

Computer Network

Songtuan Lin u6162630

May 1, 2019

1 Basic Terminology

Network, particularly in terms of Computer Science, is consist of several devices that are able to *communicate* with each other. The term *communicate* here means the device within Network can send or receive message from other devices. In order to perform this communication, each device within Network must be able to be located, or simply speaking, each device must have a *address*. There are several methods to assign *addresses* to devices, each of them corresponding to a specific technology that is used to construct Network. One popular technology is *Internet*, which assign a *IP* to each device within Network as their address. In here, *Internet* is the technology that used to construct Network and *IP* is the method that Internet used to address(locate) devices within it.

Furthermore, in order to let devices within Network communicate with each other, one essential requirement is that there must be a thing that can carry the message and deliver it between devices. Such thing, which can be called as *media*, consist the *Physical Layer* of Computer Network

2 Physical Layer

As described in last section, *Physical Layer* is consist of the media that can carry and deliver messages. Particularly, as Figure 1 illustrated, the media is the line that connect Device A and B. Furthermore, based on the



Figure 1: Media between Different Devices

characteristics of the media, it can be categorized into following three classes::

1. **Simplex:** This type of media can only transport message with a *fixed direction*, e.g. from Device A to Device B.
2. **Half-Duplex:** This type of media can transport message with both directions. However, it can only choose one direction during each transportation, e.g. from A to B or B to A.
3. **Duplex:** This type of media can transport message with both direction during a transportation, e.g. transport message from A to B and B to A in the same time.

Indeed, the Figure 1 give us an inspiration about how to construct a simple network, which is connecting each devices within network with a media. For example. assume we want to construct a Network contain four devices, the easiest way to do this is shown in Figure 2, in which, each device is connected with the rest. However, the problem of this topology is that the number of medias will grow exponentially with the increasing number of devices. Therefore, an improvement topology along with an algorithm(protocol) to control how the messages are delivered is required, which relate to our next topic: *Data-Link Layer*

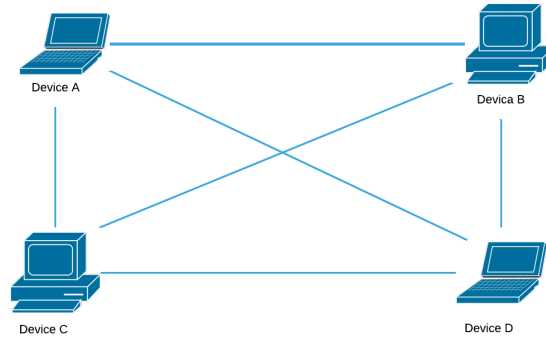


Figure 2: Basic Topology

3 Data-Link Layer

The solution of the problem remaining in last section is straightforward: By adding one or more *Switch* which depend on the number of devices within network. The Switches work as bridges that connect different devices in Network, as illustrated in Figure 3: Even though the switches reduce the number of connection, the new

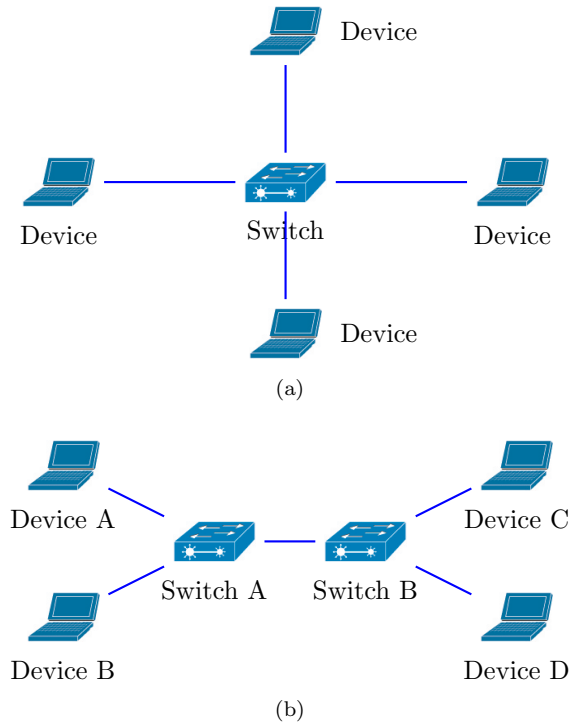


Figure 3: Two Example of Network Topology

problem is how the the different devices communicate with each other under the condition that they are not direct connected. This problem may be better illustrated by an example: Assume Device A in Figure 3b is willing to send a message to Device D. The first thing Device A need to do is locating Device D, or we can say, Device A need to find the *address* of Device D. The address here is called *MAC address* and in order to find Device B's MAC address, Device A will *broadcast* a request message which contain a query: "What is the MAC address of Device B" and it's own MAC address. The *broadcast* here means A will send this query to all the devices within this Network(Device B, C and D). In order to indicate this query is a broadcasting message, Device A will set the destination MAC address of this query as a special value: FF-FF-FF-FF-FF-FF and send it to Switch A. When Switch A receives this query and recognizes this message is a broadcasting message as it has FF-FF-FF-FF-FF-FF as destination MAC address, Switch A will send this query to every device it connect to, include both Device B and Switch B. Furthermore, when Switch A receive this query, it will record the source MAC address of this query(Device A's MAC address) along with the index of port that receive this query into a table, e,g Switch A will record the pair:

Device A's MAC address	Port 5
------------------------	--------

into the table if the query message come through port 5 of Switch A. This table is called *ARP table*. By maintain such a table, when Switch A receive a message which has Device A's MAC address as destination, it can directly forward this message to Device A through port 5 by searching Device A's MAC address along with the corresponding port within ARP table. After Switch A broadcast the query, Switch B and Device B will both receive this query message. Device B will realize that Device A did not ask its' MAC address and will simply discard this query message without further action. By contrast, Switch B will keep this broadcasting which finally cause Device D receive this query. One thing that need to be noticed is that Switch B will also record Device A's MAC address and the corresponding port index into ARP table by reading the source MAC address within query message. When Device D receive this query, it will response Device A with a respond message which use Device A's MAC address as destination address, its' own MAC address as source address and carry its' MAC address as query result. This respond message will be transferred back to Device A through Switched with the Device D's MAC address(source address) be recorded into each Switch's ARP table. After Device A receive the MAC address of Device D, it can then send the further message by setting Device D's MAC address as destination address. The Switch can also know which port should be used to send these messages by searching the ARP table to find Device D's MAC address.

It can be seen from the above procedure, in order to perform message delivering within this Network topology. The messages must have some specific content, *e.g.* source address and destination address, or without loss generality, we can say, the messages that being transferred under this mechanism must have s specific format. This message delivering mechanism with specific message format is called *protocol* and the message delivering mechanism within the Network topology described above is called *ARP Protocol*.

Indeed, we can give the messages that flowing within this Network a name: *Ethernet Frame*. According to above discussion, we know that Ethernet Frame must follow a specific format. The exact Ethernet Frame format is:

Preamble	Destination Address	Source Address	Type	Actual Message Content	CRC code
----------	---------------------	----------------	------	------------------------	----------

In which, the most two important field: Destination Address and Source Address, have been introduced in the previous part.

The most content we discussed in these two sections is based on the wired connection, or we can say, use physical line as media in Physical Layer. However, in recent years, the wireless connection become more and more popular. By using wireless connection, one device can communicate with the other without using physical line. Indeed, this wireless connection technology is not as fancy as many people think. In wireless network, the devices use wireless connection connect to a *base station* and this base station will connect to the normal wired network topology as described before. So, an example wireless network is: In which, the dashed line

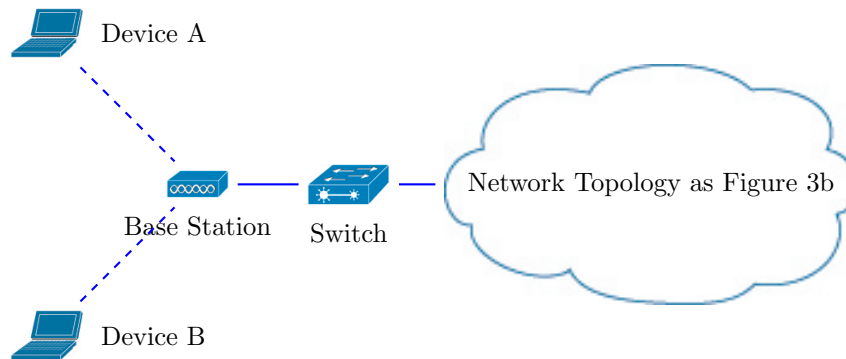


Figure 4: Wireless Network Topology

between Device A, B and Base Station denote the wireless connection. The name of base station and the function it provide may varied depend on what wireless communication protocol it is used. Particularly, in *802.11 protocol*, which is the protocol we will focus on, the base station is given name: *access point* or *AP*. Furthermore, since wireless connection is more unstable than wired connection, the majority wireless protocols, include 802.11, require message receiver to send a acknowledgment back to message sender so that the sender can know this message has been successfully received. Before we move to more detail of 802.11 protocol, there is one more problem that we need to realized: In Figure 4, even though we only draw the wireless connection between Devices and Base Station, the message sent or received by one device will indeed being received by all the others, which means, the message sent by one device will be broadcast to all the other device that connect to the same base station. Hence, the equivalent topology of wireless connection can be illustrated as Figure 5, which indicated these devices are actually sharing a communication line. The problem of this connection is straightforward: When there are more than one device are sending messages through this line, these messages

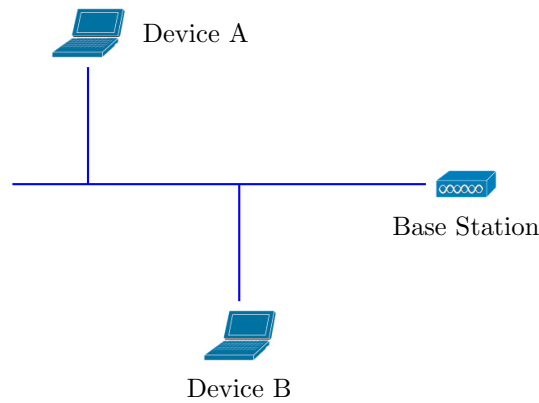


Figure 5: Equivalent Wireless Connection

will have collision with each other, and hence cause the messages be corrupted. The solution of this problem is a protocol called *CSMA/CA* stand for *Carrier Sense Multiple Access with Collision Avoid*. The CSMA/CA protocol work as follow: When a device have a message to be sent, it will:

1. Check whether the line is idle(no message is currently transferring on it). If the line is idle, the device will start to transfer message after a fixed small period of time known as *Distributed Inter-frame Space*(DIFS).
2. If the line is not idle, the device will choose a random delay time and set up a timer to count down this value. One thing that need to be noticed here is the timer only start to count down when the device senses the line is idle, which means, the timer will freeze if there are messages being transferred through the line.
3. When the timer count down to zero, the device will start to transfer the message and wait to receive the acknowledgement.
4. If the acknowledgement is received, the device know this message has been received. Otherwise, device will go back to step 2 and choose a delay time to prepare to re-transmit this message.

The reason for using DIFS is that: There will always be a transmission delay between Device A and Device B which caused by the distance between A and B. Hence, when Device A start to transfer a message, Device B will sense the line is idle until the message reach the location of B. Hence, if B start to transfer message at this time period(the time that the message hasn't reach B but the line is actually not idle), the collision will happen. The DIFS is designed to try to avoid this time period and hence, further decrease the probability that collision happen.

After we have described some fundamental mechanism of wireless connection, we can now start to discuss some detail about 802.11 protocol. The most important question is: How the device connect to AP(base station). In order to connect to a AP, the first thing a device need to do is to figure out which AP it will connect to, or in more generally, device need to know the identification of each AP so that it can choose which one to connect to. In 802.11, each AP is identified by a *Service Set Identifier*(SSID) and will be used by device as its' identification. Even though the SSID provide a method that allow device to know the identification of each AP, device still need to know the list of APs(identified by SSID) that it can connect to and then establish the connection, 802.11 provide two mechanisms to achieve this goal:

Passive Scanning : The procedure of passive scanning is:

1. Each AP will broadcast a *Beacon Frame* which inform the devices within this area with its' SSID and told them it is willing to be connected.
2. Each device will receive multiple Beacon Frame and hence get a list of APs it can connect to. The device will select one of them and send a *Request Frame* to request establishing a connection with this AP.
3. The chosen AP will respond with a *Response Frame* to confirm the connection is established.

Active Scanning : The procedure of active scanning is:

1. Each device will broadcast a *Prob Request Frame* to each AP to request APs' SSID.
2. When AP receive this Prob Request Frame, it will respond with a *Probe Response Frame* which indicate its' SSID.

3. The device will get a list of available APs by receiving multiple Probe Response Frame. The device will choose one AP and send a Request Frame to request establishing connection.
4. The chosen AP will respond with a Response Frame to confirm the connection is established.

4 Network Layer

The Network Topology described in above section is usually called *Local Area Network*(LAN). LAN is actually a successful Network that can allow different devices communicate with other. However, the problem raise with the increasing number of devices, especially when some part of devices may have very long distance with the others. The topology of LAN is definitely not suitable for this large scale Network for the reasons:

1. The large number of Switched are required so that them can connect all the devices, which will cause the Network Topology become missive and raise a lot of problems such as *Broadcasting Storm*.
2. The large of number of long wire is required to connect the devices that cross long distance, which increases the cost significantly.

In order to address these problem,