 1997.

[4] S. Wu, H. Wang, and C.-H. Youn, "Visible light communications for 5G wireless networking systems: From fixed to mobile communications," *IEEE Netw.*, vol. 28, pp. 41–45, 2014.

[5] S. Shao, A. Khreishah, M. Ayyash, M. B. Rahaim, H. Elgala, V. Jungnickel, D. Schulz, T. Little, J. Hilt, and R. Freund, "Design and analysis of a visible-light-communication enhanced WiFi system," *J. Opt. Commun. Netw.*, vol. 7, pp. 960–973, 2015.

[6] S. Shao, A. Khreishah, M. Ayyash, M. B. Rahaim, H. Elgala, V. Jungnickel, D. Schulz, T. Little, J. Hilt, and R. Freund, "Coexistence of WiFi and LiFi toward 5G: Concepts, opportunities, and challenges," *IEEE Commun. Mag.*, vol. 54, no. 2, pp. 64–71, 2016.

[7] X. Bao, X. Zhu, T. Song, and Y. Ou, "Protocol design and capacity analysis in hybrid network of visible light communication and OFDMA systems," *IEEE Trans. Veh. Technol.*, vol. 63, pp. 1770–1778, 2014.

[8] B. Fahs, A. J. Chowdhury, and M. M. Hella, "A 12-m 2.5-Gb/s lighting compatible integrated receiver for OOK visible light communication links," *J. Lightwave Technol.*, vol. 34, pp. 3768–3775, 2016.

[9] Y. Wang, L. Tao, Y. Wang, and N. Chi, "High speed WDM VLC system based on multi-band CAP64 with weighted pre-equalization and modified CMMA based post-equalization," *IEEE Commun. Lett.*, vol. 18, pp. 1719–1722, 2014.

[10] F.-M. Wu, C.-T. Lin, C.-C. Wei, C.-W. Chen, Z.-Y. Chen, and K. Huang, "3.22-Gb/s WDM visible light communication of a single RGB LED employing carrier-less amplitude and phase modulation," in *Optical Fiber Communication Conf.*, 2013, paper OTh1G.4.

[11] J. Armstrong, "OFDM for optical communications," *J. Lightwave Technol.*, vol. 27, pp. 189–204, 2009.

[12] H. Elgala, R. Mesleh, and H. Haas, "Practical considerations for indoor wireless optical system implementation using OFDM," in *10th Int. Conf. on Telecommunications (ConTEL)*, 2009, pp. 25–29.

[13] S. D. Dissanayake and J. Armstrong, "Comparison of ACO-OFDM, DCO-OFDM and ADO-OFDM in IM/DD systems," *J. Lightwave Technol.*, vol. 31, pp. 1063–1072, 2013.

[14] R. Mesleh, H. Elgala, and H. Haas, "On the performance of different OFDM based optical wireless communication systems," *J. Opt. Commun. Netw.*, vol. 3, pp. 620–628, 2011.

[15] D. Tsonev, H. Chun, S. Rajbhandari, J. J. McKendry, S. Videv, E. Gu, M. Haji, S. Watson, A. E. Kelly, G. Faulkner, M. D. Dawson, H. Haas, and D. O'Brien, "A 3-Gb/s single-LED OFDM-based wireless VLC link using a gallium nitride μLED," *IEEE Photon. Technol. Lett.*, vol. 26, pp. 637–640, 2014.

[16] I. Stefan and H. Haas, "Hybrid visible light and radio frequency communication systems," in *IEEE 80th Vehicular Technology Conf. (VTC Fall)*, 2014, pp. 1–5.

[17] K.-D. Langer and J. Grubor, "Recent developments in optical wireless communications using infrared and visible light," in *9th Int. Conf. on Transparent Optical Networks (ICTON'07)*, 2007, pp. 146–151.

[18] R. Perez-Jimenez, J. Rufo, C. Quintana, J. Rabadan, and F. Lopez-Hernandez, "Visible light communication systems for passenger in-flight data networking," in *IEEE Int. Conf. on Consumer Electronics (ICCE)*, 2011, pp. 445–446.

[19] T. Komine, S. Haruyama, and M. Nakagawa, "Bidirectional visible-light communication using corner cube modulator," *IEIC Tech. Rep.*, vol. 102, pp. 41–46, 2003.

[20] A. Ford, C. Raiciu, M. Handley, and O. Bonaventure, "TCP extensions for multipath operation with multiple addresses," IETF RFC 6824, 2013.

[21] C. Paasch, G. Detal, F. Duchene, C. Raiciu, and O. Bonaventure, "Exploring mobile/WiFi handover with multipath TCP," in *ACM SIGCOMM Workshop on Cellular Networks: Operations, Challenges, and Future Design*, 2012, pp. 31–36.

[22] C. Raiciu, S. Barre, C. Pluntke, A. Greenhalgh, D. Wischik, and M. Handley, "Improving datacenter performance and robustness with multipath TCP," in *ACM SIGCOMM Computer Communication Review*, 2011, pp. 266–277.

[23] J. Postel, "Transmission control protocol," IETF RFC 793, 1981.

[24] C. A. Sunshine and Y. K. Datal, "Connection management in transport protocols," *Comput. Netw. (1976)*, vol. 2, pp. 454–473, 1978.

[25] H. Balakrishnan and V. N. Padmanabhan, "TCP performance implications of network path asymmetry," IETF RFC 3449, 2002.

[26] M. Allman, V. Paxson, and E. Blanton, "TCP congestion control," IETF RFC 5681, 2009.