

BACHELOR THESIS
COMPUTING SCIENCE



RADBOD UNIVERSITY

**Energy efficient WLAN using WiFi
standards b/a/g/n/ac on an
Archer C7 AC1750 access point**

Author:

Gunnar P. Noordbruis
s1008953

First supervisor/assessor:

dr. B.E. van Gastel
B.vanGastel@cs.ru.nl

Second assessor:

prof. dr. M.C.J.D. van Eekelen
marko@cs.ru.nl

June 28, 2020

Abstract

This thesis investigates the possibilities of increasing the energy efficiency of an access point using the WiFi standards 802.11 b/a/g/n/ac with their respective bandwidths.

An experiment was conducted to establish the power consumption of an Archer C7 AC1750 access point. Next to this, three simulations were performed extrapolating the data gathered from the experiment. This was done to get insight in to the energy consumption of the access point over the period of a month.

The experiment performed shows that the b, a and g standards draw less power compared to the their 2.4GHz and 5GHz counter parts but due to a longer transmission time the total consumed amount of energy is significantly larger. Moreover, all 2.4GHz networks have a similar idle power draw and within the 5GHz category two groups are seen a and n with a 20MHz, and the remaining standards.

The first two simulations showed that 802.11n with a 20MHz bandwidth consumes the least amount of energy within a month. However, 802.11ac with a 80MHz bandwidth becomes the least consuming standard when the network is turned off when it is not in use. The final simulation shows roughly an 23% decrease in energy consumption of the access point when switching between two standards.

Based on the results of the simulation, the most energy efficient standard is 2.4GHz 802.11n at a 20MHz bandwidth. Next to this, standard switching has shown to significantly reduce the energy consumption. The same holds for a wake/sleep solution.

Contents

1	Introduction	3
2	Background IEEE 802.11	5
2.1	Network types	5
2.2	Frame types and their use	6
2.3	Network authentication and association	7
2.4	Power conservation	8
2.5	WiFi standards b/a/g/n/ac	9
3	Measuring energy consumption of WiFi standards	10
3.1	Methodology energy consumption measurements	10
3.2	Results	12
4	Extrapolating measurement results	16
4.1	A single energy efficient standard	16
4.2	Standard Switching	17
4.2.1	Drawbacks switching	18
4.2.2	Outcome	18
5	Related Work	20
6	Conclusions	22
6.1	Future work	24

A	Appendix	28
A.1	Detailed AP setup	28
B	Energy and power the difference	31
C	Measurement code explained	32
D	Raw measurement results	36
D.1	Experiment results 2.4Ghz	36
D.2	Experiment results 5Ghz	39
E	Calculation of access point energy consumption per month	42

Chapter 1

Introduction

Within the Netherlands alone 98% of the households have an internet connection. In those households, smartphones and laptops are the most common devices to surf the web with¹. Next to this, WiFi networks are generally used for longer sessions transferring larger amounts of data than their cellular counter part [1]. WiFi capable devices like smartphones and laptops are also known as stations (STAs) and use WiFi (IEEE 802.11) to connect with an access point (AP). The AP is a gateway that allows the devices to be wireless but still maintain access to the internet, transmitting and receiving the packets to and from stations using a specific version of the standard with a set bandwidth. The bandwidth is a specific range of frequencies that the AP uses to transfer data on and in this case the bandwidth can either be 20, 40 or 80MHz. A larger bandwidth allows for more data to be transferred but also has a higher risk of interference with other WiFi devices. More and more of these networks are being installed. In 2007 communication networks, personal computers, and data centers used 3.9% of the total energy consumption worldwide, in 2012 this grew to 4.6% [10]. From the total amount of electricity consumed by ICT a significant 29% is consumed solely by networks [4].

Within 802.11 only stations have the option to enter an energy efficient state. In this state the STA tells the AP that it will go into Power Saving Mode, the AP buffers the packets that arrive for the STA and when the AP broadcasts its beacon. The beacon includes which STAs have buffered packets waiting, such that those STAs can then retrieve their packets.

Different studies have been performed to make WiFi networks more energy efficient. Examples of these studies can be found in chapter 5. Next to this,

¹<https://longreads.cbs.nl/ict-kennis-en-economie-2019/ict-gebruik-van-huishoudens-en-personen/>

one could argue that ZigBee (802.15.4) or Bluetooth Low Energy (802.15.1) can be used instead because they are more energy efficient compared to WiFi. However, ZigBee has a maximum transfer rate of 250 Kbps[7] and Bluetooth without using an 802.11 link can only achieve 2Mbps [3]. Transferring the average European monthly data consumption of 175.7GB [18] would take ZigBee and Bluetooth, 68 and 8 days respectively. Only a 4G or higher cellular internet connection can compete with WiFi providing between 100 ~ 1000Mbps [21] versus the 802.11ac standard achieving between 433 Mbps and several Gbps². WiFi is vastly superior in data transfer rates compared to ZigBee and Bluetooth and similar to cellular, however greenifying WiFi access points is falling behind. Access points are not optimized to conserve energy. This leads to the main research questions:

What are the possibilities of increasing the energy efficiency of an Archer C7 AC1750 access point using the WiFi standards b/a/g/n/ac?

In other words, the main goal is to investigate if the energy consumption of an Archer C7 AC1750 access point can be reduced through the use of the WiFi standards b/a/g/n/ac.

The energy consumption of only one AP will be measured using three files with three runs each. This, is done within a constant interferenceless environment. The energy consumption results of the small scale measurements will be extrapolated to calculate the energy consumption over a longer period of a month within the average home. We will not produce our own WiFi standard nor perform monthlong measurements.

This thesis provides the measurement results, extrapolations of those results to a period of a month, the measurement setup and the code to perform the measurements. Next to this, recommendations are done regarding the most energy efficient WiFi standard, and regarding standard switching compared to a wake/sleep cycle.

To achieve the goal the energy consumption of the access point needs to be known. The measured energy consumption for transmitting and the power draw when idling are discussed in chapter 3. Section 4.1 provides and analyses the AP's energy consumption of a month, extrapolated from the results of the experiment. Next to this, in section 4.2 switching between the standards over time to reduce energy consumption over the period of a month is simulated. Chapter 5 covers related approaches to the problem at hand. Finally, in chapter 6 conclusions are drawn from the results and future work is discussed.

²<https://www.actiontec.com/wifihelp/evolution-wi-fi-standards-look-802-11abgnac/>

Chapter 2

Background IEEE 802.11

In this chapter the important facets of the IEEE 802.11 protocol related to this paper will be explained. The following information is gathered from the IEEE 802.11 specifications, M. Gast's book "802.11 Wireless Networks The Definitive Guide" and "Computer Networking A Top-Down Approach" by James F. Kurose and Keith W. Ross [11, 8, 14]

IEEE 802.11 is a set of medium access control (MAC) and physical layer (PHY) specification for wireless devices.

2.1 Network types

Two different network types exist, an independent network and an infrastructure network. On the one hand an independent network or ad hoc network is the combination of a set of stations. An example can be seen in figure 2.1, all mobile stations can communicate with each other. Each packet thus requires one hop. This type of network has a short lifespan, meaning that it is created and then utilized for a small period of time after which the network is dissolved. On the other hand an infrastructure network is created out of:

1. Mobile stations, these are laptops, phones or anything with an wireless network interface controller.
2. An access point, through which all traffic flows.

Other than in an independent network, an infrastructure network's packets flow through the access point. The communication thus takes two hops. An example can be viewed in figure 2.2. A benefit of an infrastructure network

is that the mobile stations do not have to be in each others basic service area. A collection of these stations connected to an AP is called a basic service set (BSS). Within this paper we will only be using an infrastructure network as we will be looking more closely at the AP and its energy consumption.

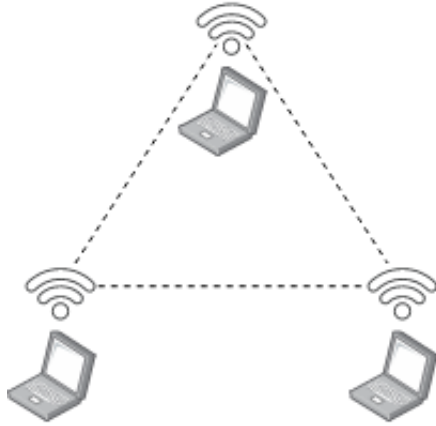


Figure 2.1: Independent network example

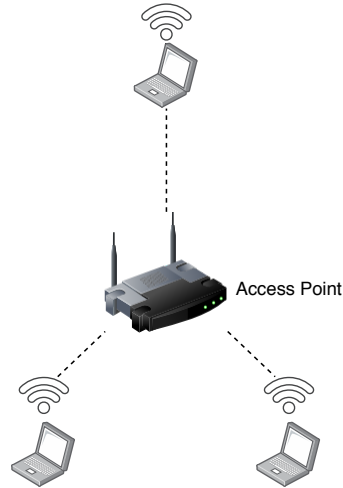


Figure 2.2: Independent network example

2.2 Frame types and their use

Within IEEE 802.11 multiple types of frames exist, the three most used frames are Data frames, control frames and management frames. These frames have different structure and uses.

1. Data frames - used to transmit data between stations and AP.
2. Management frames - used to provide authentication, connect and disconnect functionality to wireless networks.
3. Control frames - these frames help with the transmission of data. All control frames have the same 2 byte Frame Control field which is depicted in figure 2.3 A few different control frames exist:
 - (a) RTS - Request To Send, a broadcasted request from a station signaling that it wants to transmit large data frames to the AP. A threshold is set for which data frame size a RTS should be sent. When granted with a CTS other stations will be silent. Creating a moment where collision is very unlikely.

Bits	Protocol		Type=data		Sub type				To DS	From DS	More Frag	Retry	Pwr Mgmt	More Data	Prtcd Frame	Order
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Figure 2.3: Frame control fields

- (b) CTS - Clear To Send, usually a response to a RTS. Giving that station the right of way to transmit large data frames.
- (c) ACK - Acknowledgement of received data.
- (d) PS-POLL - Power Save Poll, this frame is used by stations that have been in power saving mode. The station transmits this frame to the AP to retrieve possibly buffered frames while the station was in power saving mode.

2.3 Network authentication and association

Authentication is the first step that is performed when connecting to a network, either known or unknown. Two types of authentication exist:

1. Open system
2. Shared key

The open system is used in a WiFi network that is not (password) protected. The second type, shared key, is a WiFi network that is protected using WEP, WPA or WPA2 usually with a pre-shared keys (PSK). The open system will be discussed as it is sufficient for this paper. The station starts at state one depicted in figure 2.4. To connect a station to a WiFi network, the station must first authenticate itself to the AP it wants to connect to. Because this is an open WiFi network the station does not need a PSK to authenticate. Thus the station sends an authentication request to the AP containing the stations ID which is usually the MAC address, the AP responds with an authentication response with a value of success (0) or failure (1). When the AP returns success, the station knows that it is now in state two of figure 2.4, the station is authenticated and can proceed to association.

Association is usually performed directly after authentication, the AP and station know each others identity and the station sends a simple association request. The way an AP decides when to allow a station to associate with the network can differ between APs or manufacturers. When an AP

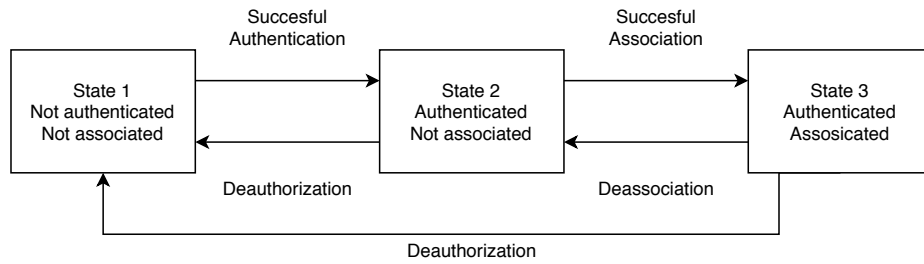


Figure 2.4: IEEE 802.11 authentication and association states

decides the station will be allowed to associate, it sends an association response with the status code of success and the Association ID (AID) for the station. This AID is used to identify the station when frames have been buffered while the stations was in Power Saving Mode. When the station received this response it knows it is now in state 3. Meaning it can start to transfer data frames with the AP.

2.4 Power conservation

Power management is currently implemented in such a way that it heavily utilizes the access point. When a station decides it wants to sleep or enter Power Saving mode (PS) it indicates this to the AP. The AP performs two tasks to provide the stations with PS capability, the first is buffering the frames that arrive for a station that is currently sleeping. Secondly, within the beacon the AP broadcasts periodically, the AP includes the stations AID for which it has buffered frames. An important limitation is that the AP has an finite amount of memory and thus the AP and station agree to a 'Listen Interval'. The AP waits at least the Listen Interval before it discards any frames. The current "Power management [system] is designed around the needs of the battery-powered mobile stations" [8].

Transmit Power Control (TPC) limits the transmission power such that it stays within the regulatory required ranges [8]. Transmitting a signal costs energy, if a signal can reaches further than what it needs to, the energy used to transmit it further is wasted. TPC was not intended to be used to conserve power but the papers [12] and [22] both use this feature to reduce the energy consumption.

2.5 WiFi standards b/a/g/n/ac

The first WiFi version became available to consumers in 1997 it was called IEEE 802.11. Over time the standard received updates, usually becoming faster as the original 802.11 supported only speeds of up to 2Mbps. These updates are called amendments and the first two amendments were both released in 1999, 802.11b 802.11a. 802.11b uses a 2.4GHz frequency and has a theoretical limit of 11Mbps, 802.11b does not bring new features to the WiFi family, it is merely an update of the first standard. 802.11a on the other hand brought orthogonal frequency division multiplexing (OFDM) to the table. Next to this, 802.11a operates on the 5GHz frequency band allowing a max theoretical max throughput of 54 Mbps.

In 2003 a faster 2.4GHz standard was released, 802.11g. wanted "to combine the best of both IEEE 802.11a and IEEE 802.11b"¹. That is why, 802.11g is backwards compatible with IEEE 802.11b and also implements OFDM achieving a theoretical max throughput of 54Mbps. The 2.4GHz devices are cheaper to manufacture compared to 5GHz². Only 6 years later IEEE 802.11n was released bringing Multiple Input Multiple Output (MIMO) with it. Simply put MIMO, uses several antennas to send and receive more data at a time, achieving a theoretical max speed of 600Mbps using MIMO and 300Mbps without³. The IEEE 802.11n works on both 2.4GHz and 5GHz frequencies. IEEE 802.11ac is the first standard to reach speeds north of 1 GB/s. This, mostly done through:

1. A maximum bandwidth increase to 160MHz and a mandatory capability of 80MHz (however, 802.11ac can also be forced to use 20 and 40 MHz)
2. Implementing beamforming, instead of sending a signal in a circle around the device, it is send more precisely towards a device
3. Implementing Multi User MIMO (MU-MIMO). Meaning that multiple devices that are in a different spatial area can be served at the same time.

¹<https://www.lifewire.com/wireless-standards-802-11a-802-11b-g-n-and-802-11ac-816553>

²<https://alethea.in/iot-age-wifi-6/>

³<https://www.intel.com/content/www/us/en/support/articles/000005714/network-and-i-o/wireless-networking.html>

Chapter 3

Measuring energy consumption of WiFi standards

In this chapter, the measurement methodology that is used to measure the power consumption of the WiFi standards is described. Next to that results of these measurements will be discussed.

The goal is to reduce the energy consumption of a wireless access point using the WiFi standards. To determine which approaches can reduce the energy consumption of the AP, the energy consumption of the different WiFi standards must be known. This, will be done using an experiment measuring the power draw when retrieving a hosted file and the power draw when idling with the network on and off. To be precise, idle means that there is no data being transferred between the AP and any of its STAs.

3.1 Methodology energy consumption measurements

The file that will be used to measure the transmitting energy consumption will be hosted locally on the network to guarantee identical circumstances among each run of the experiment. The server that hosts this file is connected to the AP with an ethernet cable. Both of these attributes create an environment, in which any delay or congestion can only be caused by the WiFi network. The tests have been performed in a location without cellular signal and interference from other WiFi networks/devices. Because of this, the consequences of the variable is measured, this gives high validity measurement results.

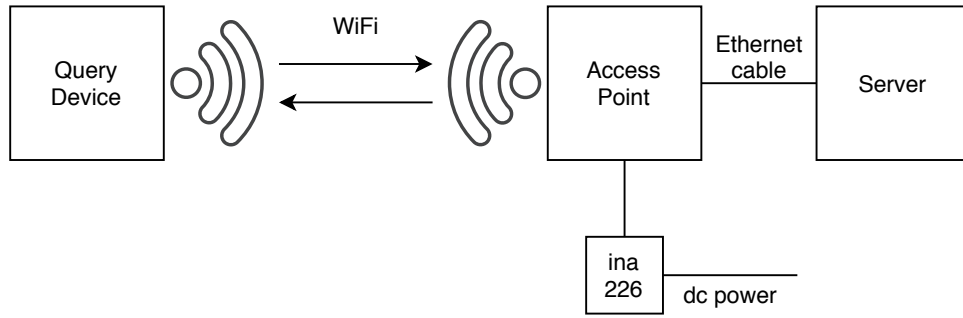


Figure 3.1: Setup of test network

The file that will be retrieved with a run is one of three different sizes, 1MB, 10MB and 100MB. We decided to diversify the sample files to gain data that represents normal use more compared to only one size. The actual size of the sample files is arbitrary. Other file sizes like 3 MiB would work just as good, but would have been less easy to work with. The sizes could be associated with for example a websites, images or video streaming. Next to that, we want to limit the possibility that one standard has unexpected benefits with a specific file size.

Measuring the power consumption is done using a tinkerforge voltage/current bricklet v2¹, which is made up out of an ina 226. This, bricklet measures DC energy consumption, meaning that it is placed between the AP and its power plug. Next to this, the bricklet has a resolution of 1mW, 1mV and 1mA¹. The measured values thus has an error of $\pm 1\text{mW}$. The power cable has been cut and spliced open to connect the bricklet inline. This is beneficial as the power loss due to AC to DC conversion of the power plug is not measured. An overview of the setup can be viewed in figure 3.1

For the idle power consumption test one minute of power consumption was measured 3 times for each standard. The same three one minute power consumption measurement was done with the WiFi networks off. This has been done using the same setup as the file transfer measurements.

The setup in figure 3.1 consist of a few basic components:

1. Query Device - This is WiFi capable laptop of up to IEEE 802.11ac
2. WiFi - This is a WiFi network that is not connected to the world wide web. The wireless network is limited to one WiFi standard at a time.

¹https://www.tinkerforge.com/en/doc/Hardware/Bricklets/Voltage_Current_V2.html

3. Access Point - This is the device that creates the wireless network. In the case of this paper a TP-LINK Archer C7 AC1750 EU v2 running the latest OpenWRT firmware (19.07.2)
4. Server - this is a computer that is connected to the Access Point using an gigabit cat 6 ethernet cable .
5. Ina 226 voltage current bricklet - this device measures the power consumption of the Access Point. It is controlled by a Master Brick v2.1

During the experiment some issues occurred although the measurements went reasonably smooth. The following issues have all been solved and did not impact the results.

- In one instance the retrieval of the file over the WiFi standard 'b' crashed. The cause of the problem is unknown but was not encountered again in later retrievals.
- On several occasions the AP did not ad hear to the set parameters. This issues was solved by either rebooting or factory resetting and reapplying the settings if rebooting did not work.
- When sequentially doing the same test the Tinkerforge voltage/current v2 bricklet would not start or end measurement accurately. The cause of this is unknown. Next to this, it could easily be avoided by altering the program only perform one test instead of all. This gave more consistent results.
- During the measurement of the idle power draw an empty while loop would stop looping before achieving a false boolean value. The value in this case was a comparison between the number of measurements taken and the number of measurements that need to be taken. This problem caused some objects to not be destroyed. Meaning that the measurement would never stop. This was solved by printing the progress of the measurements in the while loop.

3.2 Results

In this section, the results from the energy consumption experiments outlined in section 3.1 are presented and examined in detail. The effects of the different WiFi standards on the energy consumption of the AP are explained, and various implications for design are discussed. The raw data can be found in appendix D.

We tested every WiFi standard with 3 different files. The WiFi standards are categorized into two groups. Namely, 2.4GHz(blue) and 5GHz(orange) and arranged from old(left) to new(right). The average energy consumption of the 100MB transfer can be found in figure 3.2. Next to this, we have the average time the transfer took in figure 3.3 and the average power draw in figure 3.4. Only the 100MB test results are shown below, because the energy consumption, power draw, and time results of the 1 and 10MB test have a very similar pattern. Displaying those results here too, would not contribute new information. Next to this, the 100MB test is more accurate due each sample contributing less to the total average.

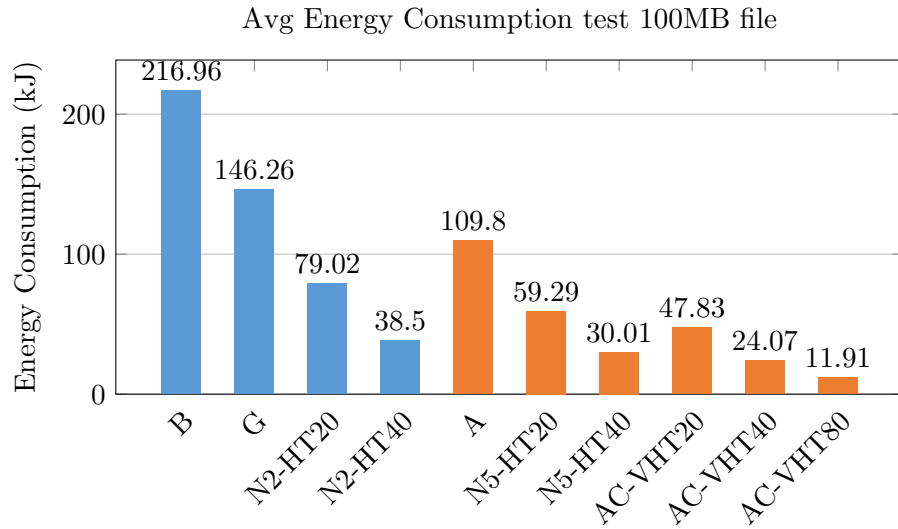


Figure 3.2: Average energy consumption of each WiFi standard during 100MB test in kilo Joule. Blue represents the 2.4GHz WiFi networks and orange the 5GHz WiFi networks.

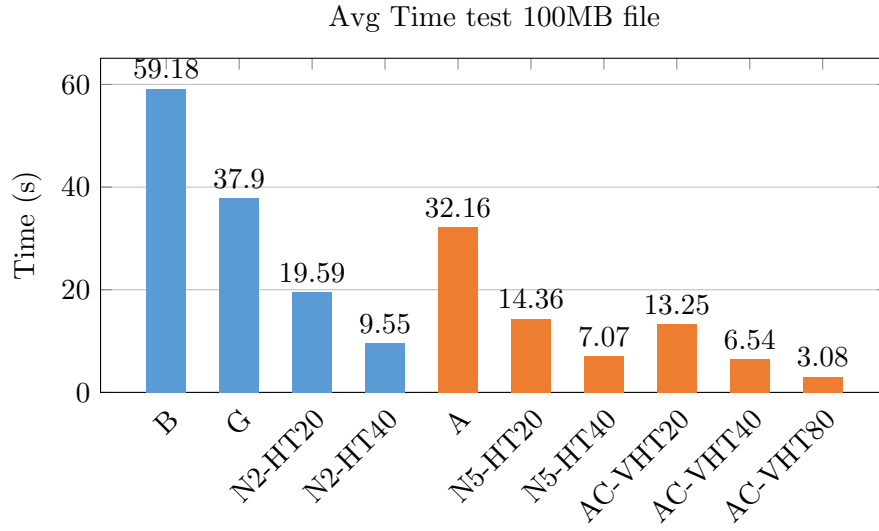


Figure 3.3: Average time consumption of each WiFi standard during 100MB test in seconds. Blue represents the 2.4GHz WiFi networks and orange the 5GHz WiFi networks.

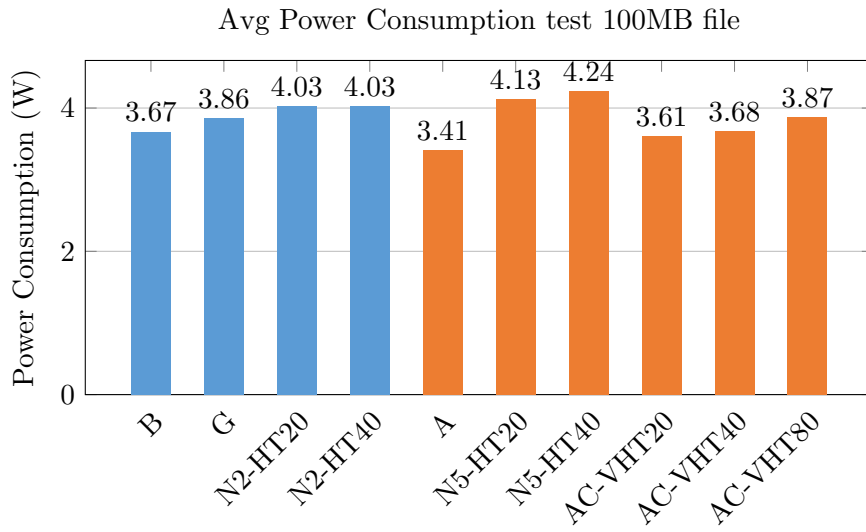


Figure 3.4: Average power draw of each WiFi standard during 100MB test in Watts. Blue represents the 2.4GHz WiFi networks and orange the 5GHz WiFi networks.

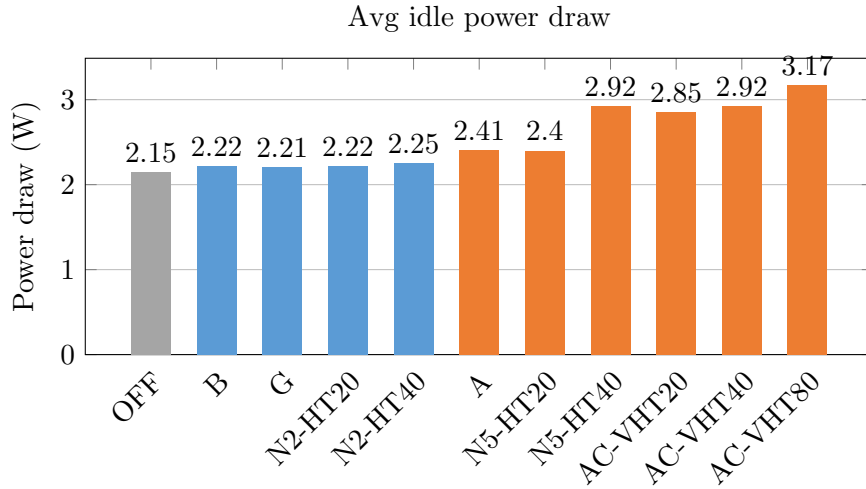


Figure 3.5: Average power draw of each WiFi standard while idling in Watts. Gray, blue and orange represent WiFi networks off, 2.4GHz WiFi networks, and 5GHz WiFi networks respectively.

When looking at figure 3.2 an decreasing pattern from old to new is visible in both the 2,4GHz and 5GHz category. Figure 3.4 shows that in the 2.4GHz category the older b and g WiFi standards have a lower power draw compared to the 802.11n standard of 0.17 to 0.36W. However, these older standards also have a lower transfer speed as explain in section 2.5 causing the transmission to take longer and thus staying longer within the higher power draw state. 802.11b and g consume 3.69 and 2.49 times as much energy as the average of 2.4GHz 802.11n standard. The same is visible on the 5GHz frequency where 802.11a draws between 0.46 and 0.83W less compared to the other 5GHz standards but due to the longer transmission it consumes 3.17 times as much energy as the average of all 5GHz standards. The idle power draw of the 2.4GHz standards are consistent and only have a small difference of 0.04W at most, this is 0.01W for the 5GHz standards 802.11a and 802.11n with a 20MHz bandwidth. The remaining 5GHz standards are closer to 3W with the largest difference being 0.32W.

The results, figure 3.5, of the idle test show that the 2.4GHz networks take $\pm 0.08W$ more compared to off. The difference is larger in the 5GHz networks, due to the 5GHz frequencies requiring more power to produce. The results of (b,g and n2-HT20), (a and n5-HT20) and (n5-HT40 and ac-VHT40) are very close this is likely due the use of the same bandwidth. However, this does not explain the difference between: (a, n5-HT20 and ac-VHT20). The measurements have been redone to eliminate possible background operations from impacting the measurement, this still gave the same results.

Chapter 4

Extrapolating measurement results

In this chapter the energy consumption of an access point using only one standard and using standard switching for the time period of a month are calculated and analysed.

This will be done using the results from the experiment in section 3. The energy consumption of an access point can be calculated using the amount of data that needs to be transferred. From that amount and the WiFi standard, the transfer and idle time can be deduced. Using this information the total amount of energy consumed during a month can be calculated. The weighted amount of data transferred in Europe is 175.7GB per month in the first quarter of 2019 [18].

The measurements in chapter 3 have been performed within an interferenceless environment. Next to this, the transferred data was hosted locally, meaning that the full or close to full speeds of the standards can be used. Both of these attributes are desirable for validity of the results. However, they decrease the applicability of the extrapolations to the real world.

4.1 A single energy efficient standard

In table 4.1 the AP's energy consumption of a month has been calculated with the AP idling with WiFi on and with WiFi off and transmitting the 175.7GB in chunks of 100MB. The consumption when using the 1 and 10MB transmit sizes have also been calculated and provide identical results. These tables can be found in appendix E. Very visible is that the newer 5GHz standards end up consuming more energy due to higher idling energy consumption. This difference is eliminated when the AP is turned off if not in

use instead of idling with WiFi on.

Table 4.1: Total number of Whs consumed by an AP per month using the weighted amount of data consumed in 1Q19, sending 100MB chunks

	Wifi standard	Wh Wifi transmitting	Wh WiFi idle	Wh WiFi off	Wh total idle	Wh total off	Total transmitting time (min)
2.4GHz	B	105.89	1553.55	1510.18	1659.44	1616.06	1733.09
	G	71.38	1570.54	1532.55	1641.93	1603.93	1109.84
	N2-HT20	38.57	1597.97	1551.80	1636.53	1590.36	537.76
	N2-HT40	18.79	1635.51	1562.35	1654.30	1581.14	279.66
5GHz	A	53.59	1717.93	1538.58	1771.52	1592.17	941.85
	N5-HT20	28.94	1734.95	1557.29	1763.89	1586.23	420.61
	N5-HT40	14.65	2122.91	1564.96	2137.56	1579.61	207.03
	AC-VHT20	23.34	2062.46	1558.46	2085.80	1581.81	388.00
	AC-VHT40	11.75	2121.17	1565.51	2132.91	1577.26	191.61
	AC-VHT80	5.81	2306.96	1569.16	2312.78	1574.97	90.10

Table 4.1 shows that the 2.4GHz WiFi networks consume the least amount of energy without turning off the WiFi network when not in use (column: Wh total idle). However, the older standards also take their time when transmitting data.

Another interesting observation is that when the WiFi network is turned off outside of the needed data transmission (column: Wh total off) the newest 802.11ac with a 80MHz bandwidth (ac-VHT80) standard becomes the most energy efficient standard.

When idling the energy consumption of this standard is the highest of all standards. But due to the fast transmissions the AP spends less time transmitting and can spend more time sleeping. These results show that by switching to a WiFi standard with lower idle consumption or using wake/sleep (on/off) cycles a lower energy consumption can be achieved.

4.2 Standard Switching

Dynamically switching between the different WiFi standards, means that the AP would choose a older/slower WiFi standard than the standard the AP has currently selected to operate on and switch to that standard when the AP is not actively used by the connected stations. Next to this, as the demand for more and faster data transmission rises that the AP if possible choose a newer/faster WiFi standard. The goal of dynamically switching is to decrease the energy consumption by the AP and at the same time maintain usability and user comfort. The switch frequency, the number of switches that the AP can make in a certain time period, is not defined. This is because if a switch is performed when the network is in use it will disturb the usability of the network and user comfort.

Statically switching is a much simpler variant of switching where the switches happen on specific times. This can for example be setup to be at night when speed requirements drop or switching from an older/slower standard to a newer/faster standard when you are bound to be home.

4.2.1 Drawbacks switching

Dynamically switching or statically switching has a few drawbacks compared to operating without WiFi standard restrictions. The most prominent drawbacks are when such a switch occurs the AP can not send or receive data as the AP has to (partially) reboot and due to that the stations will have to go through the steps explained in section 2.3 to re-authenticate and re-associate with the network. Next to this, if the WiFi network is password protected a key exchange must occur.

4.2.2 Outcome

The results from the experiment performed in section 3 have shown that older WiFi standards consume more energy when transmitting but less energy when idling. This means that standard switching, can lead to a reduction in the energy consumption of the access point.

This reduction should not be sought when transmitting data but when the WiFi point is idling. Because then the extra time the older WiFi standards take to transmit, will vanquish the small amount of power draw difference seen in figure 3.4. However, figure 3.5 shows a significant power draw difference when idling. Next to this, it is apparent from figure 3.5 that among the 2.4GHz standards no switch is needed as no gain can be made other than turning the network off. Moreover, from the newest 5GHz networks a switch could be made to either 5GHz or 2.4GHz 802.11n with a 20MHz bandwidth. The first will ensure a more energy efficient 5GHz network with a slightly higher transfer rate where the later is the most energy efficient switch. Table 4.2 shows that switching to IEEE 802.11g results in similar power reduction when compared to n2-HT20. Next to this, when a switch is made to the IEEE 802.11 b or g standard the user will see significant reduction in user comfort purely due to lower transfer speeds compared to IEEE 802.11n on a 2.4GHz frequency.

To provide an close approximation on the energy that can be conserve using this strategy, we attempt to calculate the total energy consumption of the access point. For this calculation several assumptions must be made:

1. The faster IEEE 802.11ac with its 80MHz bandwidth (AC-VHT80) is

used for 5 hours a day to provide superior speeds than any 2.4GHz standard.

2. Within those 5 hours 50% of the daily data is transmit (175.7 a month, $\frac{365 \text{ days}}{12 \text{ months}} \approx 30.42$ days a month thus $\frac{175.7GB}{30.42days} \approx 5.78$ GB per day, $\frac{5.78GB}{2} \approx 2.89GB$)
3. Within the remaining 19 hours the other 50% is transmit.
4. During the 19 hour period either IEEE 802.11g, both bandwidths 2.4 GHz 802.11n or 5GHz 802.11n with a 20MHz bandwidth (g, n2-HT20, n2-HT40, or n5-HT20) is used.
5. In this example the switches have not been accounted for.

Table 4.2: Energy consumption of GB amount of data within a specified number of hours.

WiFi standard	data (GB)	hours	Time spent transmitting (s)	Amount of J to send data	Amount of J to idle	Total consumption per x hours and 50% of data (J)
AC-VHT80	2.89	5	88.86	344.09	56719.82	57063.90
G	2.89	19	1094.64	4224.33	148567.20	152791.53
N2-HT20	2.89	19	565.90	2282.28	150459.66	152741.95
N2-HT40	2.89	19	275.82	1111.99	153608.15	154720.14
N5-HT20	2.89	19	414.84	1712.51	163143.39	164855.91
AC-VHT80	5.78	24	177.72	688.17	273043.04	273731.21

Table 4.3: Total energy consumption of one month when only both AC-VHT80,N2-HT20 and N2-HT40 compared to only AC-VHT80.

WiFi standard	Total per month (Wh)	% of energy conserved
AC-VHT80+G :	1776.08	23.38
AC-VHT80+N2-HT20:	1772.67	23.40
AC-VHT80+N2-HT40:	1789.38	22.68
AC-VHT80+N5-HT20:	1875.02	18.98
AC-VHT80 (100%):	2314.28	

Within table 4.3 we can see that by switching to a 2.4GHz standard (g, n2-HT20, n2-HT40) 22.68% \sim 23.40% of the energy consumption of the access point can be saved and 18.98% when switching to 5GHz 802.11n with a 20MHz bandwidth (n5-HT20). This is higher but comparable to the 17% the authors of [12] were able to achieve using a wake/sleep cycle and TPC.

Chapter 5

Related Work

Reducing energy consumption of an WiFi access point has seen many research attempts with a few different solutions. We will discuss three different solutions: wake/sleep cycle, TPC, and off loading. The first two of these solutions can be applied to a single AP where the third solution can only be applied to larger networks.

The approach taken by the authors of [12, 13, 15] is putting the AP to sleep in between two separate beacons. OpenWRT uses a default beacon interval of 100 ms¹ which is a small amount of time. This approach is interesting because such a system has been implemented on the station side to increase battery life which is explained in section 2.4. The advantage of such an approach is that the network can still be used. Nevertheless this approach has disadvantages too: more communication is required between the AP and its STAs to time transmitting, the sleep can add to the latency of packets and the sleep time might not be optimal. The authors of [17] noticed the disadvantage of a static sleep time and proposed a dynamic sleep time instead. This, approach is more difficult to implement as the sleep time needs to be determined periodically. However, it can reduce the possible latency added by the sleep time because the sleep time will be smaller within the peak hours. This, is comparable to the approach of dynamically switching described in section 4.2 as this also takes advantages of the lull periods to decrease the amount of energy consumed.

The second approach is TPC used by [12] and [22], which has been explained in section 2.4. In short the AP attempts to keep the power used to transmit messages as close to the power that is required for the STAs to receive the message correctly. The advantage is that the messages under go no extra latency. However, due to the lower transmission power the problem of

¹<https://openwrt.org/docs/guide-user/network/wifi/basic>

hidden station becomes bigger, this will increase possible interference. The authors of [22] use a clear channel assessment to prevent the hidden station problem.

The third approach that will be discussed can only be applied to large scale networks. As the energy consumption is achieved by turning an AP off when there are no STAs connected to it. A good example is from the authors of [6] their simulation showed that 70% of the energy can be conserved while maintaining 92% of the original coverage. Next to this, the authors of [19] take it one step further and even attempt to off load active STAs to other APs if this means that the AP can be turned off. Large savings can be achieved with such an approach but they require a large set of access points to make it worth while and some system to control the APs.

Within the cellular industry similar approaches as described above are taken to reduce the energy consumption of base stations (BSs). For example a sleep control scheme [5] or IEEE 802.16e which supports a sleep mode mechanism [2] and dynamic operation of cellular base stations, in which redundant base stations or parts of them are switched off during lull periods [16, 20]. The authors of [9] present a survey of methods that have been or will be adopted in order to reduce energy consumption of base stations. They list improvements in: the power amplifier, cooling that amplifier, power saving protocols such as sleep modes, energy-aware cooperative BS power management (Turning off a BS depending on load), and renewable energy resources instead of diesel generators. However, 40% to 45% of the energy consumed by a cellular BS is wasted as heat requiring active cooling. Due to the very low transmission power of WiFi of 200mW vs 20W and 120W in 4G and 5G cellular antennas² an WiFi AP does not need active cooling.

²<https://www.grandmetric.com/2019/03/26/5g-health-issues-explained/>

Chapter 6

Conclusions

This thesis conducts an experiment into the energy consumption of the IEEE 802.11 WiFi standards b/a/g/n/ac on an Archer C7 AC1750 access point. Several measurements have been performed to achieve this, a file transfer using 1, 10 and 100MB files and an idle power draw measurement. Every measurement has been performed three times in an environment without interference from cellular signal or other WiFi networks/devices. Next to this, we looked more closely at the energy consumption of an AP using one standard and using standard switching within a period of a month as a way of saving energy in an attempt to curb the growth rate of the energy used by networks.

The experiment performed to gather the information regarding the energy consumption of the access point resulted in several interesting findings. Specifically that, while transmitting the power draw of the older 2.4GHz b and g standard is 0.17 to 0.36W lower compared to both 20MHz and 40MHz bandwidths on 2.4GHz 802.11n and similar holds for the 5GHz standards. Where 802.11a is 0.83 and 0.46 lower compared to 802.11n with a 40MHz bandwidth and 802.11ac with a 80MHz bandwidth respectively. However, due to slower transfer rates of the older standards more energy is consumed when transmitting the same file. Next to this the idle power draw experiment showed that the idle power draw of the 2.4GHz networks are within 0.04W of each other. Within the 5GHz networks a larger difference can be found. The 802.11n standard with a 40MHz bandwidth and all 802.11ac networks draw at least 0.44W more compared to the networks running 802.11a and 802.11n with a 20MHz bandwidth.

The most energy efficient WiFi standard is not 802.11b, 802.11g or 802.11a as one might expect from their low power draw while transmitting and idling. However, 2.4GHz 802.11n with a 20MHz bandwidth is with a total energy

consumption of 1636.53Wh shown in table 4.1 while the network remains on. This result changes when the total energy consumption is calculated when the AP turns off when not in use. In this case the newest 802.11ac with a 80MHz bandwidth standard becomes the most energy efficient standard with a total of 1574.97Wh instead of 2312.78Wh. This difference comes from a high idle power draw combined with the fast transmissions rates of 802.11ac lengthening the time in which the AP can be turned off. Based on these conclusions, the best reduction can be achieved when only using 2.4GHz 802.11n at a 20MHz bandwidth.

The results described in section 4.2 show that standard switching can lead to a reduction in the energy consumption of roughly 23%. However, a key point is that the reduction will come from the difference in idling power draw between the standards and not from using less energy during transmission. Thus, among the 2.4GHz standards no switch is needed as almost no gain can be made other than turning the network off. Moreover, from the newest 5GHz networks a switch could be made to either 5GHz or 2.4GHz 802.11n with a 20MHz bandwidth. The first will ensure a more energy efficient 5GHz network where as the latter is the most energy efficient switch. A switch to 802.11g would not be useful as shown in table 4.3 the percentage of energy conserved by the switch to 802.11g is lower but close to the switch to 802.11n with a 20MHz bandwidth. More importantly, the transmission time of the data almost doubles between these different standards. Impacting user comfort and the usability of the WiFi network.

In this thesis it has been shown that standard switching can help to significantly reduce the energy consumption of an access point. The results in this paper support the solution of utilizing a wake/sleep cycle which has been applied in papers [12, 13, 15, 17]. We recommend application of standard switching to APs, which see varying degrees of activity through out a period of time over a wake/sleep cycle. This, is due to the extra required co-ordination between APs and STAs when using a wake/sleep cycle.

6.1 Future work

The Archer C7 AC1750 is a readily available mid-range router with a price around 70 ~ 100€. This leaves the question of how effective standard switching is regarding low-end or high-end routers open. A possible research option is thus applying standard switching to multiple routers.

Another possible research avenue is the fact that the setup used to gather information in this thesis was systematical and thus further away realistic home use. It would be interesting to apply standard switching to more realistic data. Furthermore, little information regarding the distribution of devices over the WiFi standards is known. With this data more precise estimates can be made regarding the impact of standard switching on the energy consumption of access points.

Bibliography

- [1] M. Afanasyev, T. Chen, G. M. Voelker, and A. C. Snoeren. Usage patterns in an urban wifi network. *IEEE/ACM Transactions on Networking*, 18(5):1359–1372, 2010.
- [2] M. J. Chang, Z. Abichar, and C. Hsu. WiMAX or LTE: Who will Lead the Broadband Mobile Internet? *IT Professional*, 12(3):26–32, 2010.
- [3] M. Collotta, G. Pau, T. Talty, and O. K. Tonguz. Bluetooth 5: A Concrete Step Forward toward the IoT. *IEEE Communications Magazine*, 56(7):125–131, 2018.
- [4] Gary Cook, Jude Lee, Tamina Tsai, Ada Kong, John Deans, Brian Johnson, and Elizabeth Jardim. Clicking clean: Who is winning the race to build a green internet?, 01 2017.
- [5] A. Dataesatu, P. Boonsrimuang, K. Mori, and P. Boonsrimuang. Energy Efficiency Enhancement in 5G Heterogeneous Cellular Networks Using System Throughput Based Sleep Control Scheme. In *2020 22nd International Conference on Advanced Communication Technology (ICACT)*, pages 549–553, 2020.
- [6] L. Fang, G. Xue, F. Lyu, H. Sheng, F. Zou, and M. Li. Intelligent Large-Scale AP Control with Remarkable Energy Saving in Campus WiFi System. In *2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS)*, pages 69–76, 2018.
- [7] Shahin Farahani. Chapter 1 - zigbee basics. In Shahin Farahani, editor, *ZigBee Wireless Networks and Transceivers*, pages 1 – 24. Newnes, Burlington, 2008.
- [8] Matthew Gast. *802.11 Wireless Networks The Definitive Guide*. O’Reilly, 04 2005.
- [9] Z. Hasan, H. Boostanimehr, and V. K. Bhargava. Green Cellular Networks: A Survey, Some Research Issues and Challenges. *IEEE Communications Surveys Tutorials*, 13(4):524–540, 2011.

- [10] Ward Van Heddeghem, Sofie Lambert, Bart Lannoo, Didier Colle, Mario Pickavet, and Piet Demeester. Trends in worldwide ict electricity consumption from 2007 to 2012. *Computer Communications*, 50:64 – 76, 2014. Green Networking.
- [11] IEEE. IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications. *IEEE Std 802.11-1997*, pages 1–445, 1997.
- [12] K. C. Jeong, W. S. Choi, and S. G. Choi. A wireless AP power saving algorithm by changing operating mode and altering transmission power in IEEE 802.11 WLAN. In *2014 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 9–10, 2014.
- [13] A. M. Kholaf, T. D. Todd, P. Koutsakis, and M. N. Smadi. QoS-Enabled Power Saving Access Points for IEEE 802.11e Networks. In *2008 IEEE Wireless Communications and Networking Conference*, pages 2331–2336, 2008.
- [14] James F. Kurose and Keith W. Ross. *Computer Networking A Top-Down Approach*. Pearson Education Limited, 11 2016.
- [15] Masakatsu Ogawa, Takefumi Hiraguri, Kentaro Nishimori, Naoki Honma, Kazuhiro Takaya, and Kazuo Murakawa. Power Saving Control Method for Battery-Powered Portable Wireless LAN Access Points in an Overlapping BSS Environment. *IEICE Transactions*, 94-B:658–666, 03 2011.
- [16] E. Oh, B. Krishnamachari, X. Liu, and Z. Niu. Toward dynamic energy-efficient operation of cellular network infrastructure. *IEEE Communications Magazine*, 49(6):56–61, 2011.
- [17] H. Oh, S. H. S. Newaz, S. Park, and J. K. Choi. Maximizing energy efficiency in off-peak hours: A novel sleep scheme for WLAN access points. In *NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium*, pages 342–348, 2016.
- [18] OpenVault. OpenVault Broadband Industry Report 1Q 2019, 2019.
- [19] R. Riggio, C. Sengul, L. Suresh, J. Schulz-zander, and A. Feldmann. Thor: Energy programmable wifi networks. In *2013 IEEE Conference on Computer Communications Workshops (INFOCOM WK-SHPS)*, pages 21–22, April 2013.
- [20] K. Samdanis, D. Kutscher, and M. Brunner. Self-organized energy efficient cellular networks. In *21st Annual IEEE International Symposium*

on Personal, Indoor and Mobile Radio Communications, pages 1665–1670, 2010.

- [21] T. Sharma, K. Ritesh, N. Chauhan, and S. Agarwal. Analogous study of 4g and 5g. In *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 2137–2140, 2016.
- [22] Youngsoo Kim, Jeonggyun Yu, and Sunghyun Choi. SP-TPC: a self-protective energy efficient communication strategy for IEEE 802.11 WLANs. In *IEEE 60th Vehicular Technology Conference, 2004. VTC2004-Fall. 2004*, volume 3, pages 2078–2082 Vol. 3, 2004.

Appendix A

Appendix

A.1 Detailed AP setup

For this experiment a few criteria should be met by the Access Point: Support up to IEEE 802.11ac, OpenWRT 19.07.02 capable and widely available. The access point thus became a TP-LINK Archer C7 AC1750 EU v2 it is an mid-range router that supports up to IEEE 802.11ac with a maximum speed of 450 Mbps and 1,3 Gbps on 2,4 GHz and 5GHz respectively. The AP can host a File Transfer Protocol (FTP) server. It is thus possible to remove the 'Server' part from the setup, however, this would diminish the resemblance of the setup to a typical AP setups. Next to that, it would burden the AP with running a FTP-server and serving the file. Which will unnecessarily cause an increase in power draw. The setup is displayed in figure A.1.

Figure A.1: Image of the setup within the interferenceless environment



AP WiFi Settings

The AP needs to be told which WiFi settings to create a WiFi network with. Among these settings are what WiFi standard to allow/use and which bandwidth to use in OpenWRT these settings are called hwmode and htmode respectively. Within this paper these two settings will be used to force the AP in to creating connections with the specified standard. Two ways exist to adjust the settings of the network namely the web interface that is integrated in to OpenWRT and a config file which can be accessed using SSH. Within this paper the SSH method is used to adjust the settings of the network directly. One of the configuration of the 5 GHz network used is displayed in listing A.1.

```
1 config wifi-device 'radio0'
2     option type 'mac80211'
3     option hwmode '11a'
4     option path 'pci0000:00/0000:00:00.0'
5     option channel 'auto'
6
7 config wifi-iface 'default_radio0'
8     option device 'radio0'
9     option network 'lan'
10    option mode 'ap'
11    option encryption 'none'
12    option ssid 'OpenWrt'
```

Listing A.1: 5 GHz IEEE 802.11a network settings

Visible here are the various settings used for the network but the interesting ones are hwmode and htmode or a lack thereof. The hwmode is set to the 5 GHz WiFi standard a, this standard does not come with the option to adjust the bandwidth. The option hwmode only supports the older versions of IEEE 802.11 (a/b/g), meaning that hwmode '11ac' does not work. However, that is where htmode can force IEEE 802.11 n and ac standards¹. Because out of the standards used n and ac are the only two that have the capability to adjust the bandwidth. Thus if the option of htmode is added to the wireless config file the AP will create a network using that bandwidth. Which are either 'HT20' or 'HT40' for IEEE 802.11 n and 'VHT20', 'VHT40', and 'VHT80' for IEEE 802.11ac. An example of this is listing A.2 which displays the wireless config file for a network using IEEE 802.11ac 80MHz.

```
1 config wifi-device 'radio0'
2     option type 'mac80211'
3     option hwmode '11a'
```

¹<https://openwrt.org/docs/guide-user/network/wifi/basic>

```
4      option htmode 'VHT80'  
5      option path 'pci0000:00/0000:00:00.0'  
6      option channel 'auto'  
7      ...
```

Listing A.2: 5 GHz IEEE 802.11ac 80MHz network settings

Appendix B

Energy and power the difference

This paper will use the terms energy and power frequently. Energy is the cost to perform work or do something whereas power is the rate of energy per unit of time. Within this paper the amount of energy that it took the AP to perform a certain task will be measured. Because not everyone might be familiar with energy and power their meaning will be explained using a simple example, Lets compare an electricity cable with a water pipe. Through this water pipe flows 1 liter of water per second. This can be compared to power. As power is how much water flows through the pipe per unit of time. Energy is how much water has flowed through the pipe, for this we need to know how long the water has flown through the pipe and how fast. Say that this was 1 minute (60s), then $60s \cdot 1l/s = 60\text{liters}$ would have come out of the end of the pipe.

However for our AP it is not that simple. The problem is that the power draw or flow will not be consistent. This means that very frequent measuring should take place to ensure that at the end of the performed work accurate energy calculations can be made.

Appendix C

Measurement code explained

Within this appendix we will explain the ins and outs of the code that is used to measure the amount of energy draw by the AP using the voltage/current v2 bricklets API¹. The code can be found on GitLab²

The program that performs the measurements uses libcurl³ to fetch the files from the server onto the query device. The code segments below are all the code surrounding the retrieval of the file within the experiment and is mainly setting up curl and executing it.

```
1 static size_t write(void* buffer, size_t size, size_t nmemb,  
    void* param){  
2     std::string& text = *static_cast<std::string*>(param);  
3     size_t totalsize = size * nmemb;  
4     text.append(static_cast<char*>(buffer), totalsize);  
5     return totalsize;  
6 }
```

The above function `write` is a callback used by curl in the next code segment. Here curl writes the incoming data to memory instead of to disk space. This should prevent waiting on write IO operations.

```
1 int retrieve_File(string* result, const char* url) {  
2     CURLcode res;  
3     CURL* curl = curl_easy_init();
```

¹https://www.tinkerforge.com/en/doc/Software/Bricklets/VoltageCurrentV2_Bricklet_C.html#voltage-current-v2-bricklet-c-api

²<https://gitlab.science.ru.nl/gnoordbruus/EnergyConsumption>

³<https://curl.haxx.se/>

```

4     curl_global_init(CURL_GLOBAL_DEFAULT);
5
6     if (url == ""){
7         url = "TEST_URL";
8     }
9
10    if (curl) {
11        curl_easy_setopt(curl, CURLOPT_URL, url);
12        curl_easy_setopt(curl, CURLOPT_WRITEFUNCTION, write);
13        curl_easy_setopt(curl, CURLOPT_WRITEDATA, result);
14        curl_easy_setopt(curl, CURLOPT_VERBOSE, 1L);
15
16        res = curl_easy_perform(curl);
17
18        curl_easy_cleanup(curl);
19
20        if (res != CURLE_OK) {
21            std::cerr << "curl_error:" << res << '\n';
22            return 1;
23        }
24    }
25    else {
26        return 1;
27    }
28    curl_global_cleanup();
29    return 0;
30 }

```

With this code segment line 3 and 4 setup the curl environment to perform the http request. On line 11 we set the URL of the file. Next to that, 12 and 13 set the function that is used to write the incoming data to a string and set the string that data should be written to. The actual curl request is performed on line 16 which returns a int which indicates either success or failure of the curl http request. Line 20-24 verifies if the request has been performed correctly. If it did not we return 1 meaning failure. If curl was not able to perform setup properly we also return failure on line 26. Then finally on line 18 and 29-30 all resources that curl used are released and success is returned.

There are two ways to get power measurements from the voltage current bricklet. One of these is using a command `voltage_current_v2_get_power` which returns an int with the current power usage in mW. Because large amounts of measurements need to be taken it is better and advised by tin-

kerforge to use a callback function. This function is just like the callback function written above. Once the callback function is registered the brick daemon calls it when some conditions are met. The power callback function that is used within this paper is:

```
1 void callBackPower(int32_t power, void* user_data) {  
2     (void)user_data; // Avoid unused parameter warning  
3  
4     dataPoint point;  
5     point.power = power;  
6     dataVector.push_back(point);  
7 }
```

The callback has default arguments of which only one is needed. This is the `int32_t power`, using the newly received measurement a new `dataPoint` is created. The newly created datapoint named `point` is a struct with a timestamp in epoch ms and an int power. So every time this function is called the power and timestamp are recorded such that afterwards the energy consumption can be calculated.

The main function is used to get all user input with the function input. This sets the needed URL and filename to which the result is written at the end. Then we create and initialize an IP connection with brick daemon, this daemon is a service that handles communication between the bricklets and our API calls, the voltage current bricklet object is initialized too. Now that the everything is setup, the callback function for the power can be registerd and how often it should be called. The arguments for `voltage_current_v2_set_power_callback_configuration` are the voltage-current bricklet object, the number of ms between each callback, if the value(power) needs to change before calling, Constraint field, a minimum and a maximum value field. Now that the callback is set the voltage current bricklet will utilize the callback immediately. Thus, the file retrieval is started, and directly after retrieval the connection and voltage current bricklet are destroyed. The results are put out to the indicated file.

```
1 int main(void) {  
2     char url[1024];  
3     string filename;  
4     string result;  
5  
6     input(url, &filename);  
7  
8     ipcon_create(&ipcon);
```

```

9   voltage_current_v2_create(&vc, UID, &ipcon);
10  if (ipcon_connect(&ipcon, HOST, PORT) < 0) {
11      cout << "Could not connect\n";
12      return 1;
13  }
14
15  voltage_current_v2_register_callback(&vc,
16      VOLTAGE_CURRENT_V2_CALLBACK_POWER, (void (*)(void))
17      callBackPower, NULL);
18  voltage_current_v2_set_power_callback_configuration(&vc,
19      10, false, 'x', 0, 0);
20
21  int retrieval = retrieve_File(&result, url);
22
23  voltage_current_v2_destroy(&vc);
24  ipcon_destroy(&ipcon);
25
26  output(&result, filename, url);
27
28  return 0;
29 }

```

The idle power consumption of the access point has also been measured in section 3. The Voltage current bricklet calls the callback function every 10ms. One second is 1000ms thus we need a 100 measurements to have the results of 1 second. This while loop prints the progress this is due to a problem mentioned at the end of section 3.1. To perform this test, simply replace line 18 of the main function with `idle()`;

```

1  void idle(){
2      int second = 100;
3      int minute = 6000;
4
5      while (dataVector.size() <= (minute)) {
6          cout << "Size: " << dataVector.size() << "needed size:
7          " << minute << "\n";
8      }
9  }

```

Appendix D

Raw measurement results

D.1 Experiment results 2.4Ghz

Table D.1: Experiment results of 1MB text file and 2.5GHz

WiFi Standard	1MB file	Run 1	Run 2	Run 3	Avg
B	Avg (mW)	3308.72	3293.22	3292.41	3298.12
	Highest (mW)	3663.00	3554.00	3614.00	3610.33
	Time (ms)	670.00	670.00	620.00	653.33
	Energy (J)	2216.84	2206.46	2041.30	2154.87
G	Avg (mW)	3808.42	3466.48	3403.93	3559.61
	Highest (mW)	4345.00	4028.00	3918.00	4097.00
	Time (ms)	370.00	410.00	420.00	400.00
	Energy (J)	1409.12	1421.26	1429.65	1420.01
N2-HT20	Avg (mW)	2939.65	2798.89	2946.64	2895.06
	Highest (mW)	3115.00	3953.00	3832.00	3633.33
	Time (ms)	190.00	180.00	210.00	193.33
	Energy (J)	558.53	503.80	618.79	560.38
N2-HT40	Avg (mW)	3053.00	2834.00	2675.00	2854.00
	Highest (mW)	3053.00	2981.00	2675.00	2903.00
	Time (ms)	100.00	110.00	140.00	116.67
	Energy (J)	305.30	311.74	374.50	330.51

Table D.2: Experiment results of 10MB text file and 2.5GHz

WiFi Standard	10MB file	Run 1	Run 2	Run 3	Avg
B	Avg (mW)	3568.11	3564.04	3584.01	3572.05
	Highest (mW)	4136.00	4038.00	4184.00	4119.33
	Time (ms)	6220.00	6340.00	6180.00	6246.67
	Energy (J)	22193.65	22596.00	22149.17	22312.94
G	Avg (mW)	3887.71	3887.15	3890.54	3888.47
	Highest (mW)	4368.00	4710.00	4381.00	4486.33
	Time (ms)	3560.00	3710.00	3610.00	3626.67
	Energy (J)	13840.26	14421.32	14044.86	14102.15
N2-HT20	Avg (mW)	3862.84	3822.03	4037.09	3907.32
	Highest (mW)	4027.00	4050.00	4539.00	4205.33
	Time (ms)	1950.00	1690.00	1870.00	1836.67
	Energy (J)	7532.53	6459.23	7549.35	7180.37
N2-HT40	Avg (mW)	3712.18	3723.08	3720.56	3718.61
	Highest (mW)	4002.00	3990.00	4027.00	4006.33
	Time (ms)	960.00	940.00	900.00	933.33
	Energy (J)	3563.69	3499.70	3348.50	3470.63

Table D.3: Experiment results of 100MB text file and 2.5GHz

WiFi Standard	100MB file	Run 1	Run 2	Run 3	Avg
B	Avg (mW)	3632.91	3679.77	3686.67	3666.45
	Highest (mW)	4404.00	4611.00	4478.00	4497.67
	Time (ms)	61170.00	58780.00	57600.00	59183.33
	Energy (J)	222225.01	216297.16	212352.01	216958.06
G	Avg (mW)	3881.06	3838.22	3858.93	3859.40
	Highest (mW)	4502.00	4551.00	4660.00	4571.00
	Time (ms)	37060.00	38580.00	38060.00	37900.00
	Energy (J)	143832.07	148078.56	146871.01	146260.55
N2-HT20	Avg (mW)	4036.35	4031.71	4030.97	4033.01
	Highest (mW)	4563.00	4575.00	4563.00	4567.00
	Time (ms)	19830.00	19450.00	19500.00	19593.33
	Energy (J)	80040.81	78416.67	78603.98	79020.49
N2-HT40	Avg (mW)	4032.99	4027.33	4034.17	4031.50
	Highest (mW)	4502.00	4527.00	4491.00	4506.67
	Time (ms)	9600.00	9520.00	9530.00	9550.00
	Energy (J)	38716.72	38340.14	38445.65	38500.84

D.2 Experiment results 5Ghz

Table D.4: Experiment results of 1MB text file and 5GHz

WiFi Standard	1MB file	Run 1	Run 2	Run 3	Avg
A	Avg (mW)	3063.88	3090.67	3095.88	3083.48
	Highest (mW)	3333.00	3369.00	3357.00	3353.00
	Time (ms)	330.00	350.00	320.00	333.33
	Energy (J)	1011.08	1081.73	990.68	1027.83
N5-HT20	Avg (mW)	3706.00	3287.53	3623.26	3538.93
	Highest (mW)	4234.00	3296.00	3870.00	3800.00
	Time (ms)	170.00	160.00	180.00	170.00
	Energy (J)	630.02	526.00	652.19	602.74
N5-HT40	Avg (mW)	3332.00	3354.20	3332.00	3339.40
	Highest (mW)	3332.00	3369.00	3332.00	3344.33
	Time (ms)	80.00	90.00	90.00	86.67
	Energy (J)	66.56	301.88	299.88	289.44
AC-VHT20	Avg (mW)	2922.07	2926.33	2892.87	2913.76
	Highest (mW)	3200.00	3249.00	2919.00	3122.67
	Time (ms)	140.00	140.00	140.00	140.00
	Energy (J)	409.09	409.69	405.00	407.93
AC-VHT40	Avg (mW)	2893.55	2933.78	2870.00	2899.11
	Highest (mW)	2907.00	3115.00	2870.00	2964.00
	Time (ms)	100.00	80.00	90.00	90.00
	Energy (J)	289.35	234.70	258.30	260.79
AC-VHT80	Avg (mW)	3139.00	3127.00	3237.00	3167.67
	Highest (mW)	3139.00	3127.00	3237.00	3167.67
	Time (ms)	60.00	60.00	50.00	56.67
	Energy (J)	188.34	187.62	161.85	179.27

Table D.5: Experiment results of 10MB text file and 5GHz

WiFi Standard	10MB file	Run 1	Run 2	Run 3	Avg
A	Avg (mW)	3435.17	3334.95	3439.72	3403.28
	Highest (mW)	3846.00	3821.00	3833.00	3833.33
	Time (ms)	3190.00	3220.00	3200.00	3203.33
	Energy (J)	10958.19	10738.55	11007.11	10901.28
N5-HT20	Avg (mW)	3967.67	3956.11	3718.57	3880.79
	Highest (mW)	4380.00	4062.00	4575.00	4339.00
	Time (ms)	1460.00	1420.00	1470.00	1450.00
	Energy (J)	5792.80	5617.68	5466.30	5625.59
N5-HT40	Avg (mW)	3975.74	3988.85	3610.85	3858.48
	Highest (mW)	4196.00	4208.00	4196.00	4200.00
	Time (ms)	720.00	780.00	750.00	750.00
	Energy (J)	2862.53	3111.30	2708.14	2893.99
AC-VHT20	Avg (mW)	3718.52	3438.01	3514.19	3556.91
	Highest (mW)	4014.00	3554.00	4002.00	3856.67
	Time (ms)	1320.00	1370.00	1330.00	1340.00
	Energy (J)	4908.44	4710.08	4673.87	4764.13
AC-VHT40	Avg (mW)	3563.18	3489.91	3829.32	3627.47
	Highest (mW)	3990.00	3652.00	4026.00	3889.33
	Time (ms)	660.00	660.00	670.00	663.33
	Energy (J)	2351.70	2303.34	2565.65	2406.90
AC-VHT80	Avg (mW)	3431.31	3432.35	3465.24	3442.97
	Highest (mW)	3954.00	3905.00	3906.00	3921.67
	Time (ms)	280.00	300.00	320.00	300.00
	Energy (J)	960.77	1029.71	1108.88	1033.12

Table D.6: Experiment results of 100MB text file and 5GHz

WiFi Standard	100MB file	Run 1	Run 2	Run 3	Avg
A	Avg (mW)	3403.08	3425.99	3412.63	3413.90
	Highest (mW)	3846.00	3833.00	3845.00	3841.33
	Time (ms)	32840.00	31820.00	31830.00	32163.33
	Energy (J)	111757.17	109014.97	108623.86	109798.67
N5-HT20	Avg (mW)	4147.20	4120.04	4117.07	4128.10
	Highest (mW)	4611.00	4587.00	4599.00	4599.00
	Time (ms)	14360.00	14130.00	14600.00	14363.33
	Energy (J)	59553.83	58216.10	60109.29	59293.07
N5-HT40	Avg (mW)	4247.52	4240.23	4246.24	4244.66
	Highest (mW)	4624.00	4611.00	4624.00	4619.67
	Time (ms)	6990.00	7240.00	6980.00	7070.00
	Energy (J)	29690.15	30699.30	29638.77	30009.41
AC-VHT20	Avg (mW)	3640.24	3592.81	3596.14	3609.73
	Highest (mW)	4062.00	3990.00	4050.00	4034.00
	Time (ms)	13280.00	13230.00	13240.00	13250.00
	Energy (J)	48342.35	47532.91	47612.85	47829.37
AC-VHT40	Avg (mW)	3663.27	3715.93	3657.13	3678.78
	Highest (mW)	4014.00	4026.00	4026.00	4022.00
	Time (ms)	6370.00	6350.00	6910.00	6543.33
	Energy (J)	23335.06	23596.17	25270.78	24067.34
AC-VHT80	Avg (mW)	3916.16	3856.54	3845.20	3872.63
	Highest (mW)	4099.00	4087.00	4026.00	4070.67
	Time (ms)	3020.00	3060.00	3150.00	3076.67
	Energy (J)	11826.81	11801.00	12112.38	11913.40

Appendix E

Calculation of access point energy consumption per month

Table E.1: Total number of Whs consumed by an AP per month using the weighted amount of data consumed in 1Q19. sending 10MB chunks (Europe: 175.7GB per month)[18]

		Wh Wifi transmitting	Wh WiFi idle	Wh WiFi off	Wh total idle	Wh total off
2.4GHz	B	108.90	1550.00	1506.72	1658.90	1615.62
	G	68.83	1572.30	1534.27	1641.13	1603.09
	N2-HT20	35.04	1599.30	1553.08	1634.34	1588.13
	N2-HT40	16.94	1635.75	1562.58	1652.69	1579.52
5GHz	A	53.20	1718.08	1538.72	1771.29	1591.92
	N5-HT20	27.46	1734.79	1557.15	1762.25	1584.61
	N5-HT40	14.12	2122.30	1564.51	2136.42	1578.63
	AC-VHT20	23.25	2062.25	1558.31	2085.50	1581.56
	AC-VHT40	11.75	2121.04	1565.42	2132.78	1577.17
	AC-VHT80	5.04	2307.08	1569.24	2312.12	1574.28

Table E.2: Total number of Whs consumed by an AP per month using the weighted amount of data consumed in 1Q19. sending 1MB chunks (Europe: 175.7GB per month)[18]

		Wh Wifi transmitting	Wh WiFi idle	Wh WiFi off	Wh total idle	Wh total off
2.4GHz	B	105.17	1546.90	1503.71	1652.07	1608.88
	G	69.30	1568.28	1530.34	1637.58	1599.65
	N2-HT20	27.35	1598.25	1552.07	1625.60	1579.42
	N2-HT40	16.13	1633.18	1560.13	1649.31	1576.26
5GHz	A	50.16	1716.56	1537.35	1766.72	1587.51
	N5-HT20	29.42	1731.86	1554.52	1761.28	1583.94
	N5-HT40	14.13	2120.63	1563.28	2134.76	1577.41
	AC-VHT20	19.91	2061.42	1557.68	2081.33	1577.58
	AC-VHT40	12.73	2117.67	1562.93	2130.39	1575.66
	AC-VHT80	8.75	2302.96	1566.44	2311.71	1575.19