A WiFi-Direct Based Local Communication System

Zijian Wang[†], Fuliang Li[‡§(*)], Xingwei Wang[†‡], Tengfei Li[‡], Tao Hong[‡]
[†] Software College of Northeastern University, Northeastern University, Shenyang, China
[‡] School of Computer Science and Engineering, Northeastern University, Shenyang, China
[§]Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education

***Abstract*—The infrastructure-based networks will be unavailable in the case of infrastructure failures (e.g., earthquake and tsunami) or in crowded areas (e.g., concert and conference hall). This promotes the evolution of location communication systems, which also benefit offloading computing, mobile edge computing and mobile crowdsourcing. In this paper, we utilize Wi-Fi Direct (WFD) to develop a local communication system. We not only present an intra-group communication solution by the native characteristics of WFD, but also propose a general solution for the bidirectional inter-group communication, which is beyond the scope of WFD specifications. Experimental results show that the maximum throughput of intra-group communication could reach up to 31.7 $Mbps$, and the maximum throughput of inter-group communication is 8.26 $Mbps$.**

I. Introduction

There will be 1.5 mobile devices per capita and the number of smartphones will be over 50% of global devices by 2021 [1]. With the explosive growth of mobile devices, wireless communication demands increase sharply. This promotes the development of radio access technologies (e.g., 5G and Wi-Fi), which greatly enriches the means of communications. Wireless communications mainly depend on infrastructure-based networks, such as cellular networks and wireless local area networks (WLANs). However, such infrastructure-based communications may be interrupted by infrastructure failues, such as natural disasters or terrorist attacks. These communications may also be unavailable in some crowded areas (e.g., concert and conference hall). Therefore, device-to-device (D2D) technologies like Bluetooth, Wi-Fi Direct and Wi-Fi hotspot have emerged. Based on these technologies, local communication system [2] can be developed to provide an instant communication environment for mobile users to communicate with each other in a certain range.

Considering the resource constraints of mobile devices, mobile cloud computing provides mobile users the powerful computation and storage capabilities through clouds [3]. However, it is challenging to process the tremendous request generated by mobile devices, resulting in the intolerable latency and bad quality of experience. To overcome such challenging issues, a new paradigm called mobile edge computing is proposed [4, 5]. It changes the computation position from the cloud to the edge, reducing the load of clouds and providing a better quality of experience. Shi et al. [6] pointed out that the edge can be any computing and network resources along the path between the data sources and the cloud data centers. Therefore, to relieve the pressure of clouds or edge servers, mobile devices can be regarded as a computational edge and used for constructing local communication systems to solve some collaborative tasks. In this paper, we implement and evaluate a WFD-based local communication system on Android devices. Our contributions are summarized as follows.

(1) We implement the intra-group communication based on the native characteristics of WFD. We not only realize the communication between the group owner and the group member, but also allow the communication between any two group members. We also propose a group member discovery mechanism to maintain the member list in each device.

(2) More importantly, we propose a general solution for the inter-group communication, which is not introduced in WFD specifications. The proposed inter-group communication is composed of a specific LC/GO connection schema for connecting with each other among groups and a node relaying based approach for inter-group communication. In addition, we also consider how to manage the members in the connected group and present a multicast based inter-group member discovery mechanism. The proposed solution is general and feasible, which can be applied on off-the-shelf Android devices.

(3) We use a real test bed to evaluate the effectiveness and the performance of the proposed local communication system. Experiment results show that we can get a maximum throughput of 31.7 $Mbps$ for the intra-group communication and a maximum throughput of 8.26 $Mbps$ for the inter-group communication.

The remainder of this paper is organized as follows. Section II and III present the intra-group communication and inter-group communication mechanisms separately. Section IV evaluates the performance of the proposed local communication system. The related work is introduced in Section V. Finally, Section VI concludes the whole paper.

II. WFD Based Intra-Group Communication

At the beginning, we give the specific definitions of the terms used in this paper.

- *group owner (GO)*: WFD group creator.
- *group member (GM)*: the device in the current group.
- *legacy client (LC)*: the device that connects to the current group through Wi-Fi interface.
- *gateway node*: the GO that acts as a LC to connect to another group for inter-group communication.
- *relay node*: the device that is responsible for relaying the traffic toward another group.

978-1-5386-2542-2/18/$31.00

- *member*: the device in the current group or other connected groups.
- *list*: the list can be the member list or the GM list, which contains the device information, for example IP address. Note that the GM list is a subset of the member list.

The intra-group communication could be implemented based on the native characteristics of WFD. However, some details about communication are not introduced in WFD technical specifications, such as the communication between GMs and the management of GMs. The proposed solution could improve the WFD protocol in the aspect of intra-group communication. Fig. 1 shows the scenario of the intra-group communication, where both of the GMs (i.e., $GM1A$ and $GM1B$) connect to $GO1$ through P2P interface. That makes the three devices interconnect with each other logically. That is to say, the communication is allowed between a GO and a GM, as well as between any two GMs, which requires the GO to relay the frames at the MAC-layer. To guarantee any two devices can communicate with each other, some transport-layer tunnels are needed. In this paper, the intra-group communication is implemented based on TCP completely. Note that UDP could also be adopted.

In addition, we present a GM discovery mechanism to maintain the GM list in each device as follows. 1) When a device connects to the GO, the GO will send the current GM list to the new connected device by TCP. 2) When the GO detects some devices joining in or leaving the group, it will broadcast such devices to the GMs. And each GM will update the current GM list according to the type of the received broadcast message. Note that the type of the broadcast message can be adding, deleting or replacing. 3) The GO will periodically broadcast the GM list to the GMs, ensuring the consistency of the GM list in each GM.

III. WFD Based Inter-Group Communication

Inter-group communication is not introduced in WFD specifications. However, we can extend the use of WFD, allowing any two devices from different groups to communicate with each other. In this section, we present a general solution to implement the inter-group communication on the widely popularized Android devices.

A. LC/GO Connection Schema

By leveraging the concurrent connection characteristics of WFD, we propose a LC/GO connection schema as shown in Fig. 2, where one GO as a LC utilizes the Wi-Fi interface to connect to another GO, building a communication bridge between two groups. Based on the proposed connection schema, the connection between two groups is established. We define two different types of nodes: gateway node and relay node. As shown in Fig. 2, $GM1A$ and $GM1B$ are the relay nodes of $group1$ created by $GO1$, while $GO2$ is a gateway node between $group1$ and $group2$.

In the LC/GO connection schema, the communication between two GOs is not allowed because they have the same IP address (i.e., 192.168.49.1). For example, in Fig. 2, when $GO1$ sends a packet to $GO2$, $GO2$ will discard this packet because the source address of this packet is the same to its own address of the P2P interface. And when $GO2$ sends an IP packet to $GO1$, this packet will be sent to the local loop and not to the Wi-Fi interface of $GO2$. Given the above, unicast does not take effect in inter-group communication. To solve this problem, we adopt multicast for inter-group communication, which could specify the interface of sending or receiving multicast packets [7]. Taking the communication between $GO1$ and $GO2$ as an example, $GO2$ needs to specify the Wi-Fi interface to send or receive the multicast packets, while $GO1$ needs to specify the P2P interface.

B. Node Relaying Based Inter-Group Communication

On the basis of LC/GO connection schema, we propose a node relaying based inter-group communication solution. It is realized by constructing specific transport-layer tunnels as depicted in Fig. 1. 1) A GO relaying based approach could be adopted in this solution. The inter-group communication depends on the multicast between two GOs completely. However, the multicast encapsulates a one-to-many unicast communication, its transmission rate is far less than unicast. As a result, the communication between two GOs based on multicast will be a bottleneck when the data size is large. 2) To relieve the bottleneck, we propose a GM relaying based approach. If there are GMs in a group, we will choose a GM randomly as a relay node to relay the traffic toward another group based on unicast. And the multicast between two GOs is only utilized to transmit the member information for inter-

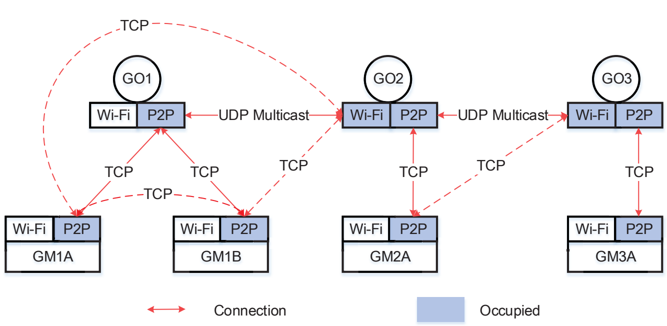


Fig. 1. WFD Based Intra/Inter Group Communication.

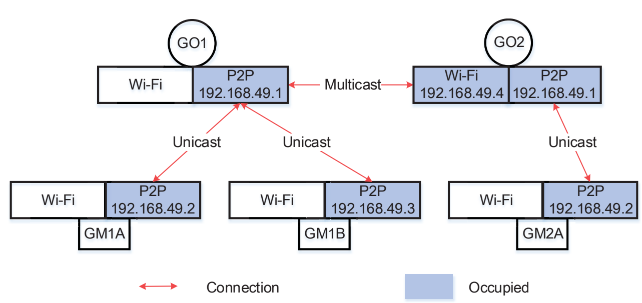


Fig. 2. LC/GO Connection.

group member discovery. In summary, if a group has GMs, the GM relaying based approach will be applied. Otherwise, the GO relaying based approach can be utilized. The proposed solution combines these two relaying approaches to realize the bidirectional inter-group communication. It does not require any GM as the relay node in a group, and also performs inter-group communication without switching frequently between two groups compared to the time sharing strategies [7, 8]. In this paper, we mainly consider and evaluate the GM relaying based approach. Note that the proposed solution could be applied to a large local communication system, which is composed of many groups. However, for the sake of clarity, we take the scenario in Fig. 1 as an example and give a detailed description of the proposed solution as follows.

1) Transport-Layer Tunnel: The UDP multicast is adopted for the communication between two GOs, while the TCP unicast is used for other communication situations. In terms of inter-group communication, the bidirectional communication between relay node (such as $GM1B$) and gateway node (such as $GO2$) can only be realized by TCP. For example, when $GO2$ uses the Wi-Fi interface to connect to $GO1$, $GM1B$ could send packets to $GO2$ by means of TCP or UDP, because it is a intra-group communication between the GM and the LC. However, UDP can not be used in the data transmission from $GO2$ to $GM1B$. This is because $GO2$ will use the P2P interface to send packets by default according to the fact that the priority of P2P interface is higher than the Wi-Fi interface[1]. However, as shown in Fig 1, the data transmission from $GO2$ to $GM1B$ can only use the Wi-Fi interface, not the P2P interface. Through extensive experiments, we find that TCP could solve this problem, since $GO2$ could reuse the previous TCP connection established by $GM1B$. Of course, it requires $GM1B$ to construct a TCP tunnel with $GO2$ in advance.

2) Multicast Based Inter-Group Member Discovery: The inter-group member discovery mechanism is realized by multicast. The purpose is to sense the information of devices in the connected groups. Details of the proposed mechanism are described as follows.

a) Group Connection: When a GO connects to another GO through the Wi-Fi interface, both of the GOs will send out its own GM list through a multicast message using both of the Wi-Fi and P2P interfaces.

b) Multicast Message Processing: The multicast message at the application-layer contains three fields: *messageType* is used to identify the message types, including replacement, adding one member or deleting one member; *senderMACAddress* records the MAC address of the sender device; *members* contains the members' information of a group. When a GO receives a multicast message (due to the MTU limitation, this message may be split into many packets) from another GO, it will extract the application-layer information of the packets. Then, three operations will be performed: 1) The GO records the *senderMACAddress* and extracts the members information from the *members* field. These members will be marked as the members belonging to another group. 2) If the *messageType* is adding one member or deleting one member, the GO performs the corresponding operation and then sends the broadcast message about adding or deleting to other GMs. If the *messageType* is the replacement, the GO first uses the *senderMACAddress* to find the existing members of that group in the member list. Then, it performs an intersection operation between the found members and the received members extracted from the *members* field. At last, the GO updates the member list and broadcasts it to the GMs. 3) The GO encapsulates the received message and sends it out through multicast using both of the Wi-Fi and P2P interfaces. Note that the sender (i.e., the GO of another group) can also receive this message. Therefore, a mechanism is put forward to discard the redundant multicast message. For example, $GO1$ receives the multicast message about the member deletion from $GO2$. After deleting one member from the member list, $GO1$ sends the received message by multicast. Then, $GO2$ can also receive the multicast message through the Wi-Fi interface. Such redundant message can be identified and discarded by $GO2$ through the proposed mechanism.

c) GM Arriving: When a GO detects a new device joining in the current group, it will update the current member list and broadcast the new GM to other GMs within the group (i.e., GM discovery mechanism in intra-group communication). Then, the GO sends a message through multicast about adding the member using both of the Wi-Fi and P2P interfaces.

d) GM Leaving: When a GO detects a GM leaving the group, it will update the current member list and broadcast the leaving GM to other GMs. Then, the GO sends a message through multicast about deleting the member using both of the Wi-Fi and P2P interfaces. However, there are some special cases about GM leaving in multicast message processing. For example, as shown in Fig. 1, when $GO1$ detects $GO2$ leaving the group, it will delete all the members directly connecting to $GO2$ or cascaded connecting to $GO2$, including $GO2$, $GM2A$, $GO3$ and $GM3A$. In addition, these deleted members should be notified to other groups through multicast.

3) Application-Layer Forwarding: In this section, we present a forwarding mechanism at the application-layer. The operations below the application-layer in Android require the root permission, which is beyond the average utilization ability of the users. Additionally, modifications on Android operation system may affect the performance of Android devices and cause some unexpected bugs. Thus, the proposed forwarding mechanism is a general solution without any modifications on Android operation system, which could be applied on off-the-shelf Android devices immediately. However, some security issues, such as privacy protection [9], are not considered in this paper.

The TCP unicast message at the application-layer contains three fields: *messageType* (it is used to identify the intra-group communication or the inter-group communication,), *destinationMACAddress* and *data*. When two devices in different groups wish to communicate with each other, one device sends

[1]We come to this conclusion through some experiments on Android 6.0 or 7.0, which is opposite to the usages in Android 4.X [7, 8].

a message (may be split into one or more TCP segments carried by IP packets) to the corresponding gateway node. Note that when the device can not directly communicate with the gateway node or destination node, the relay node will be used to relay the traffic toward the gateway node or destination node. Then, the gateway node will parse the receiving message at the application-layer and check the destination node through *destinationMACAddress* matching. 1) If the destination node is itself, the gateway node will parse the *data* of the message immediately. 2) If it is one of the GMs within the current group, the gateway node encapsulates the received message and then forwards it to the GM. 3) Otherwise, the message will be forwarded to the next corresponding gateway node. The above process will not terminate until the message reaches the destination node. We use a simple case in Fig. 1 to illustrate the forwarding process.

$GO1 \rightarrow GO3$: 1) $GO1$ finds that the destination node is not within the group and then it sends the received message to the gateway node $GO2$, which is relayed by the relay node $GM1A$ or $GM1B$, since the unicast communication between $GO1$ and $GO2$ is not allowed. 2) $GO2$ finds that the destination node ($GO3$) is within the group. However, $GO3$ is a gateway node and it has the same IP address to $GO2$. They can not directly communicate with each other. Consequently, $GO2$ forwards the message to the relay node $GM2A$. 3) Finally, $GM2A$ delivers the message to the destination node of $GO3$. There are 4 hops at the IP-layer ($GO1 \rightarrow GM1A/GM1B \rightarrow GO2 \rightarrow GM2A \rightarrow GO3$).

IV. PERFORMANCE EVALUATION

We develop a local communication system based on Android studio 2.3.3 by means of WFD technology. In order to validate the performance of the system, we utilize 13 off-the-shelf and non-rooted Android devices equipped with Android 6.0 or 7.0, including 3 HUAWEI Honor 8, 2 SAMSUNG Galaxy C7000 and 8 Redmi Note 4X. All the experiments are conducted in a conference room at night.

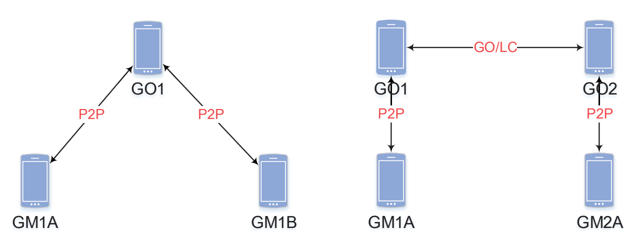
A. Intra-Group Communication Evaluation

We evaluate the performance of the intra-group communication from the aspects of throughput (i.e., average throughput) and packet loss rate. We investigate how the load, distance and interference affect the performance. As shown in Fig. 3(a), three Android devices (Redmi Note 4X) construct a WFD group. We not only evaluate the performance between GO and GM, but also evaluate it between two GMs. We use *iPerf* [10] to conduct the experiments and each experiment lasts 20 seconds.

1) Load: The load means the different sending rates. In this experiment, we only consider the UDP based communication, because the sending rate can not be specified for TCP by *iPerf*. As shown in Fig. 4(a), at the beginning, the throughput increases with the growth of the offered load. Then, the throughput tends to be stable when the offered load is greater than a value (36 $Mbps$ for GO – GM and 24 $Mbps$ for GM – GM). The maximum throughput of GO – GM could reach to 31.7 $Mbps$, while the maximum throughput of GM – GM is only 20.9 $Mbps$. That is to say, the GO – GM (one hop at the MAC-layer) benefiting from a shorter transmission distance at the MAC-layer outperforms the GM – GM (two hops at the MAC-layer) in throughput. Fig. 4(b) shows the results of the packet loss rate, where the packet loss of GM – GM presents an increase with the growth of the offered load (at most 43%), while the packet loss of GO – GM can be almost negligible (below 1%).

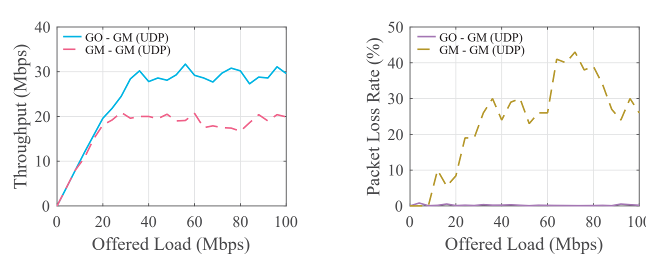
2) Distance: We conduct an experiment to explore the influence of distance. Initially, three devices are located in close proximity of each other. Then, we make the GO move approximately 3 $meters$ every time, while keep the two GMs stay in their place. In addition, the load for UDP is set to 100 $Mbps$, while the load for TCP is the default setting. As shown in Fig. 5(a), the throughout decreases with the increase of distance and the GO – GM is still higher than the GM – GM in throughput. Benefiting from the connectionless characteristic of UDP, the UDP based GO – GM communication has a maximum throughput of 31.4 $Mbps$, while the maximum throughput of such communication based on TCP is 22.3 $Mbps$. However, in Fig. 5(b), the packet loss rate of UDP based GO – GM mode reach up to 36%, which leads to a worse quality of experience.

3) Interferences from Nearby WFD Groups: To explore the interferences from nearby WFD groups, we consider a scenario that several WFD groups coexist. Each interference group is composed of two devices and offers a load of 100 $Mbps$, while the offered load for TCP is the default setting. In total, 23 Android devices are used in this experiment and they are set on the same desk. As depicted in Fig. 5(c), the growth



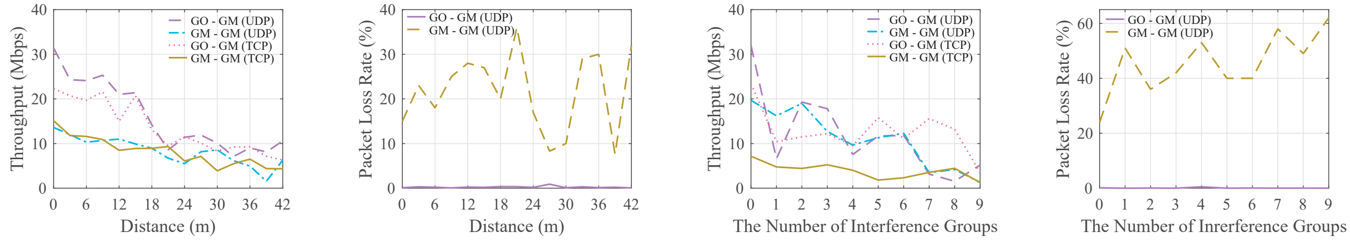
(a) Scenario a. (b) Scenario b.

Fig. 3. Evaluation Scenarios.



(a) Throughput. (b) Packet Loss.

Fig. 4. Load.



(a) Throughput. (b) Packet Loss. (c) Throughput. (d) Packet Loss.

Fig. 5. Distance and Interference.

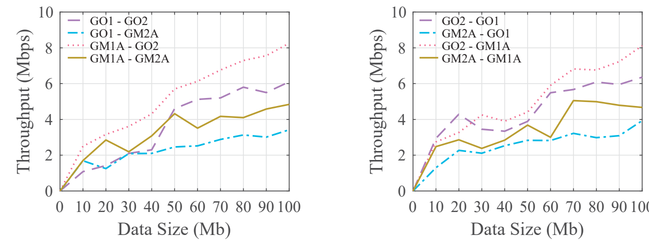
of the interference groups causes the decrease of throughput. TCP based GO – GM has the better performance when the number of groups is greater than 4. Under the 9 interference groups, the throughput of all communications are less than 6 $Mbps$ due to the severe channel interference. Fig. 5(d) shows the packet loss rate, where the interference influence on GM – GM is notable (up to 62% under 9 interference groups) while such influence on GO – GM is negligible (below 0.12%).

From aforementioned experimental results, we draw three conclusions. 1) Due to the shorter transmission distance at the MAC-layer, the performance (i.e., throughput or packet loss rate) of GO – GM communication outperforms the GM – GM communication. 2) Throughput of UDP is a little higher than that of TCP owning to the connectionless characteristic of UDP. However, UDP presents a relatively severe packet losses. Therefore, TCP may be a preferred solution in some use cases. Nevertheless, the reliable UDP is also acceptable. 3) In order to reduce the interference, the number of coexistence groups should be controlled. In addition, some mechanisms to reduce the channel collisions are needed.

B. Inter-Group Communication Evaluation

Fig. 3(b) shows the experimental scenario of evaluating the inter-group communication. Four devices (Redmi Note 4X) are used to construct two WFD groups and the proposed LC/GO connection schema is utilized to enable two GOs to interconnect with each other. Based on the application-layer forwarding mechanism, the packet forwarding does not follow the routing rules utilized in traditional networks. As a result, *iPerf* can not be adopted to evaluate the performance of the inter-group communication. Given the above, we control the sending data size of the sender and obtain the application-level throughput from the receiver. The throughput is computed through the amount of received data divided by the time starting from the first packet arriving to receiving the last packet. To reduce the experiment errors, we repeat the experiments for 10 times.

Fig. 6(a) and Fig. 6(b) depict the application-level throughput of bidirectional inter-group communication. In Fig. 6(a), the $GM1A$ – $GO2$ (i.e., the data transmission from $GM1A$ to $GO2$) reaches up to 8.26 $Mbps$. Other communications reach a maximum value of 6.1 $Mbps$ ($GO1$ – $GO2$), 4.84 $Mbps$ ($GM1A$ – $GM2A$) and 3.42 $Mbps$ ($GO1$ – $GM2A$). These results conform to the fact that the throughput is related



(a) From Group1 to Group2. (b) From Group2 to Group1.

Fig. 6. Inter-Group Communication.

to the transmission distance, e.g., as shown in Table I, the $GM1A$ – $GO2$ requires one hop at the IP-layer and two hops at the MAC-layer while the $GO1$ – $GM2A$ requires three hops at the IP-layer and four hops at the MAC-layer. Therefore, the $GM1A$ – $GO2$ outperforms the $GO1$ – $GM2A$ in throughput due to the shorter transmission distance. Fig. 6(b) shows the throughput of another directional communication, which is similar to Fig. 6(a). Thus, the proposed inter-group communication solution could achieve the similar performance in bidirectional communication.

V. Related Work

Wi-Fi Direct [11] is a D2D technology presented by Wi-Fi Alliance, defining direct Wi-Fi connections between devices. Through WFD, we can construct an infrastructure-less peer-to-peer (P2P) network with robust Wi-Fi Protected Access 2 (WPA2) security and typical Wi-Fi data rate and range. The mobile device could be a group owner or a group member (P2P client) within a WFD group. In addition, the legacy client that is not compliant with the WFD specifications is also supported. The evaluations of the native characteristics of WFD were given by Camos-Mur et al. [12]. Some optimizations on WFD were conducted by Sun et al. [13] who proposed a simple and efficient scheme to optimize the device discovery mechanism of WFD, and Zhang et al. [14] who presented an improved WFD group formation protocol.

The intra-group communication is the native characteristics of WFD, whereas inter-group communication is beyond the scope of WFD specifications. Few studies focus on such communication. Casetti et al. [8] proposed a client relaying based approach, while an extra relay node was needed and the

TABLE I
TRANSMISSION PATH

Communication	*IP-Layer*	*MAC-Layer*
$GO1$ – $GO2$	$GO1 \rightarrow GM1A \rightarrow GO2$	$GO1 \rightarrow GM1A \rightarrow GO1 \rightarrow GO2$
$GO1$ – $GM2A$	$GO1 \rightarrow GM1A \rightarrow GO2 \rightarrow GM2A$	$GO1 \rightarrow GM1A \rightarrow GO1 \rightarrow GO2 \rightarrow GM2A$
$GM1A$ – $GO2$	$GM1A \rightarrow GO2$	$GM1A \rightarrow GO1 \rightarrow GO2$
$GM1A$ – $GM2A$	$GM1A \rightarrow GO2 \rightarrow GM2A$	$GM1A \rightarrow GO1 \rightarrow GO2 \rightarrow GM2A$

data transmission rate was different in bidirectional communications between any two nodes. Funai et al. [7] given several possible connection schemas for inter-group communication and mainly presented three solutions to realize such communication, including time sharing, UDP multicast and hybrid. However, the UDP multicast and the hybrid solutions need to ensure that there is at least one group member within the group. And the overhead of the time sharing approach is hard to tolerant. In addition, these studies were conducted based on the older Android versions (Android 4.4.2 or earlier). Some mechanisms of the Android have been modified and improved. Consequently, their designs were not able to be applied on off-the-shelf Android devices.

VI. CONCLUSION

To cope with the possibility of infrastructure failures, as well as meet the needs of executing the collaboration tasks in mobile edge computing, we develop and evaluate a local communication system using the WFD technology. We not only present an intra-group communication based on the native characteristics of WFD, but also propose a general solution for the inter-group communication supported by the current implementation of WFD. Experiment results show that the maximum throughput of intra-group communication could reach up to 31.7 $Mbps$, and the maximum throughput of inter-group communication is 8.26 $Mbps$.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China under Grant Nos. 61602105 and 61572123; China Postdoctoral Science Foundation under Grant No. 2016M601323; the Fundamental Research Funds for the Central Universities Project under Grant No. N171604006; CERNET Innovation Project under Grant No. NGII20170121; the Major International(Regional) Joint Research Project of NSFC under Grant No. 71620107003; the National Science Foundation for Distinguished Young Scholars of China under Grant No. 71325002; the Program for Liaoning Innovative Research Term in University under Grant No. LT2016007.

REFERENCES

[1] *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016-2021 White Paper*, Cisco, March 2017.

[2] OpenGarden, "Firechat," https://www.opengarden.com/.

[3] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: architecture, applications, and approaches," *Wireless Communications & Mobile Computing*, vol. 13, no. 18, pp. 1587–1611, 2013.

[4] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017.

[5] A. Ahmed and E. Ahmed, "A survey on mobile edge computing," in *International Conference on Intelligent Systems and Control*, 2016.

[6] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.

[7] C. Funai, C. Tapparello, and W. Heinzelman, "Supporting multi-hop device-to-device networks through wifi direct multi-group networking," *arXiv preprint arXiv:1601.00028*, 2015.

[8] C. Casetti, C. F. Chiasserini, L. C. Pelle, C. Del Valle, Y. Duan, and P. Giaccone, "Content-centric routing in wi-fi direct multi-group networks," in *2015 IEEE 16th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. IEEE, 2015, pp. 1–9.

[9] M. Shen, B. Ma, L. Zhu, R. Mijumbi, X. Du, and J. Hu, "Cloud-based approximate constrained shortest distance queries over encrypted graphs with privacy protection," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 4, pp. 940–953, 2018.

[10] "iperf," https://iperf.fr/.

[11] "Wi-fi peer-ot-peer (p2p) technical specification v1.7," WiFi Alliance, Tech. Rep., 2016.

[12] D. Camps-Mur, A. Garcia-Saavedra, and P. Serrano, "Device-to-device communications with wi-fi direct: overview and experimentation," *IEEE Wireless Communications*, vol. 20, no. 3, pp. 96–104, 2013.

[13] W. Sun, C. Yang, S. Jin *et al.*, "Listen channel randomization for faster wi-fi direct device discovery," in *IEEE INFOCOM 2016 - IEEE Conference on Computer Communications*, 2016, pp. 1–9.

[14] H. Zhang, Y. Wang, and C. C. Tan, "Wd2: an improved wifi-direct group formation protocol," in *ACM MOBICOM Workshop on Challenged Networks*, 2014, pp. 55–60.