Factors affecting speeds of networks about us

Ross Maloney

December 17, 2016

Abstract

A local area network consisting of a Toshiba laptop, a Mac mini, a Mac Pro cylinder, and a Netgear NAS, all running Linux, were used to measure behaviour of data flow between networked devices. A file orientated approach was used using data transfer speed as the measure. Download and uploads of multiple files of different sizes were used and their individual and relative behaviour determined. Thin wire ethernet and WiFi at 2.4Ghz and 5.0GHz network connections were studied. Powerline devices as a means of extending the network were examined. Finally, network parts were used to examine speed behaviour of an Australian NBN FTTP 100/40 service.

1 Introduction

Local area networks are becoming common both in the home and in small businesses. The component parts which form such networks are sold at general retail outlets. Many claims are made about the speed with which data can be transferred through those networks. But how true are such claims or what conditions need to be place on those claims for them to remain true?

Better grade devices have giga-bit ethernet interfaces for connecting to hardwired ethernet networks. So what data transfer speed can be expected? This device has dual-band AC 1900 WiFi. Does this give better network speed than WiFi to 802.11 n and how does that speed change with distince between the source and the receiver of the wifi? Answers to such questions are impoortant but difficult to find particularly with quantitative data to support the answers. Without the supporting data, answers can be open to question.

Table 1: Devices used in the test network

Make	Model	Descriptiom	Name assigned	Network Address
Netgear	GS724v4Tv4	Switch	Switch	`192.168.10.155`
D-Link	DAP 1650	Range Extender	Range Extender	`192.168.8.240`
D-Link	DSL-2900AL	Viper Model Router	Viper	`192.168.8.244`
Toshiba	Satellite Pro P100	Centrino Duo laptop	Toshiba	`192.168.8.5`
Apple	Mac Mini Server 2011	2 HD server	macMini	`192.168.8.7`
Apple	Mac Pro 2013	6 core standard	macPro	`192.168.8.9`
Netgear	ReadyNAS 316	4x2T HD RAID-5	Array 1	`192.168.8.107`

The research reported here is aimed at exploring networking over giga-bit ethernet, powerline, and WiFi in the 2.4 GHx and 5.0 GHz frequency bands together with the influence of distance between WiFi transmitter and receiver. Both downloading and uploading speeds are measured. A laptop and two desktop computers were used together with two netword access file servers. All devices ran on the Linux operating system. Details of each device used in this research is given in Table 1.

2 A baseline network speed

All the data transfers in this reasearch used the TCP/IP protocol encapsulated in ethernet frames. All IP addresses were 32 bit. Both hardwired, be they Giga-bit ethernet connections or powerline, and WiFi used such a frame. A Giga-bit is 128 Mega-bytes. Bytes are the usual measure of computer files. A Giga-bit network transports all the bits of an ethernet frame: ethernet headers and user data. The interest is not the rate of bit transport across the network but the user data transfer rate.

Table 2: Make up of an ISO/IEC 8802-3:1993 standard ethernet frame

Component	Size
preamble	7 bytes
start of frame	1 byte
destination address	6 btyes
source address	6 btyes
data length	2 btyes
data	38 - 1492 bytes
frame check	4 bytes
interpacket gap	12 bytes

The ethernet frame shown in Table 2 formed the Data Link layer of the network. This Table shows the overhead of an ethernet frame is 38 bytes and it's maximum length is 1530 bytes. Into the data field of this frame the IP packet was inserted by the networking software to form the Network layer. The IP packet consisting of a 20 byte header and a data field which filled the ethernet frame data field. Into this IP data field the network software inserted the TCP packet to provide the Network Transport layer. The TCP packet consisted of a 20 byte header with the remaining space filled with user data. In both the IP and TCP packets the option field was not used. Data transferred between computers was stored in the data field of the TCP packet. So, the possible data tranferred with each ethernet packet was in the range 0 to 1452 bytes. Each ethernet frame contained 78 bytes of overhead consisting of ethernet address information, frame preamble, IP address information, TCP information, etc.

All data in this research used application program scp to transfer the data.

RFC 4253 (The Secure Shell transport Layer Protocol) states (page 5) the overhead introduced by the SSH packet format is not significant for large packet sizes. There is no RFC relating to scp. Instead, scp is implemented using ssh packets. Figure 3 shows the ssh packet. From this Figure, the overhead of each ssh packet is 21 bytes.

Table 3: Make up of a RFC 4253 SSH transport packet

Component	Size
packet length	4 bytes
padding length	1 byte
data carried	7 to 32768 bytes
padding	4 to 255 btyes
Message Authentication Code	12 byte recommended

The ssh overhead reduces the maximum data which can be held in each ethernet frame during a scp copy to 1431 (=1452-21) bytes. In copying a large file, maximum ethernet packets were used (except of the last frame of the transfer). So a scp copy over TCP/IP uses 93.5% (=1431/1530) of the network's capacity when using standard ethernet frames. If the network capacity is 1 Gbit/s, then 935 Mbits/s of data can be handled, which is equivalent to 117 MB/s.

There is overhead introduced by computing involved in setting up each packet at the sender's end for transmission and unpacketing and storing at the receiver's end. However, these computing times are orders of magnitude smaller than a 1Gbit/s network transmission speed and are negligible for the current work.

3 Experimental setup

The experimental setup consisted of three parts; devices, networks, and files. The devices were those shown in Table 1. These devices formed the hardwired and wireless networks used in the trials. The ethernet interface on each computer and NAS array was used to form the hardwired ethernet networks. The Viper router and Range Extender were used to form the WiFi network.

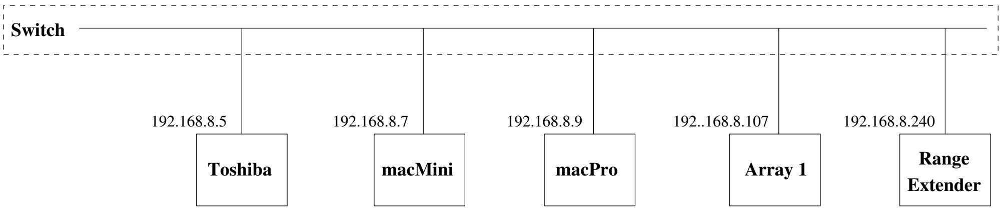


Figure 1: Configuration for testing hardwired networking

The ethernet connection of each device was through the Netgear switch. The Range Extender was also connected to the Netgear switch. This is shown in Figure 1. A powerline interface was connected to the switch. Each connection between a device and the Netgear switch was via a Cat 5E thin-wire ethernet cable with a RJ45 clip on its ends.

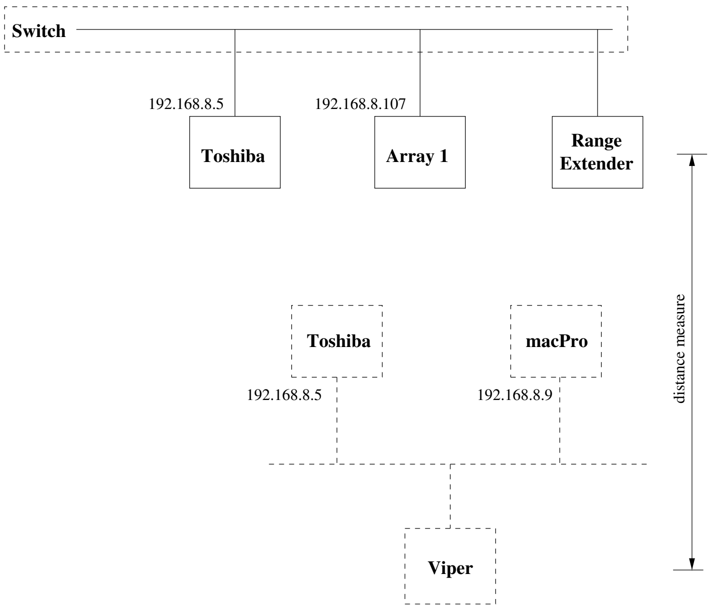


Figure 2: Configuration for testing wireless networking

The Range Extender was used as a hardwired-to-wireless network interface for the devices connected to the switch. It was set in wireless `repeater` mode. The network address of the Range Extender was not set into the network configuration the switch nor any of the devices connected to the switch.

Sufficient data for the purpose of addressing the questions considered in this research could be obtained without using all devices shown in Figure 1. Since wireless network transfer of files in slow compared with that of hardwired networks this reduced considerably the time needed to gather the data. Figure 2 shows those used. The Toshiba and Array 1 devices were connected to the Range Extender by the Netgear switch. The Viper wireless router was directly connect by thin-wire ethernet cable to the device. Only one device was connected to the Viper at a time.

A selection of files were copied across network combinations of the devices of Table 1. These files were collected into directories and all files in a directory were copied between a device pair. The characteristics of the files in each directory are shown in Table 4. Each file was a video show stored in one of two standard video formats as shown in the table. An identification index was assighed to each directory of files for use in the naming of the data file produced using that file in the experiments. Table 4 shows indices 4 and 5 to be missing. They were removed while the experiments were being performed which was after the initial index assignment. Sufficient data was gathered without the need to used those directories of files initially thought to be required.

Table 4: Summary of files in directories copied and their data indicies

directory	Number s of files	File type	Total [MB]	smallest [MB]	median [MB]	mean [MB]	largest [MB]	Index
atkh	190	wmv	50,523	66.3	230.8	265.9	801.5	1
18only	136	mp4	74,602	173.7	551.9	548.5	1075.0	2
brcc	80	wmv & mp4	69,257	197.6	666.5	769.5	1701.0	3
yanks	246	mp4	42,652	6.8	170.1	173.4	541.5	6

Data in Table 4 characterize each of the file collections transferred in these trials. Each collection was a directory of individual files. The data give in the table were obtained by running the systen command `ls -l` on each directory and directing the output into a file. Using an editor, the column containg the file sizes was selected and copied to another file. On the first line of this new file, a line containing the word `size` was added to indicate the name of the variable contained in the file which R was to associate with column of data. This file was then processed by the `R` statistical package. Assuming the name of this file was `length.data`, the command sequence used in R to obtain the data was:

```
datafile <- read.table("length.data",header=T)
attach(datafile)
size
summary(size)
```

This edit/process combination was repeated seprately for each directory listing.

Table 5: Summary of the directory used on each device and it's maximum size

Device	directory	Size
Toshiba	`/home/trial`	92 MB
macMini	`/home/trial`	131 MB
macPro	`/home/trial`	100 MB
Array 1	`/home/trial`	3.62 TB

For a device to receive and send a directory of test files across a network, it required enough *free space* on the device to contain the whole directory. The contents of only a single directory were stored at any time. Table 5 shows the amount of space avialable on each device. It also shows the name of the directory on each device from where, and into which, the data was moved.

4 Data generating network combinations and their naming

The three parts of devices, networks, and files transferred were bought together in different combinations to generate the data required in this research. Only a few of all the possible combinations were used.

The four devices `Toshiba, macMini, MacPro`, and `Array 1`. Those devices were connected using the network media given in Table 6. Those networking media were GiGi-bit ethernet, WiFi on the 2.4 GHz and 5.0 GHz bands, and powerline networking using AV500 and AV1200 adapters manufactured by TP-Link. In the case of WiFi, the distance between the transmitter and receiver was changed.

Table 6: Networking media data indicies

Media	Distance	Index
ethernet		a
2.4 GHz	1.0 m	b
	11.0 m	c
	18.0 m	d
5 GHz	1.0 m	e
	11.0 m	f
	18.0 m	g
AV500		h
AV1200		i

The four devices were grouped in pairs as shown in Table 7. Data was transferred only between the devices in the pair. First one device in a pair would generate the file transfer, then the other device. The shell on the device was used to execute the file transfer command. With each device having to execute the shell command results in the repeating of each pair in Table 7. However, the shell of `Array 1` was not used in this manner.

Table 7: Device source and target combination data indicies

Shell device	Paired device	Index
Toshiba	macMini	A
macMini	Toshiba	B
macMini	macPro	C
macPro	macMini	D
Toshiba	macPro	E
macPro	Toshiba	F
Toshiba	Array 1	G
macMini	Array 1	I
macPro	Array 1	K

The *direction* of the file transfer with respect to each device pair was also recorded. A given device in a pair could copy files from itself to the other device of the pair, or copy files from that other device to itself. This is shown in Table 8.

Table 8: Data transfer direction data indicies

Direction	Index
download	X
upload	Y

The indices shown in Table 6, Table 7, and Table 8 were used with those of Table 4 to name the total combination used. A name consisted of the index from Table 7, Table 6, Table 8, and then Table 4. The filename extension `.raw` was attached to the renamed `transcript` file, while `.data` to the processed file ready for us with the R statistical package.

5 How the data was gathered

Data was collected by working from the directories listed in Figure 5 for each device in turn. Data was copied **into** the directory of the device using a command like:

```
script -c "scp -p 192.168.8.9:/data/trial" .
```

which copied from the device with TCP/IP address `192.168.8.9` (macPro) in this particular case. Data was copied **from** the directory using the command:

```
script -c "scp -p * 192.168.8.9:/data/trial" .
```

into directory `/home/trial` on the device with address `192.168.8.9`. Before giving these commands the contents of the directory to receive the data was cleared using the system's `rm *` command. The `-p` switch was used in each `scp` command to copy the date of each file unchanged. From version 7.2p2 of `openssh` was obtained `scp` used for all file copies on each device.

The files to be copied were first moved into the `/home/trial` directory of one device. Those files were then transferred to and from a second device, then another device, and so on. This was done to minimize movement of files the movement of which generated no data.

In all cases of using `script` a file named `transcript` was generated. It was stored in the directory from which the `script` command was executed. This was the required data file. After each `script` execution the `typescript` file was renamed according the nameing convention given in Section 4 and stored away.

A difficulty with this approach was the contents of the output file (`typescript`) produced by this `script/scp` process not only required renaming, but also required editing. Assumed the `typescript` file was renamed `AaX2.raw`. The first step was to use the `tr` utility, thus:

```
cat AaX2.raw | tr "\r" "\n" > a.data
```

to change all `^M` (return `\r`) characters in the `AaX2.raw` file into `\n` characters.

The output file `a.data` produced from that process was then passed through the `Perl` script:

```
#!/usr/bin/perl -w

print "                     filename             "; # header for R
print "percent  size    MBps   min sec";
printf "\n";
while ( <STDIN> )  {
```

```
  if ( /100%/ ) {
    @fields = split;
    @finish = split(/:/, $fields[4]);
    if ( $finish[0] =~ "--" ) {
    }
    else {
      chomp($_);
      printf "%40s  %5s  %6s  %6s",
        $fields[0],
        split(/%/,$fields[1]),
        split(/MB/, $fields[2]),
        split(/MB\/s/, $fields[3]), $fields[4];
      printf "    %2s  %2s\n", $finish[0], $finish[1];
    }
  }
}
```

which produced the required data for subsequent processing by R. For the `AaX2.raw` this data was stored as file `AaX2.data`. The first line of the data file contained:

filename percent size MBps min sec

to act a a header line for the R statistical package.

Most data required was the average speed to transfer each file. The above Perl script produced the required file. But the `transcript` file contained intermediate values which were required in a small number of cases. By removing the 100 from the `100%` value in the `if` statement of the Perl script, the data file containing those intermediate values resulted.

6 Analysis of results

The measurements obtained were examined statisticly using the R package. The mean of the measured transmission speed of each file in the data sets detailed in Table 4 was used as the indicator of the network speed for the indicated network connected hardware. Those speeds were in units of Mega bytes per second.

Both upload and download speed was measured for each pair of network device. In the analysis, the device combinations were indicated using the indices given in Table 7. To those pair indicators an X was appended to indicate download or a Y to indicate upload as in Table 8.

Speed data was collected for each file transferred between each pair of network connected devices. The mean of those speeds was tabulated as the transfer speed of that network combination and file/dataset. To test whether those speeds were different, the data streams from which each mean was calculated were compared. A paired Wilcox statistical test was used for this comparison. The Wilcox test freed the need to consider the speed measurements as being drawn from a Normal distribution. In the analysis tables the probability of the data transfer speeds being the same is tabulated. For example, in Table 9(c), the probability of AY and BX being the same is calculated as 0.0020. A figure of 0.05, or greater, was taken as indicating similarity, or being equivalent.

The analysis results are presented in matrix form. The means computed of the transmission speeds are in the left hand colomn. An under-triangular matrix only of computed probabilities of similarities of those means is given because that matrix is symmetric with the diagonal being 1.0.

In the following analysis each network connected device became the device to issue the `scp` command to download and upload to its networked pair. So, for example, download transferred data in both directions in the network. The exception is Table 12 where Array 1 was not used in that manner.

6.1 Ethernet LAN

Tables 9, 10, 11, and 12 give analysis of the measurements obtained using the devices of this study connected by thin wire ethernet.

Table 9: Comparison of Toshiba-MacMini omparisonupload and download speeds

(a) atkh dataset, 190 pairs

	mean	AY	BX	BY
AX	37.5	2.2e-16	2.2e-16	3.0e-06
AY	44.3		0.006	1.3e-15
BX	46.3			2.2e-16
BY	39.7			

(b) 18only dataset, 136 pairs

	mean	AY	BX	BY
AX	37.9	2.2e-16	2.2e-16	2.7e-9
AY	43.6		0.00005	8.4e-16
BX	45.1			2.2e-16
BY	39.9			

(c) brcc dataset, 80 pairs

	mean	AY	BX	BY
AX	37.2	6.1e-14	4.0e-14	6.7e-10
AY	43.6		0.0020	7.3e-11
BX	45.2			3.9e-13
BY	39.6			

(d) yanks dataset, 246 pairs

	mean	AY	BX	BY
AX	37.0	2.2e-16	0.0008	5.9e-6
AY	44.6		2.2e-16	9.2e-16
BX	38.9			0.195
BY	39.5			

Table 10: Comparison of MacMini-MacPro ethernet upload and download speeds

(a) atkh dataset, 190 pairs

	mean	CY	DX	DY
CX	103.0	2.2e-16	2.2e-12	0.244
CY	79.0		0.083	2.2e-16
DX	82.6			2.6e-16
DY	106.1			

(b) 18only dataset, 136 pairs

	mean	CY	DX	DY
CX	89.8	8.7e-6	0.034	0.643
CY	83.2		0.003	4.4e-5
DX	86.5			0.0015
DY	90.7			

(c) brcc dataset, 80 pairs

	mean	CY	DX	DY
CX	85.4	0.00037	0.690	1.9e-6
CY	78.2		1.2e-8	2.9e-12
DX	85.1			8.6e-8
DY	96.2			

(d) yanks dataset, 246 pairs

	mean	CY	DX	DY
CX	100.0	2.2e-16	2.2e-16	0.672
CY	78.8		0.218	2.2e-16
DX	76.9			2.2e-16
DY	100.4			

From the analysis contained in those tables the following can be given. If there be an *ethernet speed* then it is between 33 to 106 MB/s, with 49.6 MB/s being the average measured. This measured speed is for transferring files using `scp` secure copy using TCP/IP between devices using ethernet interfaces which are stated as being of Giga-bit per second capacity. The theoretical maximum is 117 MB/s.

Despite the *1000Mbit per second* rating of the ethernet interfaces on each of the four devices used, the macPro appeared to perform fastest.

A number of qualitative observations can be made from the analysis. These are:

Table 11: Comparison of MacPro-Toshiba ethernet upload and download speeds

(a) atkh dataset, 190 pairs

	mean	EY	FX	FY
EX	38.0	2.2e-16	2.2e-16	1.3e-8
EY	46.3		0.0033	2.2e-16
FX	48.0			2.2e-16
FY	40.7			

(b) 18only dataset, 136 pairs

	mean	EY	FX	FY
EX	38.4	2.2e-16	2.2e-16	2.7e-7
EY	45.7		3.8e-9	2.2e-16
FX	47.5			2.2e-16
FY	40.1			

(c) brcc dataset, 80 pairs

	mean	EY	FX	FY
EX	38.0	1.2e-14	8.0e-15	1.4e-6
EY	45.8		7.5e-11	3.7e-14
FX	47.4			1.2e-14
FY	39.8			

(d) yanks dataset, 246 pairs

	mean	EY	FX	FY
EX	37.9	2.2e-16	0.00011	6.4e-6
EY	45.6		2.2e-16	9.7e-16
FX	39.9			0.840
FY	40.2			

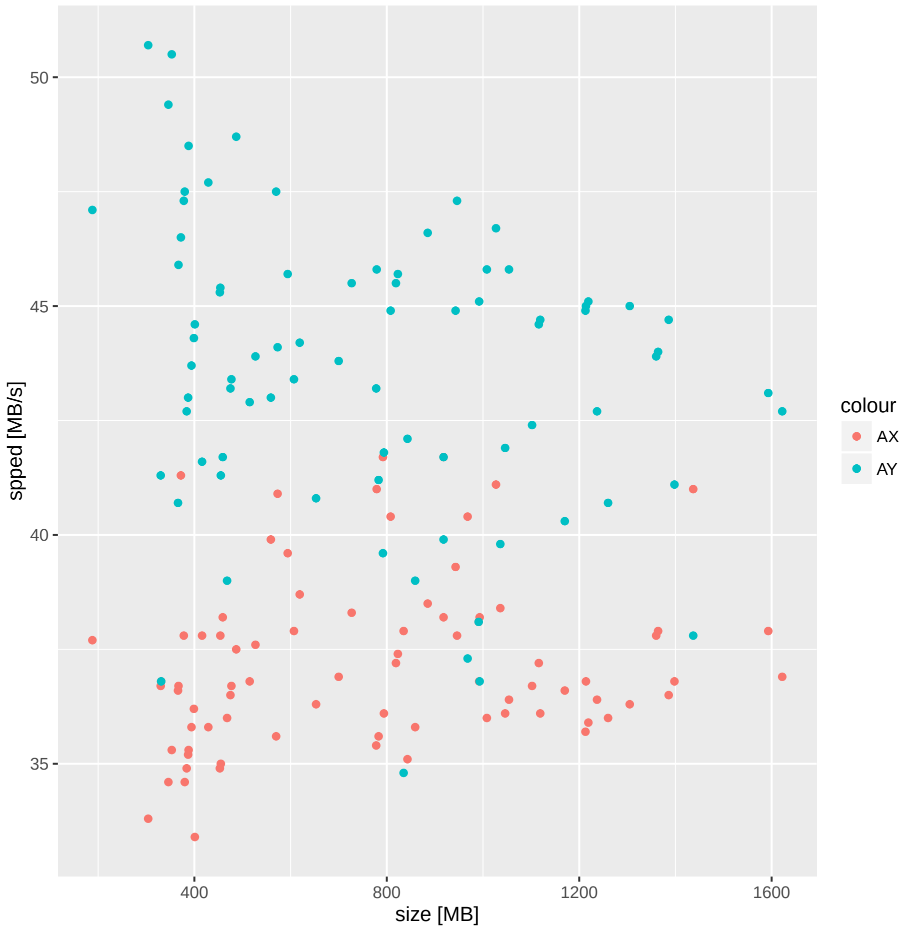


Figure 3: Transmission speeds verses file size for dataset 3 on ethernet connecting Toshiba laptop and MacMini with Toshiba master"

- Upload speed and download speeds are not equal;
- Each network device in a network connection influences the transmission speed across that network both when uploading and downloading files;

Table 12: Comparison of Toshiba, MacMini, MacPro-Array 1 ethernet upload and download speeds

(a) atkh dataset, 190 pairs

	mean	GY	IX	IY	KX	KY
GX	34.3	0.841	0.136	0.00066	2.55e-5	0.0248
GY	34.3		0.174	0.00031	3.87e-7	0.0196
IX	34.3			8.69e-6	0.0051	0.00060
IY	33.4				1.57e-12	0.191
KX	35.8					1.07e-10
KY	33.7					

(b) 18only dataset, 136 pairs

	mean	GY	IX	IY	KX	KY
GX	33.1	1.84e-11	0.699	9.81e-5	4.01e-12	0.518
GY	34.4		5.40e-10	0.00066	0.968	1.38e-12
IX	33.0			0.00016	4.11e-9	0.647
IY	33.8				0.00070	1.11e-5
KX	34.4					3.15e-11
KY	33.1					

(c) brcc dataset, 87 pairs

	mean	GY	IX	IY	KX	KY
GX	33.8	0.029	2.13e-6	0.320	0.061	1.24e-8
GY	34.2		2.76e-6	0.022	0.371	1.96e-11
IX	32.7			3.24e-5	4.08e-6	0.669
IY	33.8				0.037	1.43e-8
KX	34.1					2.96e-11
KY	32.6					

(d) yanks dataset, 246 pairs

	mean	GY	IX	IY	KX	KY
GX	33.2	0.016	0.093	0.423	0.012	0.830
GY	34.2		0.538	0.014	0.361	0.065
IX	33.9			0.086	0.351	0.200
IY	33.4				0.010	0.739
KX	34.3					0.038
KY	33.4					

- The scatter plot of Figure 3, which is typical for the measurements obtained, indicates transmission speed across a network is not a constant;
- Transmission speed is not dependent on the number of files transferred nor their size as indicated from the results obtained from the four data set used in which those parameters were different;
- Figure 4 shows typical network speed variation of successive files indicating no apparent structure in the speed variation, either during upload or download of files.
- Tables 9, 10 and 11 show the upload and download speeds of network pair where each member is the master i.e. who the upload or download is relative. This data does not suggest the master device controls the speed of the file transfer.
- Network behaviour did nat remain constant or consistent. For example, in Table 9 only the BX behaviour is similar to that of BY when data set 6 (`yank`) is used but in all other cases the network behaviours are non-similar.

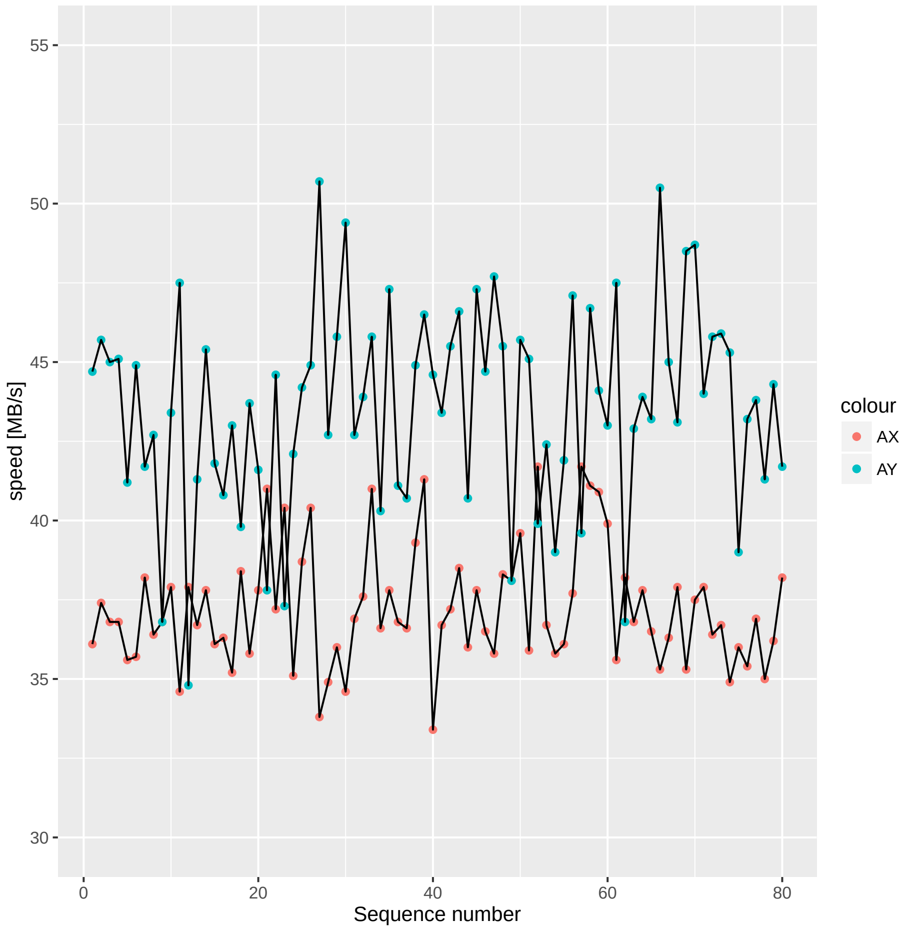


Figure 4: Sequential transmission speeds for dataset 3 files on ethernet connecting Toshiba laptop and MacMini with Toshiba master"

6.2 WiFi LAN

WiFi LANs currently occur in two types seprated by the frequency of their radio carrier; either 2.4MHz or 5.0MHz. In the same fashion as used for the ethernet network, Table 14 and Table 13 give an analysis of the measurements made using the WiFi network . In these measurements only data set 6 (`yanks`) was used which contained 246 files.

Table 13: Measured network file transfer speeds for configuration G

Network	Media	download	upload
ethernet	a	33.23	34.19
2.4GHz	b	3.49	3.40
	c	3.30	3.35
	d	2.28	2.50
5.0GHz	e	3.21	3.55
	f	3.53	3.50
	g	3.48	3.54

Table 13 shows the means of the transfer speeds measured. Of particular note is the difference between the speeds on the ethernet and the two WiFi networks.

Table 14: WiFi network compared with ethernet

(a) yanks dataset, 246 pairs

	mean	CaY	CbX	CbY	CeX	CeY
CaX	100.0	2.3e-16	2.2e-16	2.2e-16	2.2e-16	2.2e-16
CaY	78.8		2.2e-16	2.2e-16	2.2e-16	2.2e-16
CbX	3.2			0.013	2.2e-16	2.2e-16
CbY	3.2				2.2e-16	2.2e-16
CeX	3.4					2.2e-16
CeY	3.5					

(b) yanks dataset, 246 pairs

	mean	KaY	KbX	KbY	KeX	KeY
KaX	34.3	0.038	0.00011	2.2e-16	2.2e-16	2.2e-16
KaY	33.4		0.00109	2.2e-16	2.2e-16	2.2e-16
KbX	31.6			2.2e-16	2.2e-16	2.2e-16
KbY	3.3				2.2e-16	2.2e-16
KeX	3.5					NA
KeY	3.5					

A problem occurred in collecting CbY and KeX data. In Table 14, the computed means are given for the speed data which was collected, but the problem with KeX prevented comparison of the network behaviour with that of KeY.

The measurements obtained allowed the following observations to be made:

- Table 13 indicates little difference in speed between the 2.4GHz and 5.0GHz networks. In Table 14 the C results for the 2.4GHz network and the K results for the 5.0GHz network show this same pattern. So the frequence of the network appears not to affect the transmission speed.
- Figure 5 shows a particular case of what is typical distribution of speed measured on the WiFi network at three distances between the source and destination. The speeds are similar. The speeds tabulated in Table 13 show a similar pattern. So distance appears not to affect the transmission speed.
- The (a) and (b) tabulations in Table 14 indicate for WiFi the speeds are similar despite different hardware being used, the hardware being the C and K combinations tabulated in Table 6. This data suggests the hardware used does not affect transmission speed.
- The similarity in the speeds for the download and uploads in Table 13 and Table 14 suggest download and upload speed is the same for both 2.4GHz and 5.0GHz networks. Download and upload speeds appear to be similar.
- Figure 6 shows the speeds of successive files measured on a 2.4GHz network which was typical of the observed behaviour. Each upload and download appeared to form *preferred* speeds accompanied by the occasional high or low speed. The speed of transmission is not affected by the file size.
- The mean speeds tabulated in Table 14 for the Ca and Ka cases conpared to the other C and K cases respectively indicate WiFi transmission is much slower than with hardwired ethernet.

In this work, the 2.4GHz and 5.0GHz frequence networks were used separately. Either the 2.4HGhz network or the 5.0GHz network was used. In AC WiFi currently offered on devices, including the MacPro and the Viper devices used here, these frequences are used simultaneously. Any

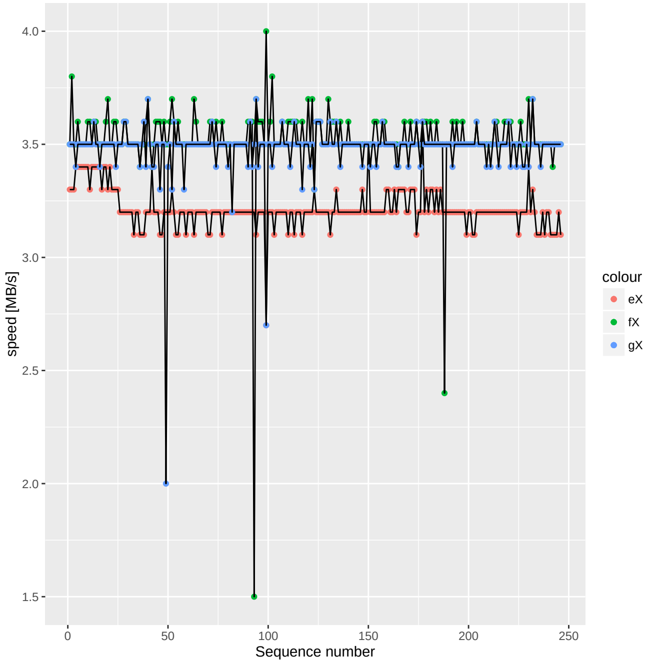


Figure 5: Download transfer speed for each file in dataset 6 between Toshiba laptop and Array 1 at 1m, 11m, and 18m on 5.0GHz WiFi

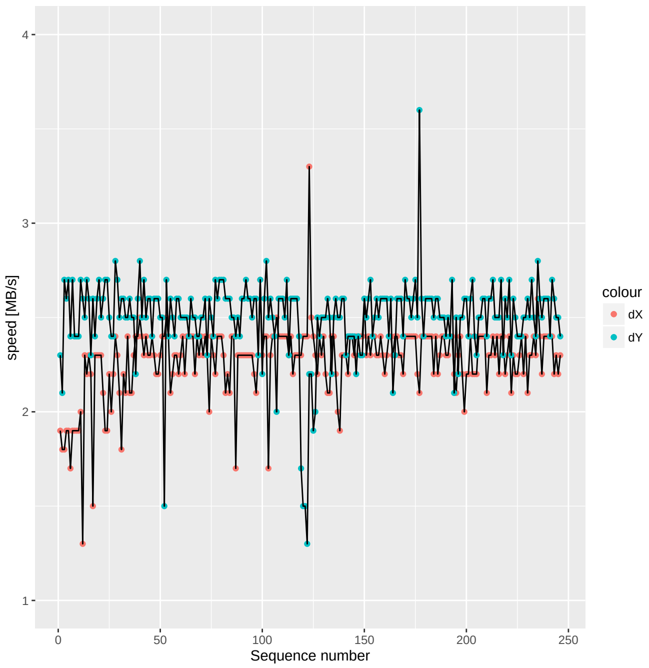


Figure 6: Download transfer speed for each file in dataset 6 between Toshiba laptop and Array 1 at 18m on 2.4GHz WiFi

particular data stream can be either transmitted over one or the other frequence network, but not split between both.

A major problem with WiFi is the drop in speed in going onto a wireless carrier. One metre distance between the WiFi transmitter and receiver as indicated in the `b` speeds of Tables 13 and 14 was measured a reducing the transmission speed by a factor of 10 with respect to the hardwired ethernet speeds (the `a` speeds).

6.3 Powerline LAN

The LAN of Figure 1 was altered so Array 1 was connected to the switch via a powerline adapter pair. One adapter was connected to the Array 1 via a RJ45 ethernet cable and another adapter connected via a RJ45 ethernet cable to the switch. The adapters were configured as a pair. Each adapter was plugged into and powered by the household wiring in a single circuit protected by one wire fuse. The recommendation is to have adapters on the same power circuit. The adaptors used were separated by approximately 10 metres of household power circuit wiring.

Two pairs of adapter models were used, one pair at a time. One pair was a TP-LINK 500Mbps adapter and the other a TP-LINK 1200Mbps adapter. Unlike most network devices, these adapters (as with all powerline adapters) do not have network or MAC addresses. The only configuration available was by pressing the *Pair* button on the device to connect that device into a networking relationship with similar powerline adapters.

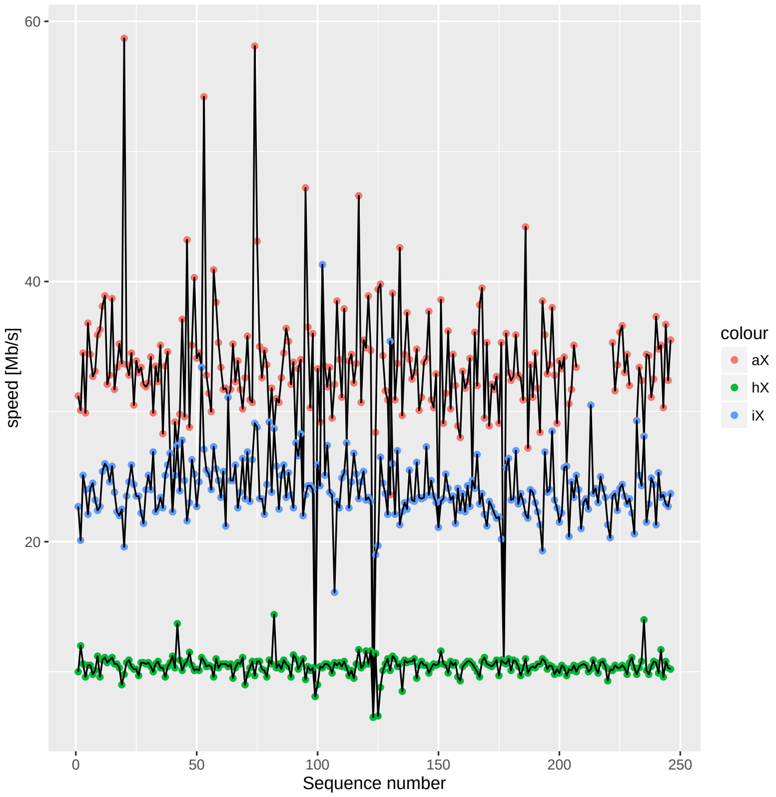
From Section 2 the theoretical data transfer rate for a 1Gbps network should be 117Mb/s. So if the 500Mbps rating is correct for the TP-LINK 500Mbps adapter, if should pass 58MB/s. The TP-LINK 1200Mbps adapter should pass 140MB/s. Do they?

The files of dataset 6, detailed in Table 4, were downloaded from Array 1 to the Toshiba laptop using the network configuration outlined. Then those files were uploaded. This process was repeated using the TP-LINK 500Mbps and TP-LINK 1200Mbps powerline adapters. The speeds measured are shown in Figure 7. For comparison, the speeds measured connecting those network devices using thin wire ethernet is also shown. Table 15 shows the mean speeds of each of the network transfers types.

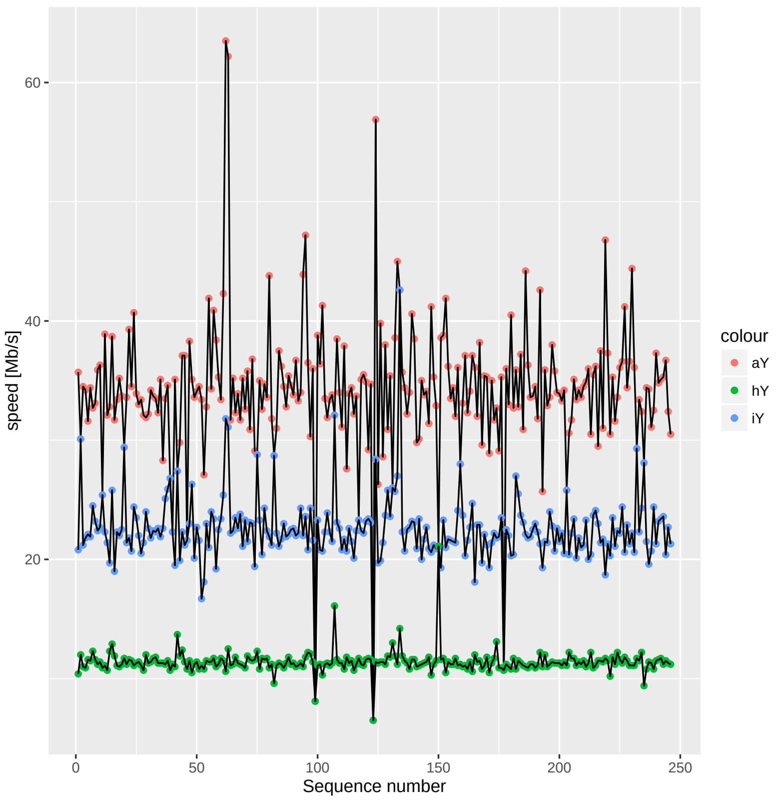
Table 15: Mean transfer speeds measured for powerline adapters and ethernet using dataset 6 between Toshiba laptop and array 1

Network element	downloading	uploading
thin wire ethernet	33.3	34.2
TP-LINK 500Mbps	10.4	11.4
TP-LINK 1200Mbps	23.9	22.4

From these measurements, the TP-LINK 1200Mbps adapter has approximately twice the through put of the TP-LINK 500Mbps adapter. However, in comparison to the thin wire ethernet which is rated at 1Gbps, both adapters fall short of the data through put which would be expected from the bit rating claimed for each.



(a) Downloads



(b) Uploads

Figure 7: File transfer speed between Toshiba laptop and Array 1 of dataset 6 connected by ethernet, 500Mbps and 1200Mbps powerlines

6.4 NBN behaviour

The LAN of Figure 1 was connected to the NBN by using a pair of powerline adapters. The speeds shown in Table 15 suggested the TP-LINK 500Mbps adapter to be the appropriate choice, being the lesser in cost and closest to the expected speed of the NBN. One of the adapptors was connected by a RJ45 ethernet cable to the Netgear switch. The other adapter was connected to one of the LAN ports of a Viper router. An RJ45 ethernet cable connected the WAN port of the Viper to LAN 1 port of the NBN connection box. The household power circuit containing the pair of adapters was protected by a single wire fuse. Approximately 20 metre of household power wiring connected the two powerline adapters.

The NBN service was for separate telephone and Internet, i.e. the telephone service was not done through VoIP on the Internet modem. A nominal 100/40 Mbps service was provided by the Internet Service Provider. A FTTP (Fibre To The Premises) connection to the NBN was made.

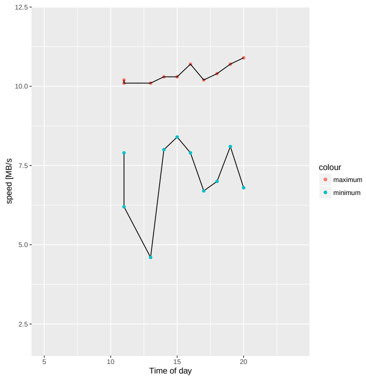
Measurement of the Internet service was done by running `Firefox` on the Toshiba laptop. Firefox provided both a continuous indication of the accumulated download amount, plus, by moving the mouse pointer into that download indicator sub-window, the instanteous download speed in units of MB/s. The file `nbn-test.dat` from URL `ftp://ftp.iinet.net.au` was downloaded. This server was located in Subiaco while the download equipment was located in Booragoon, both suburbs of Perth, Western Australia, and separated by 14km road distance. Although the file downloaded was larger than 14GB, only part of it was used to obtain the measurement.

After clicking the mouse button over the required file to start the download, and indicating to save the download, the speed of the first 200MB was not considered. In this amount, the download was assumed to stabilise. In the download of the next 1.4GB, the mouse pointer was used to display the download speed approximately every 5 seconds. The value shown was used to update the maximum and minimum speed measurement. After this amount of download, it was terminated and the maximum and minimum speeds recorded with the time of day of the measurement.

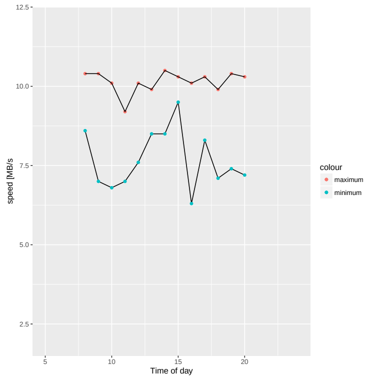
This process was repeated at hourly intervals across a day. Measurements on seven consecutive days from Sunday, 30 October 2016 to Sunday, 6 November were obtained from approximately 0800hrs to 2000hrs. Figure 8 shows the measurements.

The measurements show the NBN speed varies over both days and time of day. The greatest speed observed was 10.9MB/s and the lowest 2.0MB/s. The NBN operates as an ethernet network carrying TCP/IP packets. In Section 2 such a network with a 1000MBps rating was calculated as being able to provide a data transport speed of 117MB/s. So a 100MBps network rating of the NBN should have a theoretical maximum data transport of 11.7MB/s. The maximum speeds measured were close to this value.

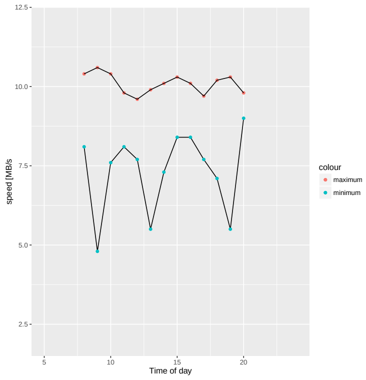
A possible reason for the variation of the maximum sppeds and the minimum speed different from those maximum is the shared use of the NBN cable. This occupance was neither observed or could be control when obtaining these measurements.



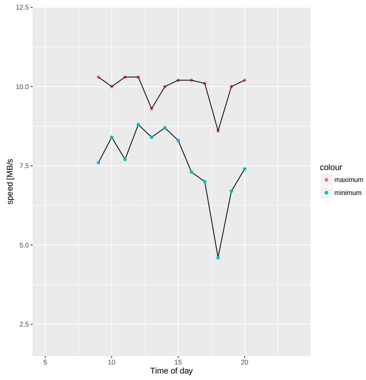
(a) Sunday



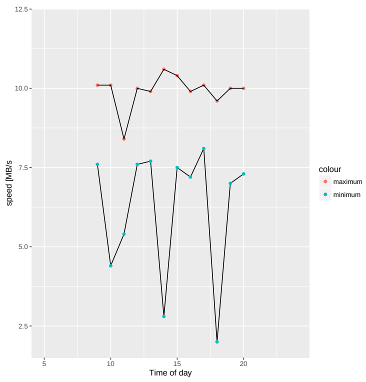
(b) Monday



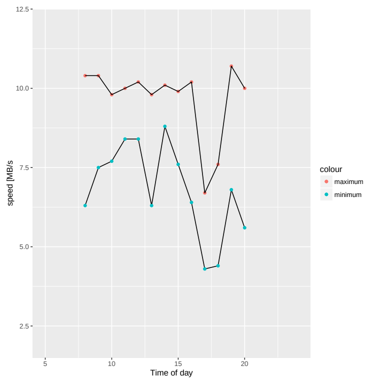
(c) Tuesday



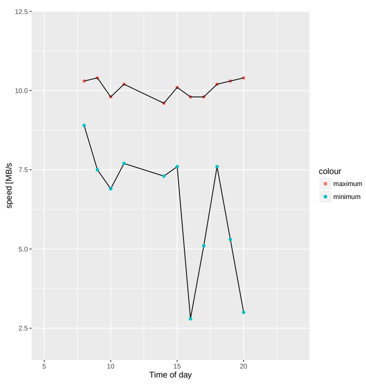
(d) Wednesday



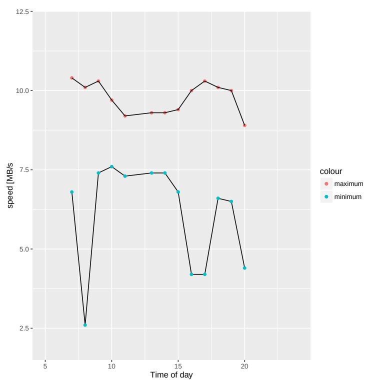
(e) Thursday



(f) Friday



(g) Saturday



(h) Sunday

Figure 8: Snapshot of file download speeds across days of a one week peiod on the NBN