Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Computer Engineering and
Networks Laboratory

Sandra Siby

Reducing energy consumption in hybrid wireless communication

Semester Thesis SA-2012-022
September 2012 to August 2013

Tutor: Dr. Karin Hummel
Co-Tutor: Dr. Franck Legendre
Supervisor: Dr. Bernhard Plattner

Abstract

In the aftermath of a disaster, there is a high probability that the telecommunication infrastructure is being destroyed. Rescue teams require a method to communicate during such scenarios using their mobile devices and ad-hoc opportunistic data as well as infrastructure networks. However, such a scenario requires that energy consumption of the devices be kept to a minimum. This is the inspiration for this project. The goal of this thesis is to minimize the energy consumption of individual devices belonging to a group of devices while maximizing the group's devices lifetime by sharing the load of cellular communications and to design efficient data exchange strategies among the group members. We achieve this by performing three major tasks. First, we provide a state model for the device. Second, we measure the power consumption of each state using different methods. These methods' advantages and shortcomings are discussed in this thesis. Third, we introduce the concept of delayed messaging by queuing in order to reduce power consumption. Finally, the measurements and the queuing method are used in three different scenarios to validate the proposed power saving methods.

Acknowledgements

I would like to thank the following persons with whom I have the pleasure of working:

Dr. Karin Anna Hummel and Dr. Franck Legendre for their valuable advice and assistance over the course of my project.

Dr. Theus Hossmann for his guidance during the experimental phase of my project.

Prof. Dr. Bernhard Plattner for his suggestions during my mid-semester review.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

After a disaster, there is a high probability that regular cellular telecommunication infrastructure is being impaired or destroyed. Rescue teams, which arrive on site to assess the situation and organize search and rescue of victims, need a method to communicate. They usually set up local temporary and stationary telecommunication infrastructure at the base camp headquarters in order to coordinate with central points of coordination and remote task forces. However, these modes are usually not transportable and only provide local connectivity. Rescue teams going on reconnaissance are often equipped with AM radios, which only provide voice communication and no support for data. Measures that provide data support (for example NetHope's Network Relief Kit) are usually bulky and expensive.

A lot of data such as pictures, videos etc. can be be collected using today's smartphones. Smartphones also have the capability to communicate directly via Bluetooth or Wi-Fi without requiring any supporting network infrastructure while on the go. This leads to the possibility of implementing communication between rescue team members for data exchange and local coordination using smartphones. In addition to this, rescue teams could communicate with their local headquarters through cellular communication (3G, EDGE,LTE, etc.).

While implementing such communication, it is imperative that energy consumption has to be considered. Rescue team members often spend hours performing search and rescue requiring continuous communication. On the other hand, phones' batteries are drained fast and not running out of power is essential for rescue missions. This scenario is the inspiration behind this project. The goal of this thesis is to minimize the energy consumption of a group of devices and maximize the group's lifetime.

First, a state model is introduced in which the energy states of the communicating devices are defined. Measurements are taken to get an idea of the energy consumed in each state. Then, a queue mechanism is proposed to reduce the energy consumption. Finally, the state model and queue mechanism are introduced in a use case scenario to confirm that implementing these changes do indeed lead to energy saving. We consider the use of two communication technologies - Wi-Fi and 3G cellular.

The thesis is organized as follows: Chapter 2 examines some of the existing work that has been done in the areas of opportunistic communication, energy measurements in cellphones and mobility of rescue workers. In Chapter 3, we define a state model for the device. Chapter 4 deals with measurements that were taken to enhance the mobile design. It also compares results of different measurement methods. Chapter 5 introduces a queuing mechanism to achieve lower power consumption. Finally, in Chapter 6, the design is evaluated and improvements are suggested.

Chapter 2

Literature Review

When a disaster strikes, the mobility of the rescue team determines device contacts and opportunistic networking options. In [1], real-life mobility during a disaster is described and shows us that it is possible to model mobility during a disaster scenario. In this project, we refer to papers from the following areas: Energy measurements in cellphones and opportunistic communication. Energy measurement is of interest to us because our goal is to optimize the energy consumption of the mobile device and we are interested in measuring how each component of the phone contributes to its total energy consumption. We also consider opportunistic communication because it is useful in scenarios where communication infrastructure is wholly or partially destroyed and we would like to find out how we can introduce this into our model.

2.1 Energy measurements

In [2], measurements were performed to learn about energy usage when 3G and Wi-Fi are switched on. It was found that Wi-Fi has a high wakeup/ connection maintenance energy due to scanning and association but low energy per bit transmission. On the other hand, 3G has a low connection maintenance energy but high energy per bit transmission. Also, 3G has a high 'tail energy', i.e. the phone remains in a high power state for a long time after a data transfer. The conclusion that can be drawn is that it is inefficient to switch on Wi-Fi or 3G when the amount of data to be transferred is low. Also, Wi-Fi consumes less power in total and should be used whenever available rather than 3G (availability of Wi-Fi usually tends to be less than 3G). Finally, the paper implements an application called TailEnder, where data is prefetched and bundled so that it can be sent in large chunks. This application was able to reduce energy consumption by at least 35%. This is an inspiration for the energy saving methods we propose in this thesis.

In [3], a detailed measurement of the components of the OpenMoko Neo Freerunner phone is performed by attaching sensor resistors to each component. Components measure were Wi-Fi, GSM, CPU, RAM, graphics and audio. HTC Dream and Google Nexus One phones are also analysed, albeit to a lesser extent. The paper showed how different components contribute to power consumption and how models based on different usage patterns can be derived from these measurements - a useful study for our Evaluation section (Chapter 5). However, since the experiment used an older device model that lacked a 3G interface, the study did not provide us with information on 3G power consumption.

In [4], a power and throughput study of Wi-Fi and Bluetooth in smartphones running Windows Mobile and Android operating systems was conducted. The study provides a few interesting conclusions. It measures power consumed in various states like when the device is scanning, when it is connected to an access point etc. The power values indicate higher power consumption for Bluetooth over Wi-Fi except when the Wi-Fi is connected to an adhoc network. It shows that the PSM (Power Saving Mode) of Wi-Fi is effective and that Wi-Fi does not consume much power when it is not used for sending data. It also proves that despite belief to the contrary, Bluetooth's utility in terms of power saving per data transfer is not that much higher than Wi-Fi, meaning that Wi-Fi can be used frequently even if both are available. This paper further solidified our decision to not include Bluetooth in our model.

In [5], the average energy consumption when transmitting data via Bluetooth, Wi-Fi and 3G, is analysed by testing on an HTC Wizard phone. The results agree with the findings of [2].

From these readings, we came to the conclusion that using Wi-Fi whenever possible rather than 3G could potentially lead to energy savings. Also, storing and sending data in bundles might reduce the energy consumed due to lower frequency of switching the network components on and off.

2.2 Delay Tolerant Networking

Delay Tolerant Networking or Opportunistic networking is an approach used when there is a situation where there is no full connectivity in the network. In opportunistic networks, the communication is multihop with intermediate nodes acting as routers that forward the messages addressed to other nodes. These intermediate nodes store messages meant for other nodes if they do not find any forwarding opportunities at a particular time. When they come in contact with another node, they forward the stored messages [6].

Opportunistic networks are of interest to us because network infrastructure is usually destroyed during a disaster, leading to connectivity problems and we are interested in leveraging the phones to act as forwarding agents. In [7], a scheme called Wi-Fi-Opp is proposed that forms the basis for the model described in Chapter 3. Wi-Fi-Opp leverages the mobile Wi-Fi Access Point (AP) feature (also known as tethering) as well as stationary open APs to support opportunistic communications among devices. The paper explores 5 modes of Wi-Fi-Opp - Static, Flexible AP, Flexible STA, Manual and Fixed APs. In Static mode, the device scans for an AP for a fixed amount of time; it connects to one if found and if no AP is found, it elects itself as an AP when its scanning time is over. In Flexible AP mode, an AP is given he flexibility of stopping its AP functionality after a fixed time even if it has stations (STA) connected to it. The Flexible STA mode enables stations to switch connections between APs. Manual mode leverages the ability of the user to switch on and off the AP functionality. Open AP mode makes use of public access points to which the device can connect. We use the concepts of Flexible AP, Flexible STA and open AP in our model.

Chapter 3

Device State Definition

In order to get energy measurements of the phone, we have to define the states in which it can be. We assume that the device can be in 2 modes - Infrastructure or Ad-hoc. In the Infrastructure mode, a network is present and phones can send data via 3G, e.g., to the headquarter in a rescue mission. The phone can be in state '3G On' or '3G Off'. In the Ad-hoc mode, we take inspiration from the Wi-Fi-Opp model described in the previous chapter [7], where a device can be an access point (AP), a station (STA) or in the scanning state. When a device is in AP state, it provides hotspot functionality. Devices can connect to this AP and send/receive data. Hence, when a device is in AP state, it can either have zero connections or have devices connected to it. A device connected to an AP is considered to be a station (STA). When devices are neither an AP or a STA, they are in scanning state, where thy scan for an AP.

In the Wi-Fi-Opp model, transitions between states are based on predetermined time values. In contrast, we do not consider these for now and just

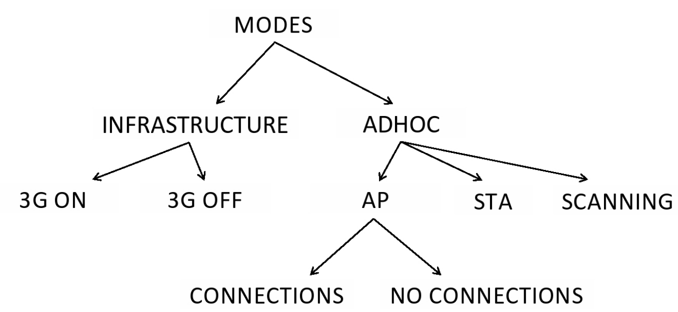


Figure 3.1: Device states for infrastructure and ad-hoc networking.

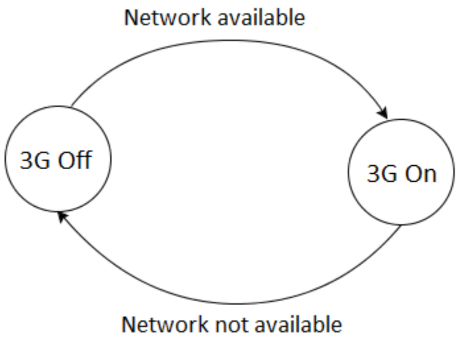


Figure 3.2: Infrastructure mode states.

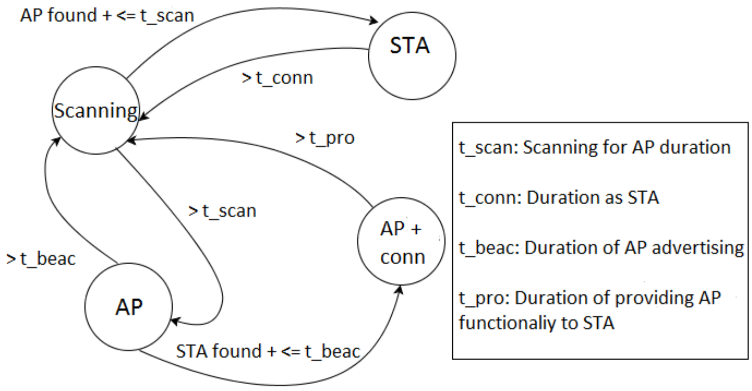


Figure 3.3: Adhoc mode states.

assume that the phone can be in AP, AP + connections, STA or scanning states.

An overview of the modes is shown in Figure 3.1. The transitions between the states for each mode are shown in Figure 3.2 and Figure 3.3.

Chapter 4

Measurements

Power measurements were taken in order to provide realistic input to the proposed model. Measurements of the energy consumption of Wi-Fi, 3G and Tethering modes were performed on the Android Nexus One phones. We measured power following three approaches: using the PowerTutor app, implementing our own Battery Measurement app and using an external power meter, the Monsoon Power Monitor.

4.1 PowerTutor

PowerTutor is an app available on the Android market that displays the power consumed by each component. Components measured by the app are the display, CPU, Wi-Fi, 3G and GPS. The app also shows the power consumed by all the apps running on the phone. An example screen-shot of the app is shown in Figure 4.1. It is a real time graph display of the power consumed by each component (power measured in mW and time in seconds).

We ran the app on the phone with various components switched on and off to obtain the power values. It was possible to get readings for the Nexus One phones but not for the Galaxy S3; there were no values for Wi-Fi and 3G when the app was run on Galaxy S3. This is due to the power models used by the PowerTutor App. According to [8], the paper on which the app is based, two phones - the HTC Dream and the HTC Magic, had been connected to a power meter to obtain some base measurements of their components. These base values would then be updated using some power models that had been developed by the creators of the app. While this method seemed reasonable to us for the phones on which the initial measurements had been performed, every phone would have to be modeled in a similar way for the app to give accurate results. However, PowerTutor used these values for

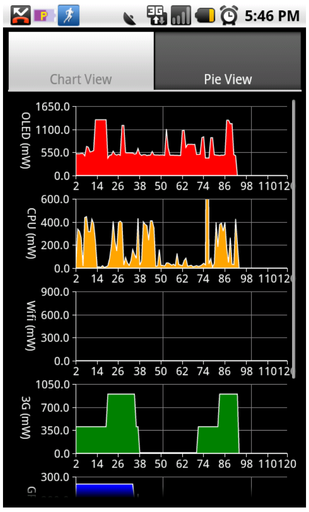


Figure 4.1: PowerTutor screen-shot.

other phone models as well and the source code further revealed there to be an incomplete section for the Nexus One. Thus, the accuracy of the app is questionable.

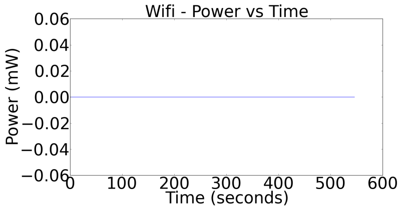
Some of the measurements obtained using PowerTutor are explained in the following subsections. We conduct 3 runs for each measurement, with each run being 5 minutes long.

4.1.1 Baseline

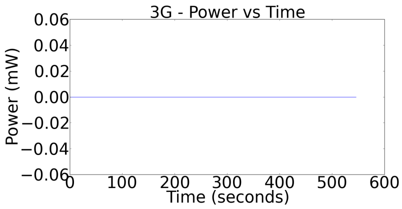
The power profile when neither Wi-Fi nor 3G is present is shown in Figure 4.2. The total power is around 410 mW on average because of the screen being switched on. The screen is switched on because PowerTutor logs power values every second only when the screen is on. When the screen is off, it logs values about once every minute approximately, which might lead to a less accurate analysis.

4.1.2 Wi-Fi

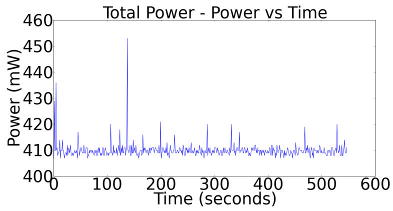
The power profile (Wi-Fi and Total Power) when only the Wi-Fi is turned on is present in Figure 4.3. In the first case, the phone is left as it is without any apps running on it. In the rest of the cases, different activities are carried out (surfing, watching a YouTube video and making a Skype call) and the power is measured. We notice that the power profile for Wi-Fi is always 34 mW in low power states and 400 mW in high power states, irrespective of the activity. These values were found to be hard-coded in the PowerTutor



(a) Wi-Fi.



(b) 3G.



(c) Total Power.

Figure 4.2: PowerTutor: Baseline measurements

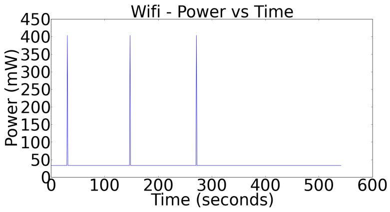
source code, calling the accuracy of the method into question again. The device is in high power state for a longer time when it is active, implying that data transfer consumes more power.

4.1.3 3G

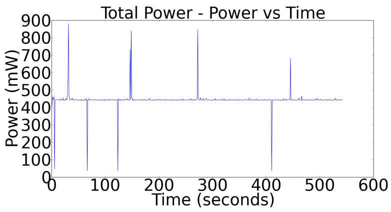
The power profile (3G and Total Power) when only the 3G is turned on is present in Figure 4.4. In the first case, the phone is left as it is without any apps running on it. In the rest of the cases, different activities (surfing, watching a YouTube video and making a Skype call) are carried out and the power is measured. As in the case with Wi-Fi, we see only specific values in all the traces (10 mW, 405 mW and 902 mW) due to them being hard-coded in the PowerTutor source code. The power profile is however slightly different from Wi-Fi - when dropping from a higher to a lower power value, the power first drops to an intermediate value and remains there for a few seconds before dropping further. This is in keeping with the 3G states usually defined for the phones; usually when a phone wants to go from high power data transfer state to a base power state, it first drops to an intermediate power state.

4.1.4 Wi-Fi and 3G

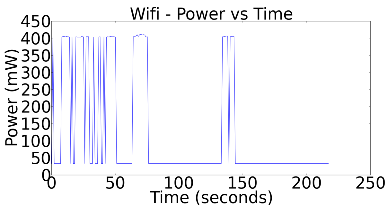
When both Wi-Fi and 3G are present, it is interesting to note that the phone uses the Wi-Fi.



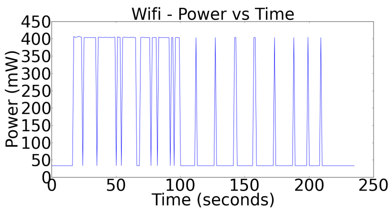
(a) Wi-Fi.



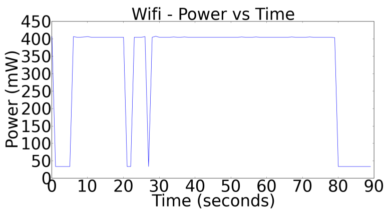
(b) Total Power.



(c) Wi-Fi - Surfing.

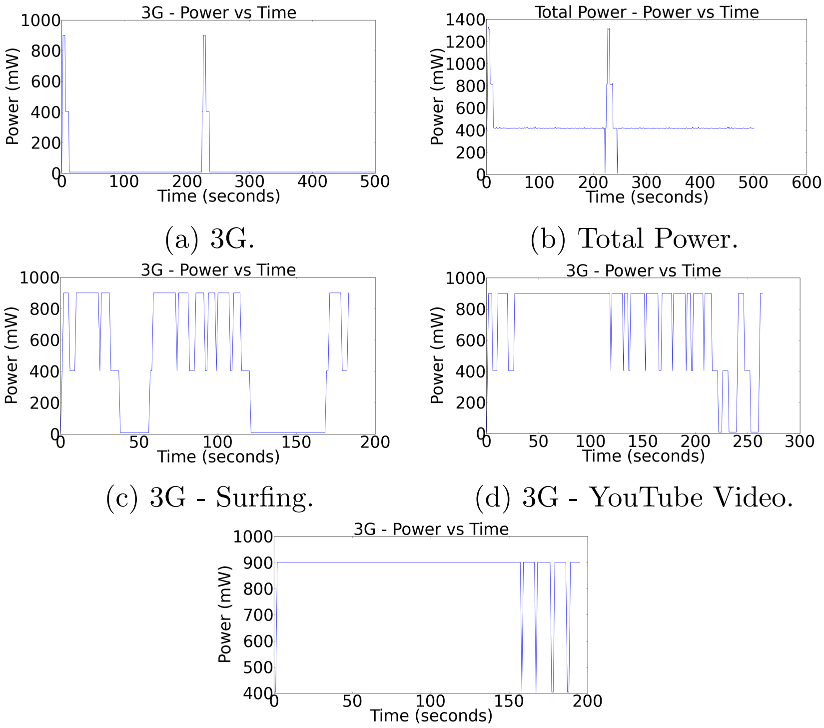


(d) Wi-Fi - YouTube video.



(e) Wi-Fi - Skype Call.

Figure 4.3: PowerTutor: Wi-Fi measurements.



(a) 3G. (b) Total Power.

(c) 3G - Surfing. (d) 3G - YouTube Video.

(e) 3G - Skype Call.

Figure 4.4: PowerTutor: 3G measurements.

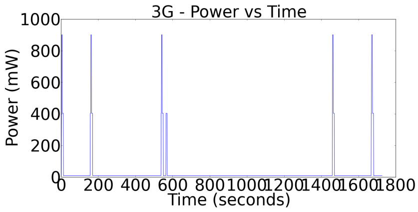
4.1.5 Tethering

When 3G is available and the phone is used as a hotspot, power profile for various cases is shown in Figure 4.5. Tethering consumes the highest power of all the cases we have seen so far.

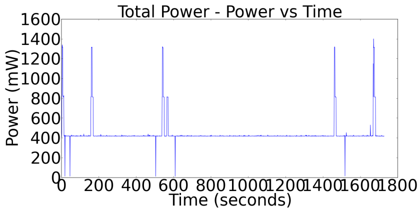
For a station connected to the hotspot, power profile is the same as in the case of Wi-Fi, i.e. 34 mW in low power state and 410 mW in high power state.

4.2 Battery Level Measurement

We were interested in observing how a phone's battery percentage varied with time. In order to do this, a simple Android app was written, which would record the time stamp whenever a phone's battery level changed. This app was loaded onto the phones which were at 100% battery level. The phones were left with the app running till their battery level dropped to 30% and the results were plotted from the log files created by the app. This was

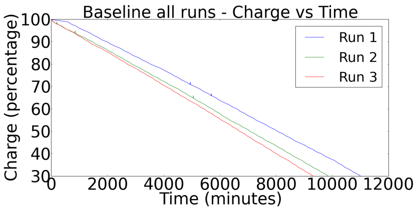


(a) 3G.

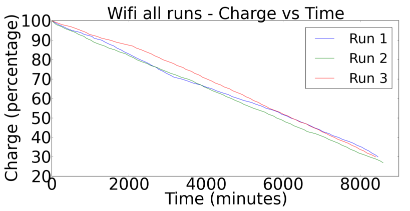


(b) Total Power.

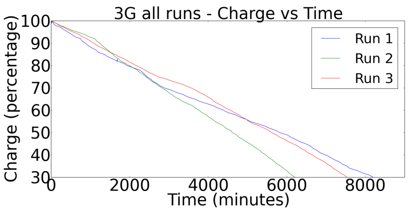
Figure 4.5: PowerTutor: Tethering measurements.



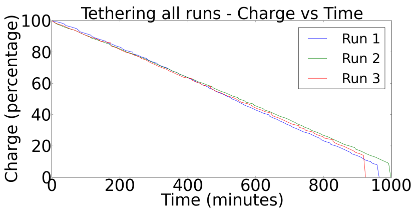
(a) Baseline.



(b) Wi-Fi.



(c) 3G.



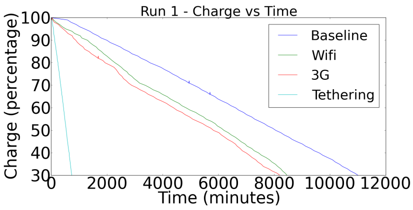
(d) Tethering.

Figure 4.6: Battery App: Individual components - 3 runs.

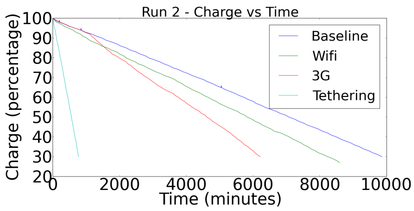
repeated for various cases - with and without Wi-Fi and 3G etc. Three runs were taken for each case. The results are shown in Figure 4.6. For ease of comparison, the level variation for each component is shown side by side for individual runs in Figure 4.7.

We observe that tethering drastically drains the battery as compared to the other cases. There isn't as huge a difference between 3G and Wi-Fi, however Wi-Fi is draining the battery less. If two phones were left with Wi-Fi or 3G continuously switched on, respectively, the phone with the Wi-Fi switched on would last about 10-12 hours more than the 3G phone.

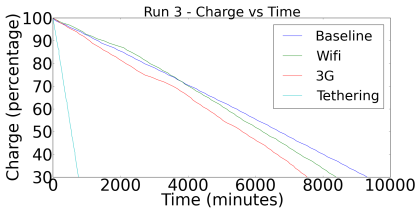
While this method was accurate, it was extremely slow (it took several days for a phone to come down from 100% to 30%). Not only this, it did not provide much useful information about the power consumed by each component.



(a) Run 1.



(b) Run 2.



(c) Run 3.

Figure 4.7: Battery App: 3 Runs.

4.3 Monsoon Power Monitor

The final method that we tried was the Monsoon Power Monitor. The Power Monitor can analyze the power on any device that uses a single lithium (Li) battery. The experimental setup is shown in Figure 4.8.

In the setup, the phone's battery is removed and the phone is connected to the Power Monitor, which supplies a voltage (we use the default of 3.7 V) to it. The software is run and records the current and power values of the phone. We used an Android app that could run different experiments. For each experiment, a test for one of the sensors (Wi-Fi, 3G) could be run and different parameters could be specified - duration of the experiment, frequency of switching the sensor on and off etc. This app was then run on the phone and the Power Meter was used to record the power when the phone was in different states. Some baseline measurements, without using the app, were also taken.

In order to get average power values for each case, the experiments were run with the screen switched off (so that the LCD screen does not contribute to these measurements). There were 3 runs of 5 minutes each for every case. The values obtained are shown in Table 4.1.

Power profiles are shown in Figure 4.9. The baseline power pattern consists of power values being around 10 mW (approximately) with spikes up to 550 mW (approximately) once every second or so. The other profiles for Wi-Fi scanning, Wi-Fi connected to an access point, 3G switched on and tethering mode switched on follow similar patterns but have longer periods

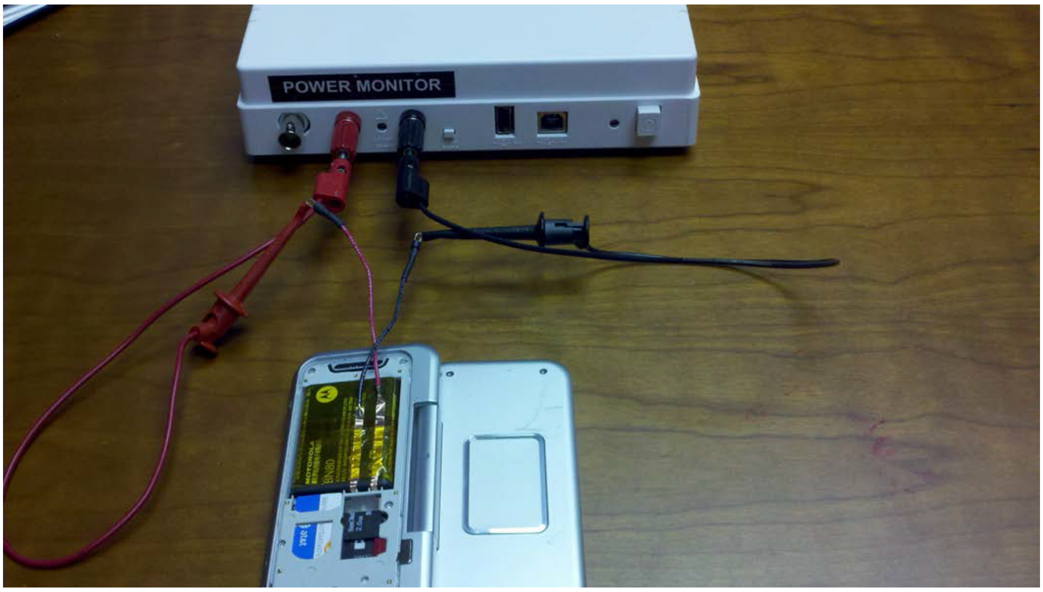
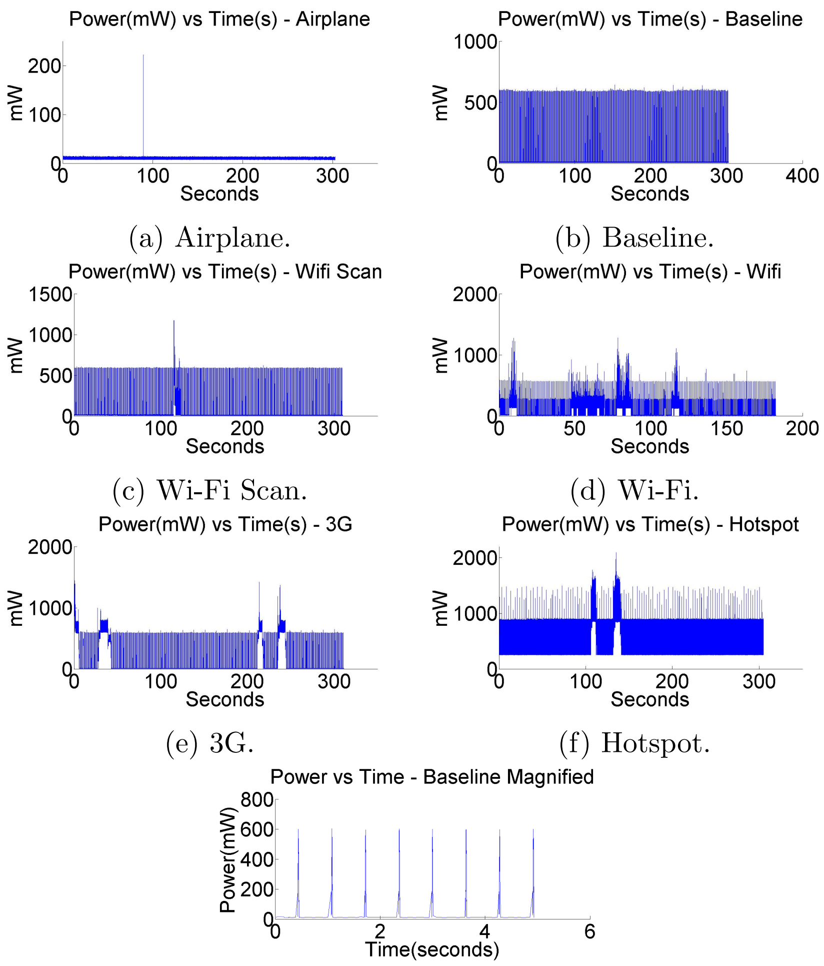


Figure 4.8: Power Monitor setup.

Mode	Mean Power (mW)	Variance
Airplane	10.02	439.76
Baseline (Wi-Fi and 3G off)	21.32	4822.46
Scanning for Wi-Fi	25.32	6237.09
Wi-Fi	77.33	14515.18
3G	188.32	40955.04
Hotspot (no 3G)	284.78	9589.65
Hotspot (with 3G)	297.63	17574.96
Hotspot (with 3G and connections)	357.06	12968.83

Table 4.1: Power measurements in different states.



(a) Airplane.

(b) Baseline.

(c) Wi-Fi Scan.

(d) Wi-Fi.

(e) 3G.

(f) Hotspot.

(g) Baseline magnified.

Figure 4.9: Power Monitor: Power profiles of various states.

in high power states. Since the data points are so closely spaced, a magnified view of the baseline power profile is shown in Figure 4.9 (g). It shows the power values for 5 seconds.

Figure 4.10 given an overview of the distribution of power values in each state - it shows the median, 25th and 75th percentile with the outliers suppressed.

It is seen that there is a difference between the airplane mode and the baseline mode, with the airplane mode lacking the spikes to higher power values. It is, thus, beneficial to switch on airplane mode whenever the phone is not sending data to conserve energy. 3G has a higher average power than Wi-Fi. When the phone is used as a hotspot, there is a large increase in

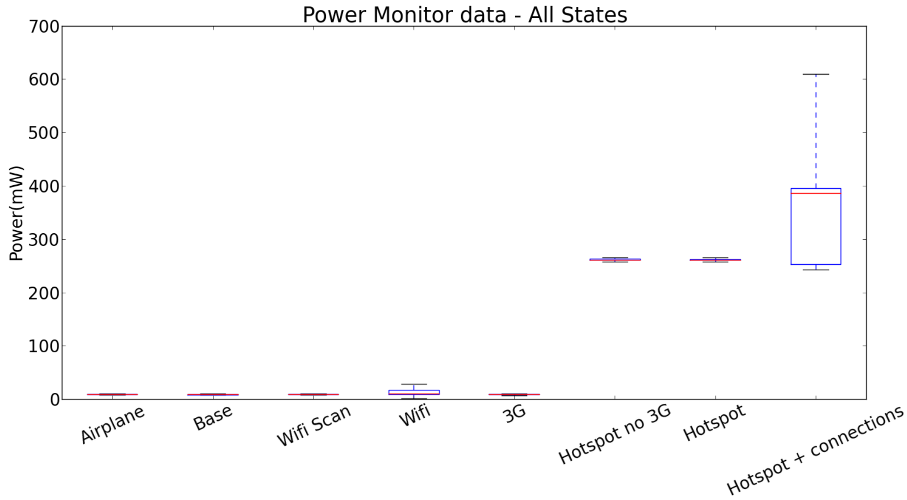


Figure 4.10: Power Monitor: All states - power distribution.

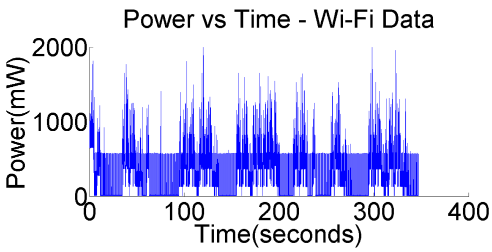
power consumption. Hence, it is advisable to switch an access point role among a group of devices so as to not burden a single device. However, there isn't a huge difference in power consumption when the number of devices connected to the AP increases. Finally, the high variance in the readings is due to the spikes that appear in all the readings.

4.3.1 Sending Data

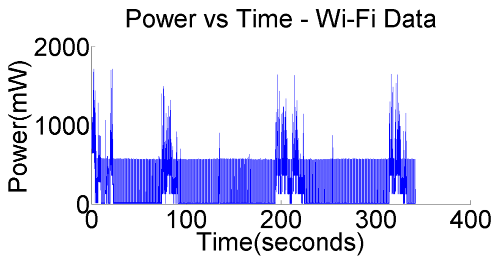
We were interested in seeing how the power consumption was affected by data transfer. We analysed 2 cases - sending data using Wi-Fi and sending data using 3G. For each case, 2 scenarios were considered. In the first scenario, small data packets of 150 bytes each were sent at an interval of 20 seconds. Every 20 seconds, the Wi-Fi/3G would be switched on, the data would be sent and the Wi-Fi/3G would then be switched off. In the second scenario, a larger data packet of 1500 bytes was sent at an interval of 1 minute. Three runs of each of these scenarios were carried out for a total duration of 5 minutes (for Wi-Fi and 3G).

The power profiles for Wi-Fi are shown in Figures 4.11 and 4.12. The power profiles for 3G are shown in Figure 4.13 and 4.14. The average values are given in Table 4.2.

We observe that there is not a huge difference between sending a larger amount of data (1500 bytes) and a smaller amount (150 bytes). These comparable values mean that if we were to transfer 150 bytes at a time, switching on the Wi-Fi/3G, it would take a lot more energy to transfer 1500 bytes (we

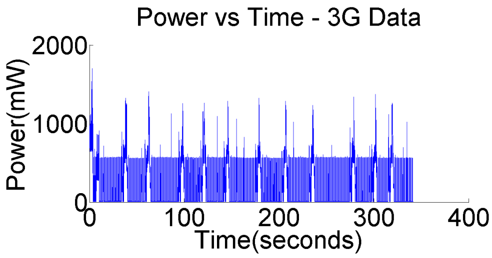


(a) 150 bytes at 20s interval.

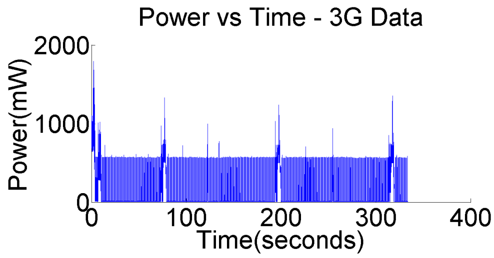


(b) 1500 bytes at 1min interval.

Figure 4.11: Power Monitor: Wi-Fi data



(a) 150 bytes at 20s interval.



(b) 1500 bytes at 1min interval.

Figure 4.12: Power Monitor: 3G data.

have to do this 10 times) as compared to transferring 1500 bytes in one go.This validates the need for a queuing system. What is interesting to note is that Wi-Fi remains in a higher power state for a longer time than 3G in the data measurements. This could possibly be due to Wi-Fi's relatively high ramp energy.

Comparison of measurement methods

Table 4.3 gives a comparison of the different measurement methods that were used in this project. PowerTutor, while convenient to use, has questionable

State	Mean Power (mW)	Variance
Wi-Fi 150b at 20s interval	256.28	68382.25
Wi-Fi 1500b at 1 min interval	282.46	40654.66
3G 150b at 20s interval	457.43	45033.08
3G 1500b at 1 min interval	467.63	26578.78

Table 4.2: Power Monitor:Power values when sending data.

	Power Tutor	Battery Level App	Power Monitor
Results	Power (mW)	Battery level (%)	Power (mW)
Measurement Duration	Instant results	>1 Day	Instant results
Effort	Low	Low	Medium
Accuracy	Low	Medium	High

Table 4.3: Comparison of methods.

accuracy due to use of limited number of phone models in its initial measurements. Battery Level App is a good visualization tool but takes days to log values and has a low time granularity. Monsoon Power Monitor requires effort to gather the measurements but the accuracy is high due to the monitor being directly connected to the phone battery. Hence, in our evaluation section (Chapter 6), we will be making use of the Power Monitor measurements to evaluate our model.

Chapter 5

Queuing Mechanism

From our measurements, we noticed that increasing the data size did not lead to an equally large increase in power consumption. Also, since the device is in a high power state for a long time for every data transmission due to high ramp and tail energies, it is beneficial to transfer data as seldom as possible. Hence, we propose a queue mechanism where data is stored and sent in chunks rather than as soon as it is received. The queue mechanism has been chosen based on the following two concepts:

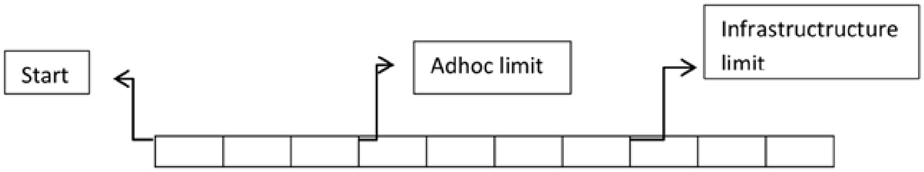
- Buffering and sending more data in bunches to reduce the energy overhead of switching on the network module.
- Usage of limits to allow for faster use of the technology requiring less energy.

An example queue is shown in Figure 5.1. In order to potentially reduce consumption, we augment the queue with two limits - the adhoc limit and the infrastructure limit, with the adhoc limit being less than the infrastructure limit. The phone switches between adhoc and infrastructure modes based on these limits.

Decisions as to which mode to pick based on the queue are taken as follows:

i. If the messages have not yet queued till the adhoc limit, do not send data but continue waiting for and queuing messages till limit is reached.

ii. If the messages have crossed the adhoc limit but have not reached the infra limit yet, go into the Adhoc mode. Wait for Wi-Fi to be available through the creation of a hotspot i.e. become an STA and connect to an AP or become an AP itself (based on the time of scanning). If this is successful, send the data in the queue. If not, go on queuing messages till the infra limit is reached.




```
add_to_queue(element)

if limit >= infra_limit:

      if hotspot found:
            send_Wi-Fi(Qelements)
      else:
            send_3G(Qelements)

elif limit >= adhoc_limit:

      if hotspot found:
            send_Wi-Fi(Qelements)
```

Figure 5.1: Queue and mode decisions.

iii. Once the infra limit has been breached, go into infrastructure mode and send data via 3G. Note that if there happens to be a hotspot present, go into adhoc mode rather than infra mode since energy consumption will be lower. If no network/hotspot is available, keep queuing the messages till an avenue opens up. Then transfer to the relevant mode and send the data.

This system raises two important considerations - message deletion and message prioritization, which will be discussed in the following sections.

5.0.2 Message Deletion

Since there is no guarantee that there will always be some mode of transmitting messages at all time, there ought to be a method for deletion of messages. This helps maintain the size of the queue and removes messages that are no longer relevant. A possible option would be to have a TTL - time to live. Messages entering the queue are given a TTL value and a timer is used to reduce the TTL value. When the TTL reaches 0, the messages is removed from the queue.

5.0.3 Message Prioritization

Different messages have different priorities and we need to ensure that important messages are not deleted. We need to assign TTL values based on message priority.

In the disaster team scenario, for example, we can assume that messages for the headquarters are of higher priority than intra-team messages. Hence, these messages would be given a higher TTL value. We could also have two queues - one for intra-team and one for team-headquarters communication, with the headquarters queue having lower limits (so that a network is searched for more aggressively and messages are sent more frequently).

An alternative method of tackling this issue would be to implement multiple queues based on urgency - low, medium and high. The urgency can be manually determined by the owner of the device. This method, while requiring manual intervention, would be more accurate in determining priority.

Chapter 6

Evaluation

Now that we have defined a state model and obtained energy readings for all the states, we apply our findings to a sample search and rescue scenario to evaluate the solution.

6.1 Scenario

We assume a rescue setup where there is a headquarter (HQ) and field search and rescue teams that venture into the disaster area in search of victims. These teams will have to communicate with the HQ from time to time. Assume such a field team of 3 nodes that have to communicate with the HQ, as shown in Fig 5.1. We want to test our model where a device can switch between different modes and when in the Ad-hoc mode, it can switch between being an access point (AP) or a station (STA). Let the time period of switching between AP and STA states be 1 minute. We then form a simple traffic model - assuming that at every 10 seconds, a 150 bytes of data are generated by each STA.

Figure 6.1: Topology for evaluation scenario.

6.2 Cases

There are three cases to be evaluated:

1. No Queuing: There are no queues on the devices and data is sent immediately without getting buffered.

2. Queuing at nodes: There are queues at all nodes. Data is sent according to the model described in Chapter 3.

3. HQ Queue at AP: Each node has 2 queues, one for intra team communication and another one for messages to the headquarters. Intra team communication is similar to Case 2. For headquarter communication, the nodes pass the messages to the AP queue and the AP passes this on to the HQ.

Case 1 is a base case to compare against our model and see if we get any improvements. Case 2 buffers the data and sends it according to the queuing model described in Section 6. It is to test whether our model leads to energy savings. Case 3 is a different method of queuing and sending data where the queuing is centralised. We have this extra case to determine if using only the AP as a point of contact with the HQ would lead to energy savings.

We use the following symbols in calculations:

E_{total} = Total energy required
E_{high} = Energy in high power state (data transfer and tail energy)
E_{base} = Energy in base state (low power).

The total energy is the sum of energy in high power and low power states.

$$E_{total} = E_{high} + E_{base}$$

where Energy = average power $\times$ time in a state.

The power values used in the evaluation are obtained from the Monsoon power monitor. They are outlined in Table 5.1.

Approximate values of the energy consumed in each case are calculated in the following sections.

6.2.1 No Queuing

In Case 1, there is no queuing system and every node sends data as soon as it is generated. The topology for this case is shown in Figure 6.2.

State	Power Assumed (mW)
Sending 150 bytes of data by 3G	457.43
Sending 150 bytes of data by Wi-Fi	256.28
Sending more than 150 bytes of data by 3G	467.63
Sending more than 150 bytes of data by Wi-Fi	282.46
Power in base state	21.32

Table 6.1: Assumed power values for evaluation.

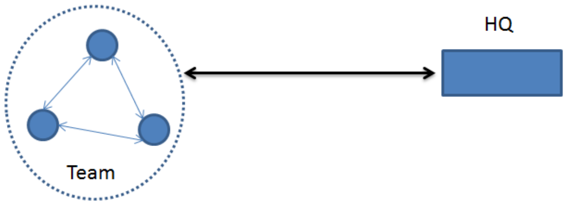


Figure 6.2: Case 1: No queuing.

In one minute, data is generated 5 times - 3G is switched on, the data is sent out immediately and 3G is switched off. However, the device remains in a higher power state for about 5 seconds after 3G is switched off before going to the lower baseline value. The state time line of a node is shown in Figure 6.3.

Using the average power values from our measurements, we can approximate the energy used in this scenario as:

$$E_{total} = (457.43 \times 25) + (21.32 \times 35) = 12.1J.$$

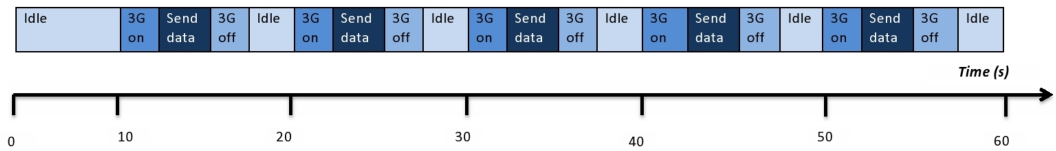


Figure 6.3: Case 1: States of a node.

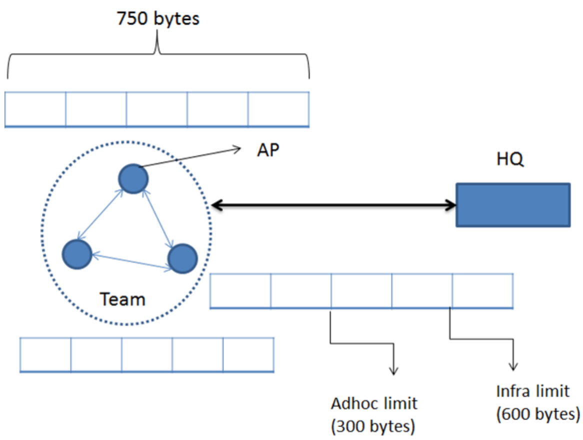


Figure 6.4: Case 2: Queuing at nodes.

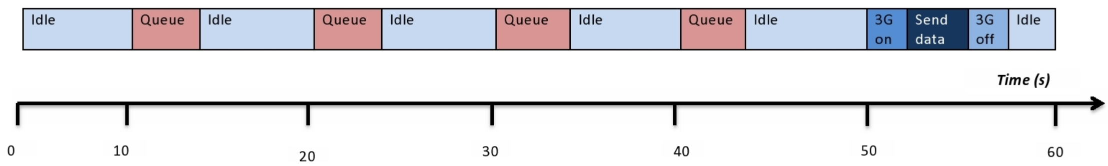


Figure 6.5: Case 2a: Node state timeline.

6.2.2 Queuing at Nodes

The topology for Case 2 is shown in Figure 6.5. In this case, we follow the queuing scheme described in the previous chapter.

Each node has a queue of size 750 bytes with adhoc limit set at 300 bytes and infrastructure limit set at 600 bytes.

In the case where the queue is filled and the node wants to send data to the HQ, in one minute, the node sends data only once. Hence, it is in a high power state for only about 5 seconds and in base state otherwise. The state time line is shown in Figure 6.6

$$E_{total} = (467.43 \times 5) + (21.32 \times 55) = 3.5J.$$

In the case where the adhoc limit has been reached but not the infrastructure limit and the node wants to send to the HQ, the node sends using Wi-Fi (assuming an AP is available) and remains in high power state for approximately 15 seconds per data transmission. It is in the low power state

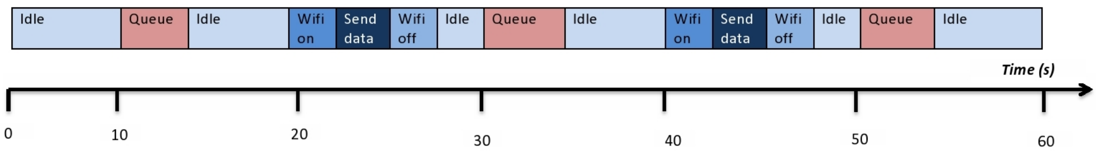


Figure 6.6: Case 2b: Node state timeline.

for the rest of the time. The state timeline is shown in Figure 6.7, where we see that adhoc limit is reached twice in a minute.

$$E_{total} = (282.46 \times 30) + (21.32 \times 30) = 9.1J.$$

We notice that the energy used up is more in the second case. This could be due to the device being in a high power state for a longer time for Wi-Fi as compared to 3G. Another possible cause is that data is sent too frequently, resulting in the device being in a higher power state several times. This indicates the need for future work to be conducted in determining an optimum size for the queue and the adhoc and infrastructure limits. However, we can conclude that using queues does lead to lower energy consumption, on comparison with Case 1. Note that in this method, at the end of 1 minute, 150 bytes are still left in the queue to be transmitted in the next round.

6.2.3 HQ Queue at AP

We tweak our model to determine if letting one node perform the team-headquarters communication results in energy savings. Figure 5.5 shows the new scenario. While intra team communication remains unchanged, the responsibility of transmitting data of all the team members falls on the node that acts as an AP at that instant. This means that the AP has a master queue; when a node wants to send data to the headquarters, it sends it to the AP and it is the AP that transmits the data to the headquarters via 3G.

Assume the size of the master queue to be 1950 bytes. Also assuming that the adhoc limit has been reached at each node and Wi-Fi is available, nodes sent to the AP after 20 seconds. At the end of 20 seconds, the AP has received 2 messages each from the other 2 nodes (600 bytes). At the end of the minute, the master queue is filled and the AP sends the data to the headquarters. The state diagram and increasing queue size of the master queue is shown in Figure 6.8.

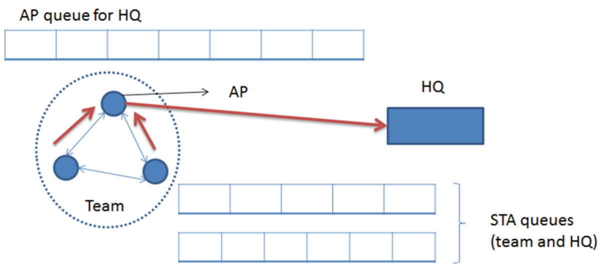


Figure 6.7: Case 3: HQ Queue at AP.

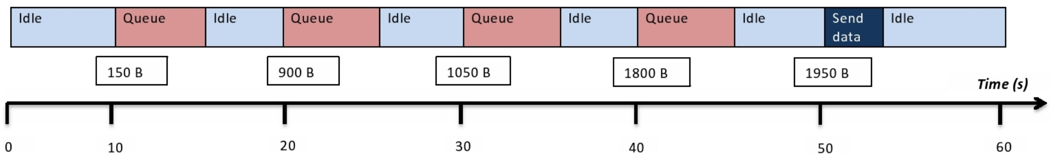


Figure 6.8: Case 3: Master queue state timeline.

For each node,

$E_{totalnode} = (282.46 \times 20) + (21.32 \times 40) = 9.1J.$

For the AP,
$E_{totalAP} = (467.63 \times 5) + (357.06 \times 55) = 21.9J.$

After 1 minute, the AP has to send this data via 3G to the headquarters.
We immediately see that the energy consumed will be more in this case (a combination of node sending to AP and AP sending to HQ). Hence, having a master queue is not beneficial when it comes to saving energy. Not only this, it complicates the system since the AP is not a fixed entity but moves from node to node. Handling the contents that remain in the queue while the nodes switch the role of AP has to be addressed. However, a master queue can have other advantages - having a centralized entity that collects all the messages from the team could be useful in discarding duplicate messages and prioritizing them.

	Case 1	Case 2	Case 3
Energy per minute (all devices)	36.3 J	10.5 J (infra)	40.5 J
Delay	Immediate sending	Depends on queue limits	Depends on queue limits (node and AP)
Bytes received at HQ after 1 minute	2250	1800	1950
Advantages	Simple, No delay	Energy efficient	Message processing at AP possible
Disadvantages	High energy consumption	More complex than case 1	High energy consumption, high complexity

Table 6.2: Summary of evaluation scenarios.

6.3 Summary

We used our measurement values to address three simple scenarios and evaluate our proposed model. We found that having no queuing mechanism uses up to as much as three times the energy as a queue. Queuing messages, while complex, does lead to energy savings. However, determination of sizes and limits based on the scenario is of utmost importance in order to achieve full efficiency. Having a centralized master queue scheme consumes more energy, even more than having no queues because there are two transfers taking place - from the station to the AP and the AP to the HQ. It also introduces more design issues. However, it could potentially be used to send messages in a more efficient manner and save energy by reducing the number of times data is sent (for example, by letting the AP delete duplicate messages and process data before sending). Table 6.2 outlines the three cases and their results.

Chapter 7

Conclusion

In this project, we tried to find a solution to minimize energy consumption of devices belonging to a group of devices. We first developed state models for the devices. Measurements were taken using three different methods to obtain power consumption values for the various stats of the device. We tried three different measurement methods and discussed their benefits and shortcomings. On analysis of these measurements, we came up with a queuing scheme that could lead to energy saving. Finally, we used these measurements and queuing mechanism to evaluate three scenarios. We concluded that usage of the queue mechanism was indeed beneficial in terms of energy savings.

7.1 Challenges

There were several challenges in this project:

a. Measurements took a large chunk of the time. Due to the limitations of the Power Tutor, we had to redo the measurements using the Power Monitor. Due to some technical issues in running the experiments, this took longer than expected.

b. Development of an ideal queuing model was quite difficult because of having to tackle the intricacies that come with it. The current model can be improved in several ways; these are described in the next section.

7.2 Improvements

There are a few avenues that could be explored further in this project:

a. Investigation to determine optimal queue sizes and limits for the queues.

b. Evaluation of another scenario where local information spreading without the use of 3G is considered.

c. Further tests with more sophisticated traffic models and a larger number of devices.

d. Consideration of more modes of WLAN-Opp. Evaluation with the time-based role-switching explained in WLAN-Opp.

Bibliography

[1] N.Aschenbruck, M.Frank, P. Martini, J.Tlle: *Human mobility in MANET disaster area simulation - a realistic approach*, 29th Annual IEEE International Conference on Local Computer Networks (LCN04)(2004)

[2] N.Balasubramanian, A.Balasubramanian and A.Venkataramani: *Energy consumption in mobile phones: a measurement study and implications for network applications*, Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference, IMC '09 (2009)

[3] A. Carroll and G. Heiser: *An Analysis of Power Consumption in a Smartphone*, 2010 USENIX Annual Technical Conference (2010)

[4] R.Friedman, A.Kogan and Y.Krivolapov: *On Power and Throughput Tradeoffs of Wi-Fi and Bluetooth in Smartphones*, INFOCOM, 2011 Proceedings IEEE (2011)

[5] R.Balani: *Energy Consumption Analysis for Bluetooth, Wi-Fi and Cellular Networks*, NESL Technical Report #: TR-UCLA-NESL-200712-01 (2007)

[6] M. Conti and M. Kumar., *Opportunities in Opportunistic Computing.* Computer, 43(1):4250 (2010)

[7] S.Trifunovic, B.Distl, D.Schatzmann, F.Legendre: *Wi-Fi-Opp: ad-hocless opportunistic networking*, CHANTS '11 Proceedings of the 6th ACM workshop on Challenged networks, Pages 37-42 (2011)

[8] L.Zhang, B.Tiwana, R.P.Dick, Z.Qian, Z.M.Mao, Z.Wang, L.Yang *Accurate Online Power Estimation and Automatic Battery Behavior Based Power Model Generation for Smartphones*, Hardware/Software Codesign and System Synthesis (CODES+ISSS) (2010)