

An Android Application on Wi-Fi Direct

Pooja M P, Pooja V S, Poonam S, Yashaswini K K , Rajshekar M B*

Department of Computer Science and Engineering, GSSS Institute of Engineering and
Technology for Women, Mysuru, India

DOI: <https://doi.org/10.21467/proceedings.1.17>

* Corresponding author email: rajashekar@gsss.edu.in

Abstract

The number of smartphone users in modern times have skyrocketed enabling a multitude of technologies to house themselves over these devices. With the introduction of Wi-Fi Direct (WiFi P2P) into the Android platform, devices are offered the capability to connect to and exchange content over a fast and reliable wireless connection. Some scenarios may require the ability to set up a temporary communications network on portable devices over the fly, regardless of their locality. To the end, the objective of this paper is to bring about a custom Wi-Fi Direct adaptation to support the creation of a multi-hop like network that enables users to communicate information across multiple Android devices.

Index Terms- Multi-hop, P2P connection, Wi-fi Direct, Wireless connection.

1 INTRODUCTION

The modern era showcases a shift from wired to wireless technologies as more and more consumers are mobile and need access to information on the fly. The year 1997 marked the release of the 802.11 protocol (now known as 802.11 legacy), which offered a transfer bit rate of 1 or 2 Mbit/sec and included forward error correction code which enabled minimal data loss over noisy channels. This was followed by the introduction of two new protocols, namely 802.11a and 802.11b two years later which were more consumers-focused. The 802.11g protocol came into existence in 2003 and offered the best of both worlds as it adopted the advantages of both the 802.11a and 802.11b. The more mainstream 802.11n protocol came into existence in 2009 and is by far the most adopted protocol to date.

In this paper we extend WIFI P2P API to support multi-hop ad-hoc networking and to enable two devices to connect and communicate with each other through a series of intermediate nodes. There may exist a scenario where devices may not require to be connected to the internet, but rather connected to other devices within its radius. For example, a laptop and a smartphone may want to directly connect to each other for a file transfer. To enable such a connection, we may need to exploit each device inbuilt hardware such as the WiFi card. By making use of Wi-Fi, we can now set up a direct connection between two devices without the existence Wi-Fi of an infrastructure. Each device, now called a node can sense and connect



© 2018 Copyright held by the author(s). Published by AIJR Publisher in Proceedings of the 3rd National Conference on Image Processing, Computing, Communication, Networking and Data Analytics (NCICCNDA 2018), April 28, 2018. This is an open access article under [Creative Commons Attribution-NonCommercial 4.0 International](https://creativecommons.org/licenses/by-nc/4.0/) (CC BY-NC 4.0) license, which permits any non-commercial use, distribution, adaptation, and reproduction in any medium, as long as the original work is properly cited. ISBN: 978-81-936820-0-5

to its neighbouring node for data transfer. This concept may also be extended to multiple devices to form a link of devices where each node plays the role of a router and can forward a packet to its neighbouring node until the intended recipient receives the packet. To this end, there are a variety of possibilities to enable such a network under the user's disposal. These include:

Wireless ad hoc networks (WANETs) does not rely on an existing infrastructure such as routers and other network devices for data communications. It follows a decentralized approach in which each node can configure and update itself to match the mobile nature of each node involved in the network.

Mobile ad hoc networks (MANETs) make use of the 802.11 protocol to aid in the creation of networks that are mobile in nature. Wi-Fi Direct which provides a single-hop solution to creating a communication link that operate at high speeds. As such, the application proposed in this paper can enable a user to adopt an ad-hoc chain that can support multiple nodes in the network creation process. Information sharing over this network can be supported locally rather than over the network.

2 MODE OF OPERATION

To establish and enable data communications within a group, we make use of “modes” that acts as identifiers that dictates the appropriate actions that are to be performed. This enables the Group Owner or Peer to take the appropriate action for the message being broadcasted. It is to be noted that all information within a group are passed to its group members through the group owner. The order in which each mode is invoked is discussed below:

Mode_Setup:

This mode is activated when a new device joins the local group. The peer that joins the network forwards a randomly generated number that is appended to the text ‘andy’. This essentially serves as a unique identifier for that device. The device's MAC address is also forwarded to the GO, which is essential for communicating outside of the group (discussed below). On the GO's end, the GO makes use of the socket connection with this new peer to capture it's local IP address. Both the ID and the IP address are pushed onto a hash table, and another table containing the ID and the MAC address is also updated.



Figure 1: Information exchange on group formation

Since these IP addresses are necessary for communications within the cluster, the hash table serves as a means to obtain the IP address based on the device's ID. Once this process is complete, the GO sets up the "acceptObj" mode as described below.

Mode_acceptObj:

Once the above mode has played its course, each device, excluding the GO switches over to server sockets. The GO, with the use of a client socket, retrieves a list of all the peers that are connected to the group and forwards a copy of the hashtable to every peer in an iterative fashion. This step is necessary for data transmission outside the group and is described in the "scatter" mode that is to come.

Mode_transfer:

This mode is used for data transmission within the group. Once the sockets are set up, the peer sends the ID and message with the mode to the GO. The GO first retrieves the IP address of the ID provided and checks if it matches with the IP '192.168.49.1', which is the default IP address of the GO. If so, it indicates that the message is meant for the GO and no further actions need to be taken. If the IP address differs, the GO sets up a connection to its intended recipient and transfers the message to that peer. Since every peer defaults back to using server sockets, this process is seamless.

Mode_scatter:

Since every peer has a list containing the IDs of the devices within the group, if the ID supplied by the user does not match with any of the entries in the local list, the "scatter" mode is activated. We first send the local GO the intended message with the mode set to scatter. The GO then sends the same message to every peer that is connected to the group. Once this message is received by an individual peer, the application gets an instance of the WifiP2pConfig class and with the list of available devices, runs a check if any of the MAC address (obtained from the config() method) exists in the local hash table. If so, we ignore that device as they exist in the local group. However, if the MAC address does not exist in the local table, the peer calls the disconnect() method to disconnect itself from the current GO and follows a call to the connect() method with the new MAC address. Once the connection is established, we send the message to this new GO and also send the hash table to prevent any unnecessary loopbacks.

It is to be noted that when the device outside of the group is connected to, it becomes a GO of its own and the connection that the device had to the original group is temporarily severed. Once the data is successfully transferred, a string containing the original GO's MAC address is used as a means to reconnect with that GO. To ensure that we don't overwrite the string containing the original GO's address, a Boolean variable essentially serves as a lock to ensure that this string isn't overwritten at any point of time in the future. Once each device reconnects to its original GO, the peer from Group-2 uses its modes to retransmit the information within

the group. Again, if group-2 doesn't contain the destination ID, the scatter mode is called once more.

3 RELATED WORK

Zhuang, T et al [1] presents a custom middleware that runs as a standard Android Service and as its own process over the user space. The server acts like any other Android application and can be installed or uninstalled by the user. More importantly, the middleware introduces no kernel modification as it is built by implementing the existing transport protocol APIs provided by Android. The proposed application communicates with other applications that could be running in the background to avoid any resource conflicts that might occur.

Wirtz, H et al [3] created a small core of mobile router nodes, providing an infrastructure mode network access to mobile station nodes by which a multi-hop communication was made achievable. A study was done to establish and maintain MANET like multi-hop ad hoc networks that support unmodified 802.11 mobile devices. Continuous communication between arbitrary pairs of devices was achieved. The traditional 802.11 family of smartphones possess certain limitations in that they only support single hop networking due to the lack of native protocol support and showcase a degraded download speed. The paper demonstrates a practical application of MA-Fi for continuous communication between devices by integrating mobile devices in a network with the roles of RONS and STANs. Additional participants with 802.11 compliant devices were also invited to join the network.

Thomas, J et al focused on providing a backup communications service during the time of crisis. The project utilizes MANET to allow smartphones to link to create a communications network with each other in a dynamic manner. The devices used to provide the ad hoc network were configured to behave in both ad hoc mode and managed mode (traditional infrastructure mode). Tests were run using the Asus Eee Pad, Samsung Galaxy SII Epic Touch and the Samsung Galaxy Nexus. Once a connection was formed, devices could communicate with each other and share access to the internet through gateways. This was done to simulate a scenario where one group owner obtains information over the internet that he/she then forwards to everyone else in the cluster.

4 CONCLUSION

The conclusion is, while the traditional infrastructure enabled wireless communications do have their role, and will most likely stay their place as the dominant form of information communications over mobile devices, WiFi Direct does have its place in respect to specific scenarios. The WiFi P2P API on Android does support a single hop connection and can be used for one-on-one file sharing or wireless streaming. WiFi Direct can be extended to support multi hop connections too. Doing so will enable smartphone users to create a network of smartphones, each running the proposed application. As such, a large, local network can be created and utilized for communications or data transfer and such a network can prove to be a great asset to scenarios such as telemedicine, disaster control, etc.

REFERENCES

- [1] COLIN FUNAI, CRISTIANO TAPPARELLO, WENDI HEINZELMAN proposed on Enabling multi-hop Ad Hoc networks through wifi-direct multi-group networking.
- [2] Lakhdar Meftah, Maria Gomez, Romain Rouvoy and Isabelle Christment proposed on Testing wifi peer-to-peer mobile apps in the large.
- [3] DANIEL CAMPUS-MUR, NEC NETWORK laboratories Andres Gracia-Saavedra and Pablo Serrano, Universidad Carlos III de Madrid proposed on Device-to-Device Communications with Wifi Direct.