

Delft University of Technology
Master's Thesis in Embedded Systems

Energy-Efficient Electricity-Meter Monitoring

Menno van der Reek



Energy-Efficient Electricity-Meter Monitoring

Master's Thesis in Embedded Systems

Embedded Software Section
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology
Mekelweg 4, 2628 CD Delft, The Netherlands

Menno van der Reek
m.vanderreek@student.tudelft.nl

14th June 2017

Author

Menno van der Reek (m.vanderreek@student.tudelft.nl)

Title

Energy-Efficient Electricity-Meter Monitoring

MSc presentation

29th June 2017

Graduation Committee

prof. dr. K.G. Langendoen (chair)	Delft University of Technology
dr. ir. R. Heusdens	Delft University of Technology
dr. ir. C.H. Taal	Quby B.V.
ing. W. van Bakel	Quby B.V.

Abstract

With the arrival of smart electricity meters energy consumption can be monitored continuously and displayed on external devices such as phones or tablets. As a consequence, users become more aware of their energy usage which may result in a reduction in energy consumption. Nevertheless, many countries in Europe still use an analog electricity meter based on a rotating disk, which rotates at a speed proportional to the consumption passing through the meter. In contrast to a smart meter it is not possible to read out the consumption directly. As an alternative, the rotating disk can be observed such that the power (W) and Energy (kWh) consumption can be derived. The current sensor device developed by Quby, uses an LED and phototransistor, where the LED emits light on the disk that is reflected towards the phototransistor. Because of the physical properties of the disk, the sampled phototransistor signal can be represented as a pulse signal, where each pulse indicates a revolution of the disk. The sensor device is mains powered, however, many electricity meters are not located near a power outlet requiring the sensor device to be battery powered. This is a challenging problem since the LED has a relatively large power consumption. The pulse-detection algorithm running on the mains-powered sensor device assumes an LED current of 10 mA with a sampling frequency of 10 kHz. Quby requires that the battery-powered device will last at least one year on an energy budget of 4200 mAh (roughly four AA batteries), and the percentage error of the determined energy consumption should be less than 5%. This implies a factor 50 in power reduction. In this thesis we propose an energy-efficient noise-robust pulse-detection algorithm to detect pulses while keeping the LED current to a maximum of 1 mA. To preserve more energy, the LED is duty cycled to at most 20% instead of 100%, and the sampling frequency is reduced to a maximum of 400 Hz. The proposed method is based on a statistical model where pulse detection is used by means of a multiple-sample likelihood ratio test. Due to the low LED current the signal statistics, such as pulse amplitude, offset and noise, are very sensitive to ambient light. Therefore, an additional method is proposed to estimate these statistics continuously. As a consequence, the detection thresholds in the likelihood ratio test are dynamically adjusted based on predefined probabilities of a false alarm and true detection. The proposed algorithm is extensively tested in a lab setup with three different analog electricity meters, a varying load and a light source emitting light in the same relative spectral-power distribution as sunlight. Moreover, measurements are performed at two households for one week each. From the experiments it can be concluded that the proposed method can last for at least one year when battery-powered, while predicting energy consumptions with an error of less than 2%.

Preface

This Master thesis is the final part of my Master of Science in Embedded Systems at Delft University of Technology. The work presented is performed at a company called Quby, which is part of the utility company Eneco. In this thesis, a noise-robust pulse-detection algorithm is proposed used to monitor analog electricity meters.

I want to express my sincere gratitude to my company supervisors Cees Taal and Wouter van Bakel for guiding me through the journey of this graduation project. They always made time available for me, something I really appreciate. I would also like to thank my university supervisor Koen Langendoen. Further, special thanks to grandpa Wim and grandpa George for writing down every four hours the meter reading for a period of one week. Without you, I would not have been able to evaluate my work in practice. Finally, many thanks to my family and friends for supporting me, distracting me when necessary, and caring for me.

Menno van der Reek

Delft, The Netherlands
14th June 2017

Contents

Preface	v
1 Introduction	1
1.1 Electricity meters	2
1.1.1 Smart meters	2
1.1.2 Digital meters	2
1.1.3 Analog meters	3
1.2 Sensors	4
1.3 Problem statement	5
1.4 Methodology	6
1.5 Contributions	6
1.6 Thesis outline	7
2 Background	9
2.1 Meter module architecture	9
2.2 Photo-reflector signal properties	10
2.3 Related work	13
2.3.1 Likelihood ratio test	14
2.3.2 Multiple-sample likelihood ratio test	17
2.3.3 Matched filter	18
3 Pulse Detection Algorithm	19
3.1 Assumed statistical model	20
3.2 Estimate signal statistics	20
3.2.1 Estimate μ_0	23
3.2.2 Estimate μ_1	23
3.2.3 Estimate σ_0	24
3.2.4 Estimate σ_1	24
3.3 Likelihood ratio test	24
3.4 Pulse to energy consumption	29
3.5 Sampling frequency and duty cycle	29

4	Evaluation	33
4.1	Lab experiments	34
4.1.1	Test setup	34
4.1.2	Switching power consumption	35
4.1.3	Ambient light influences	37
4.2	Household experiments	39
4.2.1	Test setup	39
4.2.2	Evaluation	40
5	Conclusion and Future Work	43
5.1	Conclusions	43
5.2	Future work	44
A	Different C-values	49

Chapter 1

Introduction

Many households have their own electricity meter, which keeps track of the energy consumption. Nowadays, not only the energy suppliers are interested in these numbers - to bill their customers - but also the customers themselves. With the introduction of the smart meter, the amount of electricity consumed can be communicated in real-time to the customer. This consumption is presented on a device, e.g., a smartphone or tablet, which gives more insight to the customers and motivates them to reduce their energy consumption up to 9% [1]. In 2009, the European Union set a goal to replace 80% of all electricity meters with smart meters by 2020. A commission report from July 2014 by the Joint Research Centre (JRC) estimated 72% coverage for smart electricity meters by 2020 based on the current national roll-out plans [2]. This coverage is likely to be even lower, since Germany decided to selectively roll-out the smart meters (23% at 2020) [3] instead of a wide roll-out (>80%), and the discovery of multiple malfunctions in smart meters [4][5]. For example, the Netherlands still only has a coverage of smart meters of 26% in Q4 2016 [6][7], and the UK only has 10% covered [8].

Because of the slow transition from non-smart meters to smart meters, and the interest from people to get more insight into their energy consumption, there exists a growing market for non-smart meters extended with a device that transfers information about the energy consumption to the customer. The Dutch company Quby, developer of the smart thermostat called Toon, converts these non-smart meters to semi-smart meters with their product called *meter module*. The meter module is attached to the electricity meter and keeps track of the meter readings, which are wirelessly transferred to the Toon display located in the living room. Because the Toon display is placed in the central point of a household this is the ideal location to show statistics about the energy consumption to the customer. Currently, Quby wants to expand their market reach to more European countries. This in-

introduces difficulties where electricity meters are not placed nearby a power outlet to power the meter module. Internal research at Quby shows that meters are often located in basements or even outside homes. This thesis will discuss a method to obtain meter readings, known as automatic meter readings (AMR), from these electricity meters.

This chapter will describe the following: Section 1.1 explains the different electricity meters on the market. Section 1.2 clarifies the chosen sensor to keep track of the indicated energy consumption. Section 1.3 gives the problem statement. Section 1.4 discusses the methodology. Section 1.5 shows the main contributions, and Section 1.6 describes the remaining organization of this thesis.

1.1 Electricity meters

There are three types of electricity meters:

1. Smart meters
2. Digital meters
3. Analog meters

In this report, the electricity meters will be discussed in order of relevance for this work starting with the least relevant one.

1.1.1 Smart meters

Smart meters are the most recently developed type of meters. They measure the energy consumption with a digital circuit and periodically communicate this information to the grid operator. The grid operator passes this information to the customer's corresponding utility company. Next to the data presented by the utility companies, customers also have the possibility to read out the smart meter themselves. The smart meter is equipped with a serial port, which sends a telegram message with the current meter reading every 10 seconds [9]. Some smart meters provide extra information by interfacing with the gas, thermal (heat and cold) and water meter [9].

1.1.2 Digital meters

Digital meters also measure the energy consumption, but they do not communicate with the grid operators nor do they have the ability for customers to plug in a cable. The interface of a digital meter consists of a small digital display showing the meter readings and a blinking LED, see Figure 1.1. Every n -th on-off transition of the LED means exactly 1 kWh is consumed,



Figure 1.1: Digital meter where the arrow indicates the blinking LED.

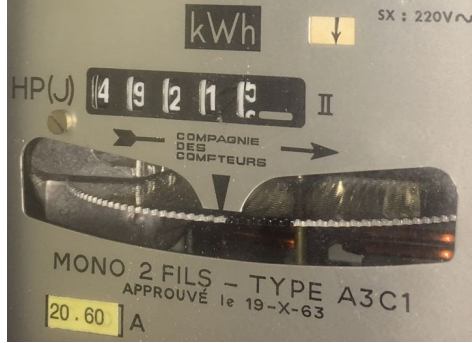


Figure 1.2: Disk of an analog meter with a serrated edge and a black stripe.

where n is a constant value indicated by the meter as impulses/kWh. So monitoring the blinking LED gives the ability to derive the energy consumption over time. The impulses/kWh value differs for each brand and is often referred to as the constant value or C-value. The higher this C-value, the higher the sampling frequency of a sensor needs to be to capture the on-off transition of the LED. By inspecting hundreds of pictures of digital electricity meters for different countries in Europe, it was concluded that the smallest encountered C-value is 600 and the largest is 10.000 impulses/kWh, see Appendix A.

1.1.3 Analog meters

Analog meters are mechanical meters containing a metal spinning disk and an analog counter, which indicate the total energy consumed. When current flows through this meter an electric field is generated that drives the disk to spin. So the angular velocity of the disk is linearly related to the consumed power. There are two types of metal disks. The first type has a serrated edge, while the other has a smooth edge. In addition, each disk contains a red or black stripe covering around 5% of the surface of the disk. Figure 1.2 shows an example of a serrated disk of an analog meter with a black stripe. Each analog meter has a C-value indicating the number of revolutions the disk needs to make before exactly 1 kWh is consumed. This constant value is often shown on the meter as C or rev/kWh (revolutions/kWh). Tracking the number of revolutions of the disk can be used to derive the energy consumption over time, similar to the blinking LED of the digital meter. One revolution can be recognized by observing the red or black stripe. The smallest encountered C-value for analog meters is 50 and the largest is 800, see Appendix A.

This thesis will mainly focus on analog meters since those are the most challenging meters to track. However, the developed method will also work for digital meters. Smart meters will not be discussed since they do not require an external sensor to track the meter readings.

1.2 Sensors

Several types of sensors can track the meter readings. Quby requires the purchase cost of the sensor to be €2 or lower, support as many meters as possible, and be easy to install by the customers themselves. They examined the following types of sensors for their meter module:

- Current clamp,
- Camera,
- Photo-reflector.

Current clamp

A current clamp is a coil, clamped around an electrical conductor. When current passes through the conductor, an electrical field is generated. The coil will measure this electrical field and translate it to a current flow. Cheap current clamps can only measure current in a single core cable, requiring multi-core cables to be split before a core can be measured. To measure a household's energy consumption a current clamp needs to be attached to the power cable entering the electricity meter. Since this cable consists of three cores, it needs to be split. Splitting the entering power cable is only allowed by a certified company, making this method not feasible since customers cannot install this sensor themselves.

Camera

A camera could be placed in front of the meter taking pictures periodically. The meter readings can be extracted from these pictures with image processing techniques. These image processing techniques could be performed on a server [10] or locally on an embedded device connected to the camera. The main advantage of using a camera is that it can be applied to almost every meter. Nevertheless, this sensor is not selected to track the meter readings, because of the relatively high price and computational power required on an embedded device.

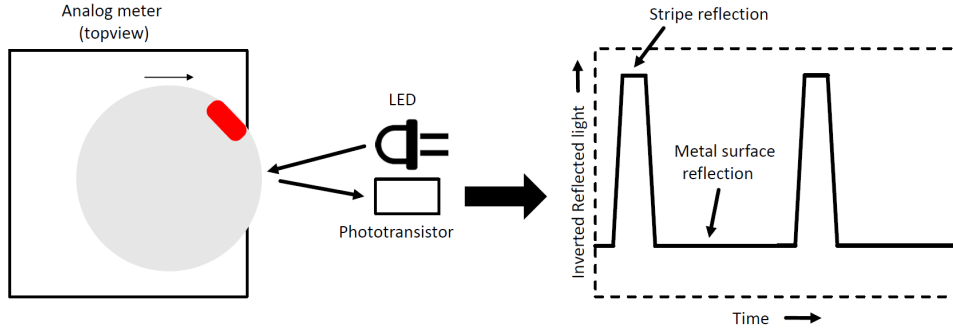


Figure 1.3: Photo-reflector measurements method. Light from the LED is reflected by the disk and sampled by the phototransistor. The sampled signal can be represented as a pulse signal where each pulse corresponds to a revolution of the disk.

Photo-reflector

The photo-reflector consist of a LED and phototransistor in one case. This sensor is positioned in front of the rotating disk, where the LED emits light which is then reflected by the disk and measured by the phototransistor. Each time the stripe passes the sensor, the measured reflection drops because light is absorbed by the stripe. Because of the hardware design of the photo-reflector, the output data is inverted, meaning a drop in reflection translates to a pulse in the recorded signal. See Figure 1.3 for a schematic overview of the photo-reflector measurement method with the inverted y -axis. Note that the photo-reflector can also be used to detect the blinking LED of a digital meter by only using the phototransistor. The low purchase cost of €1 and support for a wide range of meters made Quby choose this sensor for their meter module.

1.3 Problem statement

The meter module needs to be battery powered in order to track meter reading of electricity meters far from a power outlet. Quby wants the meter module to last at least one year on an energy budget of 4200 mAh, which corresponds to four AA batteries. Energy efficiency was not a main priority in the hard- and software design of the mains-powered meter module. In this meter module, the LED is set to a constant current of 10 mA and the sampling frequency of the phototransistor is at 10 kHz. In fact, the estimated battery lifetime will be around seven days if the mains-powered meter module without any hardware or software adjustments would be used. This means a battery-powered meter module needs to be a factor 50 more power efficient than the current mains-powered meter module. Another

requirement is that the error in measured energy consumption should be less than 5%. To limit the scope, it is assumed that the current hardware design of the photo-reflector sensor remains unaltered. This brings us to the following research question:

Is it possible to monitor analog electricity meter readings with the help of a photo-reflector sensor for at least one year with an energy budget of 4200 mAh and an error in the measured energy consumption less than 5%?

While answering this research question, significant variations in ambient light caused by sunlight needs to be taken into account, since some electricity meters are placed in outside environments.

1.4 Methodology

As will be shown in Section 2.1, the energy consumption of the photo-reflector needs to be reduced in order to meet the lifetime of the battery-powered meter module with the given energy budget. To be energy efficient, the sampling frequency and LED current need to be reduced. Reducing the LED current will result in a lower Signal to noise ratio (SNR) of the sampled photo-reflector signal making it difficult to detect pulses. Therefore, a literature study is conducted on topics that discuss how to detect pulses in noise.

To reduce the sampling frequency, prior knowledge of the household's maximum power consumption is required together with the C-value of the electricity meter. This means the minimum sampling frequency is different for each household. Finally, the LED can be duty cycled to preserve even more energy.

1.5 Contributions

This thesis presents an energy-efficient algorithm to keep track of electricity meter readings. During this research the following contributions were made:

- A noise-robust pulse detection algorithm is proposed that can detect different pulses varying in pulse frequency, offset and amplitude.
- In contrast to common pulse detection methods, we do not assume stationarity. Instead, the algorithm continuously estimates signal statistics.
- During operation time the algorithm is able to express its performance using a probability model. This performance value can be used to

control the LED current to a level such that optimal energy efficiency is reached.

- A model of the signal that is recorded by the photo-reflector is developed in python. Using this model a wide range of power consumption scenarios can be simulated and used to validate the algorithm.
- Valuable data is recorded from the photo-reflector sensor. Quby can run their future algorithms against this database. Note that the gathered data is not only data generated in the lab, but also real life data recorded at people's homes.

1.6 Thesis outline

The upcoming chapter will present the current architecture of the meter module. Chapter 2 will also discuss the related work corresponding to energy-efficient pulse detection. Chapter 3 explains the designed algorithm in three main blocks. Then, Chapter 4 evaluates the algorithm with the help of the data gathered from the lab and in actual homes. Finally, the conclusion and possible future work will be discussed in Chapter 5.

Chapter 2

Background

In this chapter, first the hardware architecture of the meter module will be discussed. Then a deeper understanding of the origin of the signal varieties sampled by the photo-reflector will be explained. Thereafter, related work is discussed.

2.1 Meter module architecture

The battery-powered meter module consists of a microcontroller, photo-reflector and wireless module, see Figure 2.1. The microcontroller samples the photo-reflector with an internal 10-bit ADC. Subsequently, a signal-processing algorithm is applied that detects the pulses and translates these to the energy consumption. Every 10 seconds the wireless module sends the updated energy consumption to the display, located in the living room.

The battery-powered meter module will consist of a new power-efficient microcontroller and wireless module, in comparison to the mains-powered meter module. The photo-reflector sensor is considered to remain the same.

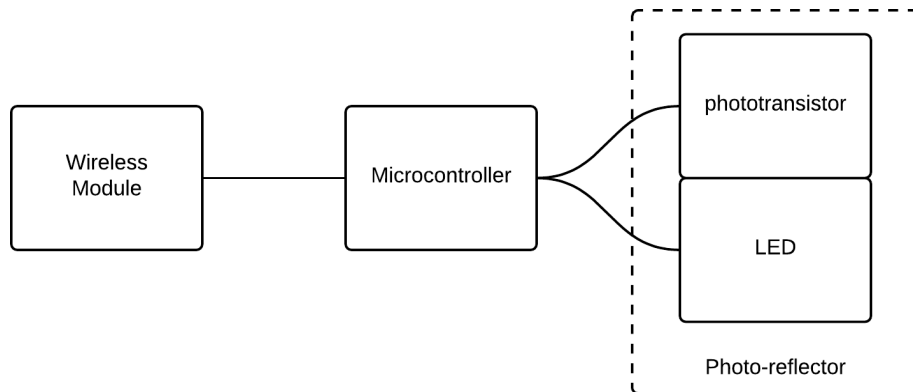


Figure 2.1: Meter module architecture.

Table 2.1: Current consumption of the battery-powered meter module components.

Module	Average current draw (mA)	Measurement method
Microcontroller	0.21	Datasheet
Wireless module	0.06	Datasheet
Photo-reflector	10	Current meter

Because the battery-powered meter module is still under development, the current consumption of the microcontroller and wireless module are estimated with the help of the datasheets. To derive the power consumption of the photo-reflector, measurements are performed with the mains-powered meter module. A current meter is placed between the photo-reflector and microcontroller, while running the algorithm developed by Quby. This algorithm samples the photo-reflector at 10 kHz and tunes the LED current to a level such that the SNR is maximized within the dynamic range of the digitized signal. After several experiments with different levels of ambient light, the LED current tends to converge to around 10 ± 1 mA.

As shown in Table 2.1 the photo-reflector is responsible for a significant part of the total energy consumption. Based on this finding the main focus in this thesis will be on reducing the current of the LED. The small value for the wireless module is because only once every 10 seconds an update is sent to the Toon display. Thus, most of the time the wireless module is in deep sleep mode. Given the requirement that the meter module should last at least for one year, the following equation needs to be met:

$$\frac{Q_{\text{battery}}}{I_{\text{microcontroller}} + I_{\text{wireless}} + I_{\text{photo-reflector}}} > 8760 \text{ hours}, \quad (2.1)$$

where Q is the capacity of the battery and I the current draw of the components. Assuming an energy budget of 4200 mAh, the photo-reflector current needs be reduced to 0.2 mA. This means that an improvement in energy consumption of at least a factor 50 is required.

2.2 Photo-reflector signal properties

The sampled photo-reflector signal can be described with the following signal features:

- Offset,
- Pulse-peak,
- Noise,
- Pulse frequency,
- Slope of the pulse.

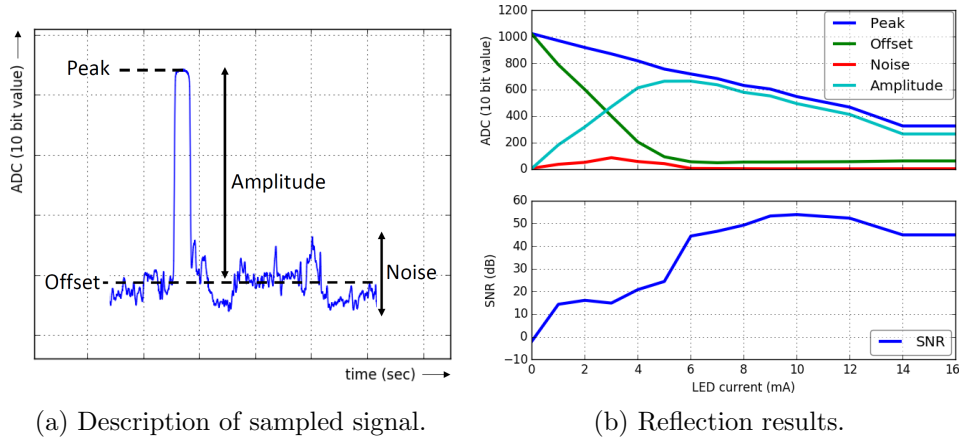


Figure 2.2: Influence of reflection for different LED current levels.

The pulse frequency and slope are related to the angular velocity of the disk of the analog meter. When more power is consumed, the disk will spin faster resulting in a higher slope of the pulse and pulse frequency. Figure 2.2a shows one pulse period with the offset, pulse-peak and the pulse amplitude. A pulse will be present in the sampled signal, if the red or black stripe of the disk passes the photo-reflector. The peak and offset value indicate the average reflected light related to the stripe and metal part of the disk, respectively. As a consequence, the amplitude is a direct result of the difference in reflected light between the two different disk surfaces. The observed noise consists of the same repetitive pattern for each revolution, which is a result of reflected irregularities on the metal part of the disk.

To analyze the influences of different LED currents relative to the sampled reflection data of an analog meter disk, the following experiment is performed: The reflection of the disk is sampled for different LED currents while the disk spins at a fixed angular velocity by applying a constant load of 2 kW. The LED current is varied between 0 and 16 mA with a step size of 1 mA. To exclude external light sources from influencing the experiment, all measurements are performed in a completely dark environment. Figure 2.2b shows the results of the experiment with respect to the described signal features. Since the slope and pulse frequency are not related to the reflected light but to the rotation speed of the disk, they are not included in the results. The peak and offset values are estimated by taking the median value during pulse and non-pulse active regions, respectively. The standard deviation is determined during non-pulse active regions in order to describe the noise. In general the results show that the offset and pulse-peak values decrease when the LED current increases. Note that, as indicated in Section 1.2, more reflection implies a lower ADC value. An explanation for the fact that the pulse-peak is also reducing in value, is that the LED emits blue

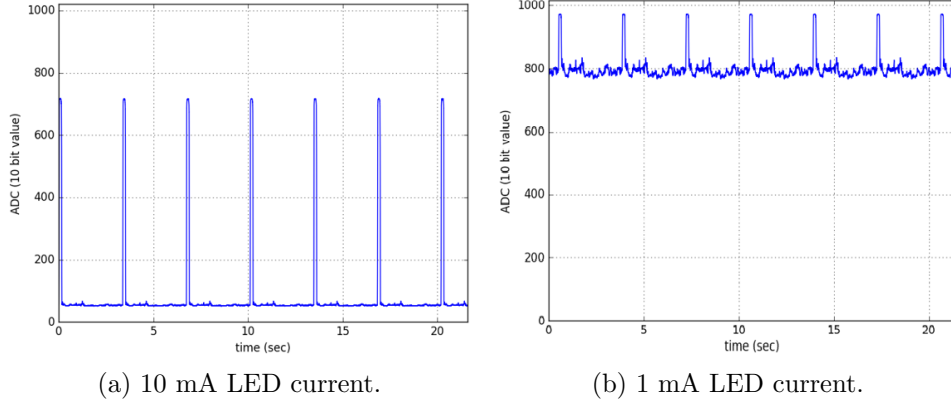


Figure 2.3: Influence of reflection for 10 mA LED current vs. 1 mA in a completely dark environment.

light that does not get fully absorbed by the red or black stripe. Further, the results show that the largest amplitude is at 6 mA. The highest SNR is at around 10 mA, where SNR is defined as the amplitude divided by the noise in the logarithmic domain. Note that 10 mA is the same LED current set by the algorithm developed by Quby.

Figure 2.3 shows the difference between the recorded reflection data with a LED current of 10 mA and 1 mA. One interesting observation, in the case of 10 mA, is the fact that the noise is attenuated significantly compared to the 1mA case. Probably the sensor is saturated, resulting in a reduced sensitivity of the dynamic range where the noise is active. This could also explain the fact that the current algorithm is active in this SNR region, since it has to be less robust against noise.

Since some meters may be placed outside in direct sunlight we also investigated the effect of ambient light on the recorded signals. One example is shown in Figure 2.4 where the LED current is fixed at 1 mA and an external light source is enabled. Because the LED current is set to the region where the photo-reflector sensor is not saturated, ambient light will have a direct influence to the offset, amplitude and noise of the recorded signal. Light has the property of adding up, so the intensity of ambient light will be summed to the intensity of the light emitted by the LED. This means that if the LED current is reduced, the lack of emitted light can be compensated by the ambient light source.

This research will focus on setting the LED current of the photo-transistor as low as possible while still being able to detect pulses. This implies ambient light will become significant and influences the offset, amplitude and noise. Also, the angular velocity of the disk will vary over time because of changing power consumption. Therefore, the features offset, amplitude,

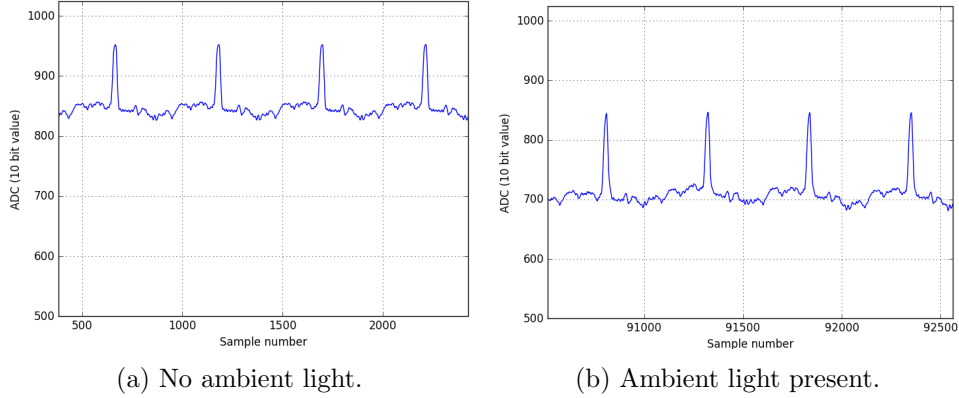


Figure 2.4: Influences of ambient light on a recorded signal with 1 mA LED current.

noise, pulse frequency and slope of the recorded signal need to be taken into account when developing a pulse detection algorithm.

2.3 Related work

Based on previous sections it is clear that our problem can be described as the detection of a pulse in a (noisy) signal. This is a very general problem and can be found in many applications. Applications like pulse oximetry, which measures a person's oxygen level based on a photo-reflector sensor; Electrocardiography (ECG) to measure the heart rate; line finders; and radar communication were examined. None of these methods describe a pulse-detection method addressing all mentioned features. Oximetry sensors are developed in such a way that light from a LED is emitted through a body part and measured at the other side [11]. Since this is considered as a closed system, ambient light can be neglected, and so the offset does not vary strongly over time. According to [12], traditional ECG detection algorithms work on interrupt basis when the sampled signal crosses a fixed threshold value. When this threshold is exceeded, a fixed time window is examined to detect the QRS signal (the pulse introduced by the heart beat). In our application we have no prior information about the maximum time a pulse takes nor the ability to set a hard threshold. Line finders remove ambient light by applying a carrier frequency on the emitted light beam from the LED. The phototransistor demodulates the received signal and will only find the reflection introduced by the LED [13]. Since ambient light has the same influences on the signal as the LED, the LED could for instance be disabled assuming enough ambient light is available. Because the goal of this thesis is to set the LED current as low as possible, it would be a waste to remove the ambient light. In radar communication the transmitted signals are pre-

defined [14]. Since our signal varies in infinite possible pulse frequencies and amplitudes, radar communication algorithms cannot be used.

Since none of the pulse detection methods address all varying signal features, we need to look at the basics of a pulse detection algorithm, which is the ability to differentiate pulses from noise. Motivated by this we decided to use an approach based on signal-detection theory. As will be shown in the next chapter this will give a flexible framework that can be extended to give a robust algorithm for detecting the reflected pulses. In the next sections a brief background on the elements of signal-detection theory will be given that is needed to understand the proposed method.

2.3.1 Likelihood ratio test

Assume we want to detect a variable s with constant value c in white Gaussian noise W , given the sensor output X . Where X and W are independent and identically distributed (i.i.d.) stochastic processes. If s is present in the sensor output, X can be expressed as $X = c + W$. In the case s is not present, then $X = W$. This results in two hypotheses for X :

1. $X = W$, this hypothesis will be named ' H_0 ',
2. $X = c + W$, which will be considered as hypothesis ' H_1 '.

Since s is constant and W is normally distributed, both hypotheses can be represented as normal distributions. The following general equation describes the normal distribution for X given H_0 or H_1 is true:

$$f(x|H) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x-s}{2\sigma^2}}, \quad (2.2)$$

where x represents a realization of X , σ is the standard deviation of the noise and s is the signal to be detected. The variable s will be equal to 0 or c , respectively if H_0 or H_1 is true.

Given the measured sample x , the hypothesis with the highest probability for x should be selected. This can be expressed as:

$$P(H_1|x) \stackrel{H_0}{\underset{H_1}{\gtrless}} P(H_0|x), \quad (2.3)$$

where H_1 is chosen if $P(H_1|x) > P(H_0|x)$ or H_0 is chosen if $P(H_1|x) < P(H_0|x)$. However, $P(H_1|x)$ and $P(H_0|x)$ are not known. Rewriting Equation 2.3 into known quantities using Bayes rule gives:

$$\frac{f(x|H_1) \cdot P(H_1)}{f(x)} \stackrel{H_0}{\underset{H_1}{\gtrless}} \frac{f(x|H_0) \cdot P(H_0)}{f(x)}. \quad (2.4)$$

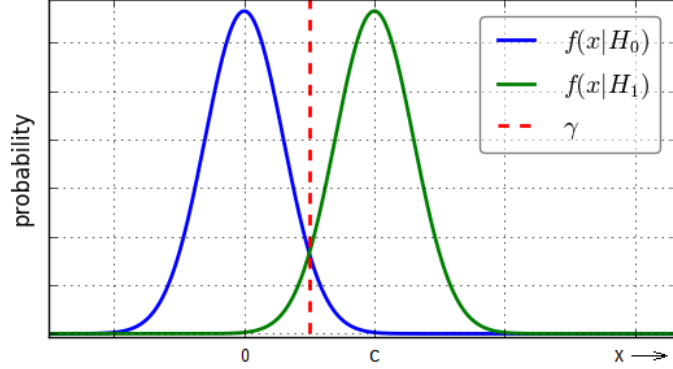


Figure 2.5: Hypothesis testing using two normal distributions and threshold γ . Samples smaller than γ belong to distribution $f(x|H_0)$ and samples larger than γ belong to distribution $f(x|H_1)$.

Assuming $f(x) > 0$, the equation can be simplified into:

$$f(x|H_1) \cdot P(H_1) \underset{H_1}{\overset{H_0}{\gtrless}} f(x|H_0) \cdot P(H_0). \quad (2.5)$$

However, this equation is often written in the form:

$$\Lambda(x) = \frac{f(x|H_1)}{f(x|H_0)} \underset{H_1}{\overset{H_0}{\gtrless}} \frac{P(H_0)}{P(H_1)} = \eta, \quad (2.6)$$

where $\Lambda(x)$ is called the likelihood ratio and η denotes the ratio between the *a priori* probabilities. This equation is referred to as the likelihood ratio test. Using Equation 2.2 and Equation 2.6 the following equation can be derived [14]:

$$x \underset{H_1}{\overset{H_0}{\gtrless}} \frac{1}{c} \left(\frac{c^2}{2} + \sigma^2 \ln(\eta) \right) = \gamma, \quad (2.7)$$

where c is the constant value of signal s , σ is the standard deviation of both distributions and γ denotes the threshold value on x . Figure 2.5 shows the distribution functions for both hypotheses using Equation 2.2. As can be seen from this figure, γ is located at the intersection of both density functions. This implies that every sample left from γ belongs to H_0 while every sample right from γ is classified as H_1 , which corresponds to Equation 2.7.

Nevertheless, wrong classifications can occur, for example there exists a probability that x originated from H_0 but has a higher value than the threshold, meaning it will be classified as H_1 . This type of incorrect classification is called a false alarm and can be mathematically described as:

$$P_{FA} = P(H_1|H_0) = \int_{\gamma}^{\infty} f(x|H_0) dx, \quad (2.8)$$

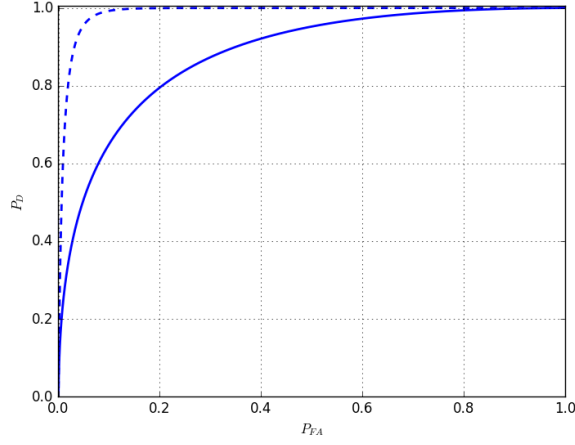


Figure 2.6: Receiver operating characteristic (ROC). Both ROC curves show the relation between P_D and P_{FA} for two different cases.

where $P(H_1|H_0)$ indicates the probability a sample is classified as H_1 while it originated from H_0 . This could also be expressed as the area under the curve of the density function H_0 where the integral covers the domain from γ to infinity. The name false alarm is chosen due to the fact that such a wrong classification will *falsely* detect the presence of signal s .

Another type of wrong classification is called a *miss* and happens when the signal was present (H_1) but was not detected (H_0). This can be described by the following equation:

$$P_M = P(H_0|H_1) = \int_{-\infty}^{\gamma} f(x|H_1)dx, \quad (2.9)$$

where the area under the curve of H_1 is calculated for the domain negative infinity to γ . Instead of using the probability for a miss, more often the probability to correctly *detect* the signal is used. This probability is called P_D and since the area of $f(x, H_0)$ is 1 by definition it follows that $P_D = 1 - P_M$. So P_D can also be described as the area under the curve of H_1 right from γ , see equation 2.10.

$$P_D = P(H_1|H_1) = \int_{\gamma}^{\infty} f(x|H_1)dx \quad (2.10)$$

The P_D and P_{FA} represent the performance of the likelihood ratio test. To control P_D and P_{FA} , γ can be altered. The relation between P_D and P_{FA} can be shown in a Receiver Operating Characteristic (ROC) curve where the threshold value is set from $-\infty$ to ∞ [14]. Figure 2.6 shows an example of two different ROC curves. A general approach is to keep the probability of the false alarm below a predefined value while maximizing the probability of detection. When the desired P_D and P_{FA} couple has been found, the

corresponding γ can be determined. Mathematically γ can be expressed in terms of P_D or P_{FA} by using the inverse cumulative distribution function (CDF) of H_0 or H_1 , also known as the Quantile function (Q). For normal distributions the quantile function can be scaled with respect to μ and σ [15]. In the following equations we assume that the quantile function is the inverse cumulative distribution function of the standard normal distribution:

$$\gamma = Q(P_{FA})\sigma_0 + \mu_0, \quad (2.11)$$

$$\gamma = Q(P_D)\sigma_1 + \mu_1, \quad (2.12)$$

where μ_0 and σ_0 are the mean and standard deviation of the distribution for hypothesis H_0 and μ_1 and σ_1 for H_1 .

Note that from Figure 2.6 it can be observed that the classification will improve if the solid line shifts more to the left top corner, like the dashed line. This shift can be achieved by moving the density functions apart. To indicate the distance between two normal distributed density functions the sensitivity index can be used:

$$d' = \frac{\mu_1 - \mu_0}{\sqrt{\frac{1}{2}(\sigma_1^2 + \sigma_0^2)}}, \quad (2.13)$$

where d' represents the sensitivity index, μ_0 and μ_1 the means of the distributions, and σ_0 and σ_1 the standard deviation of the distributions. When two density functions cannot be discriminated, d' equals zero. The more two density functions move apart the higher d' will be. When $P_D = 0.99$ and $P_{FA} = 0.01$, d' will be 4.65, which is considered by many researchers as the upper bound of d' [16].

2.3.2 Multiple-sample likelihood ratio test

To improve the classification between the assumed H_0 and H_1 even further, the multiple sample likelihood ratio test is introduced [14]. Instead of making a classification based on only one sample, N samples are used. During this approach it is assumed that at least N samples are measured from the same hypothesis in a row. Also each sample is modeled as a realization from a normal i.i.d. distributed stochastic process. For each of the N samples the probability to be in a hypothesis is calculated, then all probabilities are multiplied to each other. This multiple-sample likelihood ratio test is given by:

$$\Lambda(x) = \frac{f(x[1], x[2], \dots, x[N]|H_1)}{f(x[1], x[2], \dots, x[N]|H_0)} \underset{H_1}{\overset{H_0}{\gtrless}} \frac{P(H_0)}{P(H_1)} = \eta. \quad (2.14)$$

Since the samples are considered statistically independent:

$$f(x[1], x[2], \dots, x[N]|H) = \frac{1}{(2\pi\sigma)^{(N/2)}} \prod_{i=1}^N e^{-\frac{x[i]-s[i]}{2\sigma^2}}, \quad (2.15)$$

where $s[i]$ is the expected signal value for each of the corresponding measured samples $x[i]$. Testing for the hypothesis where the signal was not present (H_0), results in $s[i] = 0$ for every i . In the case of the hypothesis where the signal is present (H_1), $s[i] = c$ for every i . Note that similar to Equation 2.6, the multiple sample likelihood can be rewritten in the form:

$$\sum_{i=1}^N x \underset{H_1}{\overset{H_0}{\gtrless}} \frac{1}{c} \left(N \frac{c^2}{2} + \sigma^2 \ln(\eta) \right) = \gamma. \quad (2.16)$$

Since x is a realization from an i.i.d. normal distribution, the $\sum_{i=1}^N x$ remains normally distributed [17]. Because of this property γ can still be expressed using Equation 2.11 in P_D and P_{FA} .

2.3.3 Matched filter

The matched filter is a special case of the multiple-sample likelihood ratio test discussed in Section 2.3.2. Where the multiple sample likelihood ratio test assumes in Equation 2.15 that $s[i]$ is the same for every i , the matched filter does not. This way $s[i]$ is not fixed to a specific hypothesis and can be used to describe a transition in a signal, for instance a part or a complete transition of a pulse signal. Equation 2.17 shows the mathematical representation of a matched filter. In this equation $y[n]$ is the matched filter output, $x[i]$ represents the sampled signal and $s[n-i]$ is the time-reversed expected signal also called the impulse responds or template. Note that $s[i]$ is time reversed since this is considered optimal. For the full derivation from the likelihood ratio test to the matched filter see [14].

$$y[n] = \sum_{i=-\infty}^{\infty} x[i]s[n-i] \quad (2.17)$$

Shifting the template over the sampled signal will maximize the SNR by using the correlation between them.

Chapter 3

Pulse Detection Algorithm

For the proposed method we use a signal-detection approach, similarly as described in Section 2.3, where the signal statistics can vary over time. Figure 3.1 shows a schematic overview of the proposed method. From this figure we can divide the algorithm into the following three components:

- **Estimate signal statistics:** This stage estimates the assumed signal statistics such as means and variances that may vary, e.g., due to ambient light.
- **Likelihood ratio test:** Based on the estimated statistics the signal is classified whether the pulse is present or not. This stage is able to detect pulses over the full domain of household power consumptions [0, 17 kW].
- **Pulse to energy consumption:** Here the detected pulses are converted into the energy consumption and instantaneous power usage.

In the remainder of this chapter we explain the assumed statistical model, together with a detailed explanation of each of the individual three components. In each section we will motivate the proposed approach and will indicate the contributions with respect to the related work.

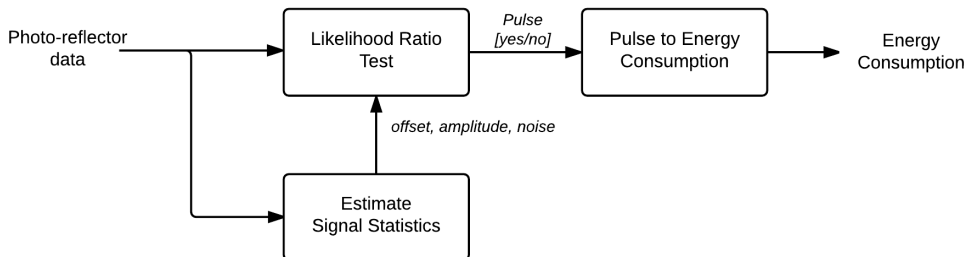


Figure 3.1: Pulse detection algorithm components.

3.1 Assumed statistical model

For the proposed method we use a signal detection approach for which we need to describe the reflected signal with a statistical model. Let $x[n]$ denote the observed sensor value at sample index n . The signal will be modeled with a stochastic process denoted by X_n . In the case that x originates from the reflection of the metal part of the disk, we model x as a normal i.i.d. distributed stochastic process with mean μ_0 and variance σ_0^2 . Similarly, in case of a stripe reflection, the signal is modeled as a normal i.i.d. distributed stochastic process with mean μ_1 and variance σ_1^2 . Note that we adjusted the statistical model from Section 2.3 by using two different variances for the pulse and non-pulse region in the signal rather than one. It is clear from Figure 3.2 that this is more in line with the actual signal, where the noise variance during the strip reflection is significantly smaller than the noise variance during the metal-reflected part. Also note that μ_0 and μ_1 actually represent the signal features offset (μ_0) and peak (μ_1), described in Section 2.2.

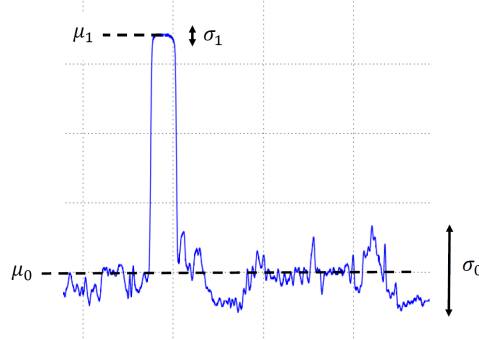


Figure 3.2: Signal statistics description.

3.2 Estimate signal statistics

For the likelihood ratio test to successfully detect if a pulse is present, the signal statistics need to be known. In this section an approach will be discussed to estimate μ_0 , μ_1 , σ_0 and σ_1 . We assume that these statistics vary slowly over individual disk revolutions, since they are influenced by ambient light, typically sunlight, which varies slowly over time. Using the property of a disk that the rotation speed is proportional to the power consumption, we propose a method where the signal statistics are only updated if the recorded signal of a revolution corresponds to a predefined power consumption range.

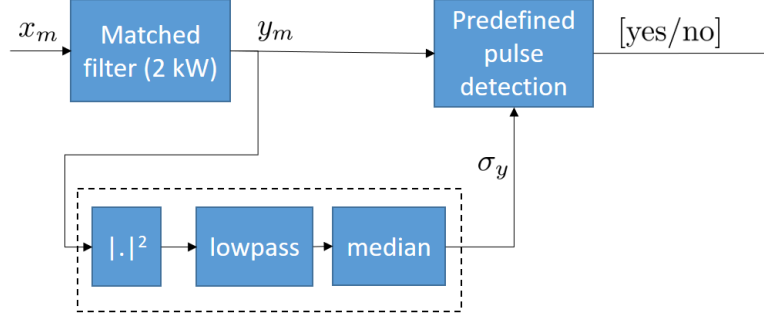


Figure 3.3: Estimate signal statistics diagram.

The suggested approach is indicated in Figure 3.3. First the signal is split in a short time window, where the signal in this window is indicated as X_m . For each window it will be verified if a pulse, corresponding to a predefined power consumption, is present. Only if this pulse is present, the signal statistics are estimated from this window. Although this method is dimensioned for the specific power consumption of 2 kW, the method has a deviation of around 30% for power consumptions below 4 kW.

The window size is equal to the number of samples required to record exactly one revolution of the disk for the predefined power consumption of 2 kW. Since the rotation speed is proportional to the power consumption, we can calculate the number of samples required with the following equation:

$$\text{window size} = \frac{3600}{P \cdot C} \cdot f_s, \quad (3.1)$$

where P is the power in kW, C is the C-value, which is assumed to be known, and f_s is the sampling frequency in Hz. Because of the physical properties of the disk, 95% of the samples originate from the metal-reflected part, while the remaining 5% originates from the stripe-reflection. So given the power consumption and properties of the disk, the period and duty cycle of the signal can be predicted. The period represents one revolution of the disk and so consists of the same number of samples as the calculated window size. The duty cycle for each period equals 5%. Since the photo-reflector sensor properties, disk rotation speed and duty cycle, are known, the measured pulse can be fully predicted.

To verify if the signal in the window corresponds to this predicted signal a matched filter is applied, which output is indicated as y_m in Figure 3.3. As a template the rising edge of the expected signal is chosen, since this is most characteristic for the predefined power consumption. However, at this point the signal statistics are still unknown, so we cannot use the matched filter as was presented in Section 2.3.3 that assumes the template has the

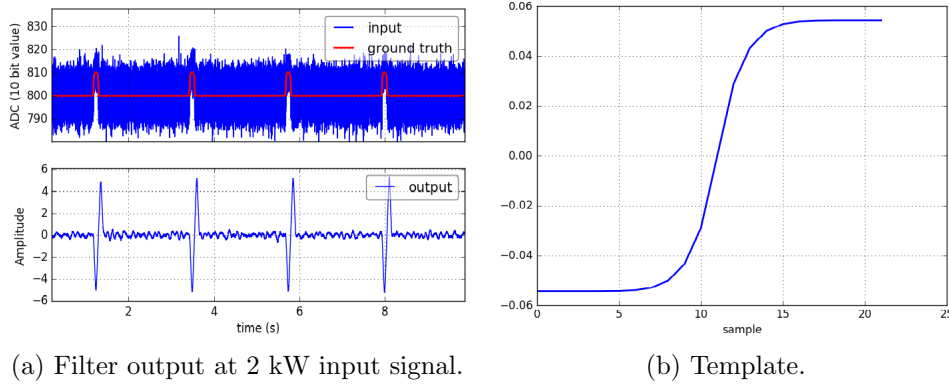


Figure 3.4: Matched filter output with corresponding template.

exact dimensions as the input signal. Therefore, the template is normalized and centered around zero. Shifting this template over the input data will give a peak value when the template matches and a value around zero when it does not. The scale of the matched filter output remains unknown since the scale of the input signal is not known. Figure 3.4 shows an example of the matched filter method. The template is the rising edge corresponding to a predefined power consumption of 2 kW. The input is a simulated signal with noise added, representing the recording of a spinning disk at 2 kW. As a reference, the input signal without noise is plotted in red. During the occurrence of a pulse, there are two peaks present in the output of the matched filter: a negative and positive one. The negative peak comes from the template matching the rising edge of the pulse. The positive one matches the falling edge of the pulse. The reason the template also responds to the falling edge is because the signal can be described as the negative of the template. In order to define whether a found negative (positive) peak in the matched filtered signal relates to the rising (falling) edge of the expected pulse in the input signal, we have to define a threshold. Since the scale of the output value is unknown, a fixed threshold cannot be used. As an alternative a variable threshold scaled with the standard deviation of y_m is applied. The scalar is chosen in such a way that the threshold value will be away from the noise. This way only peaks that differ significantly from this noise are detected when crossing the threshold value.

To estimate the standard deviation of y_m , indicated as σ_y , some signal processing is applied as can be seen in Figure 3.3. First the negative values of y_m are removed by using the absolute quadratic value. Then the signal is low pass filtered by applying a moving average with a window size smaller than the area consumed by the predicted pulse. From this filtered signal the median value will give an estimation for the variance of y_m . To get σ_y the square root of the variance is applied.

When the scalar is chosen to be equal to 5, two thresholds: $5 \cdot \sigma_y$ and

$-5 \cdot \sigma_y$ make sure only positive and negative peaks that are significantly different from the noise are considered as part of the expected pulse. If both thresholds are crossed, the window contains a pulse corresponding to the predefined power consumption. As mentioned before this method does not only apply to exactly the predefined power consumption. The matched filter will also output large peak values for power consumptions close to the predefined power consumption, but these peaks will be a little smaller. Dependent on the set scalar value these peaks will also exceed the thresholds. After experimenting, a trade off was made between the ability to differentiate the peak from the noise, and the detection range of the predefined power consumption, by setting the scalar value to 5.

If a pulse is detected the signal statistics can be estimated as will be described below. Note that for each time the signal statistics are estimated, the values will be updated with a factor 0.7 to make the system robust for outliers.

3.2.1 Estimate μ_0

The default equation to estimate the mean is [18]:

$$\mu = \frac{1}{N} \sum_{i=0}^{N-1} x_i, \quad (3.2)$$

where x_i are samples from the same distribution and N are the total number of samples. Unfortunately, this equation cannot be used since the window consists of samples from the metal-reflected part and the pulse-reflected part, which are both from a different distribution. To apply this method for μ_0 , first the pulse-reflected samples need to be removed from the window, which corresponds to removing the highest values present in the window. However, since the window is sensitive for a range of power consumptions it is unknown how many samples of the window need to be removed. Removing a fixed number of samples, for example 5% of the highest values, could then result in over- or under-estimation of μ_0 . Therefore, we propose to apply the median value on the window to estimate μ_0 . Although this is a rough estimation, this method is found 20% closer to the actual value of μ_0 than Equation 3.2.

3.2.2 Estimate μ_1

To estimate μ_1 the area between the negative and positive peak in the matched filter output (y_m) is examined. This area corresponds to the area between the center of the rising edge and the center of the falling edge of the pulse in the input signal (x_m), when the delay of the filter is taken into account. The sample of y_m equal to zero between the negative and positive

peak always corresponds to exactly the middle of the pulse in x_m . Therefore, μ_1 is estimated as the sample of x_m that corresponds to the first sample of y_m crossing the zero line between the negative and positive peak.

Note that we cannot easily use the positive and negative peaks in the matched filter output (y_m) to remove the pulse in x_m and determine μ_0 . The reason for this is that the filter is maximum correlated with the center of the rising and falling edge of the pulse, meaning it is hard to detect the start and end point of the pulse.

3.2.3 Estimate σ_0

The general equation to estimate the standard deviation is [18]:

$$\sigma^2 = \frac{1}{N-1} \sum_{i=0}^{N-1} (x_i - \mu)^2, \quad (3.3)$$

where σ is the standard deviation. For the same reasons as using Equation 3.2, this equation cannot be used. Instead, a similar approach used to determine σ_y is applied. But first, the signal is centered around zero by subtracting the estimated μ_0 value. Now the same signal-processing steps as for σ_y can be applied to retrieve σ_0 .

3.2.4 Estimate σ_1

To estimate σ_1 the area of the pulse needs to be examined. However, because of the properties of the photo-transistor sensor, σ_1 is always smaller than σ_0 . After trial and error, $\sigma_1 = \frac{\sigma_0}{3}$ is found to give in general a realistic estimation of σ_1 .

3.3 Likelihood ratio test

Section 2.3 showed that based on a statistical model a signal can be detected using a hypothesis testing method. In our application the metal-reflected part and the stripe-reflected part of the signal need to be differentiated. To differentiate between these signals, the multiple sample likelihood ratio test is used as described in Section 2.3.2.

Using the assumed statistical model of our signal, the hypotheses are described as:

$$H_0 : X_n \sim \mathcal{N}(\mu_0, \sigma_0^2), \quad (3.4)$$

$$H_1 : X_n \sim \mathcal{N}(\mu_1, \sigma_1^2), \quad (3.5)$$

where $n = 1, 2, \dots, N$, H_0 represents the metal-reflected signal and H_1 the stripe-reflected signal. In contrast to Section 2.3.2 the standard deviation

of both hypotheses are unequal and hypothesis H_0 is not centered around zero. Therefore, the classification of N samples based on $\sum_{n=1}^N x_n \underset{H_1}{\overset{H_0}{\leq}} \gamma$, shown in Equation 2.16, is not valid anymore and needs to be redefined. Using Equation 2.15 where $s[i]$ equals μ_0 and σ equals σ_0 for hypothesis H_0 , and $s[i]$ equals μ_1 and σ equals σ_1 for hypothesis H_1 , the multiple sample likelihood ratio test equation shown in Equation 2.14 can be rewritten as:

$$Y = (\sigma_1^2 - \sigma_0^2) \sum_{n=1}^N x_n^2 + 2(\mu_1\sigma_0^2 - \mu_0\sigma_1^2) \sum_{n=1}^N x_n \underset{H_1}{\overset{H_0}{\leq}} \gamma$$

$$N\mu_1^2\sigma_0^2 - N\mu_0^2\sigma_1^2 + 2\sigma_0^2\sigma_1^2 \ln \left(\left(\frac{\sigma_1}{\sigma_0} \right)^N \eta \right) = \gamma, \quad (3.6)$$

where we introduce a new random variable Y denoting the left-hand term of the equation and γ is our decision threshold. In order to set this threshold to a meaningful value we want to have γ expressed in terms of the false alarm rate (P_{FA}) and the detection rate (P_D). Observing Y reveals that it is a summation of a normal distribution and a Chi squared distribution that are not independent. This means expressing γ in terms of P_D and P_{FA} with the help of the quantile function, as shown in Equation 2.11, is a challenge, because Y is not normally distributed as is the case in Equation 2.14.

To overcome this we propose to approximate Y with a normal distribution with mean $\tilde{\mu}$ and variance $\tilde{\sigma}^2$. Since we will use a sum of N independent terms we expect this approximation to be valid due to the central limit theorem. This introduces the following new hypotheses related to the new variable Y :

$$H_0 : Y \sim \mathcal{N}(\tilde{\mu}_0, \tilde{\sigma}_0^2), \quad (3.7)$$

$$H_1 : Y \sim \mathcal{N}(\tilde{\mu}_1, \tilde{\sigma}_1^2). \quad (3.8)$$

From these new distributions the mean and standard deviation need to be estimated in order to apply Equation 2.11 to determine a value for γ . First we will discuss how to estimate the mean, and then we will discuss the variance.

Mean

Following Equation 3.6, the mean value of Y can be expressed in terms of its first and second order non-central moments:

$$E[Y] = \alpha \sum_{n=1}^N E[X_n^2] + \beta \sum_{n=1}^N E[X_n], \quad (3.9)$$

where

$$\alpha = (\sigma_1^2 - \sigma_0^2), \quad (3.10)$$

Table 3.1: Non central moments from [19].

Order	Non-central moment
1	μ
2	$\mu^2 + \sigma^2$
3	$\mu^3 + 3\mu\sigma^2$
4	$\mu^4 + 6\mu^2\sigma^2 + 3\sigma^4$

and

$$\beta = 2 (\mu_1 \sigma_0^2 - \mu_0 \sigma_1^2). \quad (3.11)$$

By using Table 3.1, which shows the non-central moments for different orders, the following equations can be derived:

$$H_0 : E[Y] = \alpha N (\mu_0^2 + \sigma_0^2) + \beta N \mu_0, \quad (3.12)$$

and

$$H_1 : E[Y] = \alpha N (\mu_1^2 + \sigma_1^2) + \beta N \mu_1. \quad (3.13)$$

Variance

The variance of Y is given by:

$$\text{Var}[Y] = E[Y^2] - E[Y]^2. \quad (3.14)$$

Since $E[Y]$ is known from the previous section, only the second moment of Y has to be found:

$$E[Y^2] = E \left[\left(\sum_{n=1}^N \alpha X_n^2 + \beta X_n \right)^2 \right]. \quad (3.15)$$

Squaring the sum in the previous equation will result in N^2 terms, from which N terms are statistically dependent ($X_n, X_m, n = m$) and $N^2 - N$ cross terms are independent ($X_n, X_m, n \neq m$). Let the dependent and independent sum of cross terms be denoted by ξ and ψ then we have:

$$E[Y^2] = N\xi + (N^2 - N) \psi, \quad (3.16)$$

where

$$\xi = \alpha^2 E[X^4] + 2\alpha\beta E[X^3] + \beta^2 E[X^2], \quad (3.17)$$

and

$$\psi = \alpha^2 E[X^2]^2 + 2\alpha\beta E[X^2]E[X] + \beta^2 E[X]^2, \quad (3.18)$$

and $E[X]$, $E[X^2]$, $E[X^3]$ and $E[X^4]$ can be expressed in the known quantities μ and σ with the help of Table 3.1.

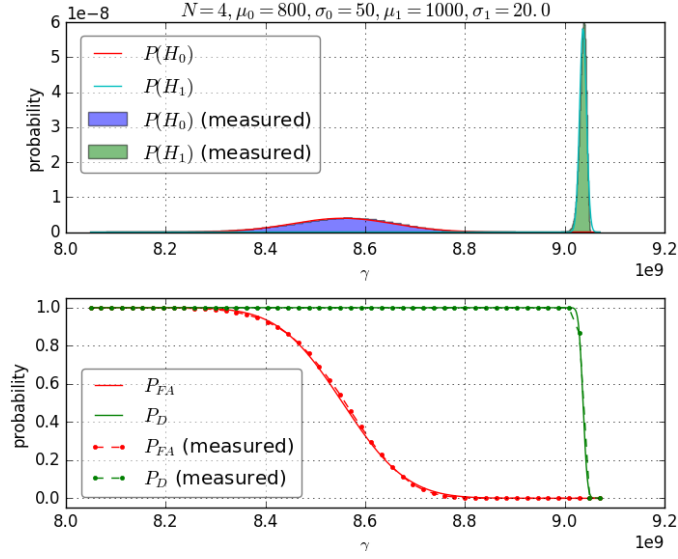


Figure 3.5: Measured and estimated variables of the statistics of Y .

In Figure 3.5 an example is shown where the statistics of Y are estimated by generating 100,000 realizations of X . For X we choose the following statistics, in the case of H_0 : $\mu_0 = 800$ and $\sigma_0 = 50$, and in the case of H_1 : $\mu_1 = 1,000$ and $\sigma_1 = 20$. These values correspond to the observations when the LED is set to 1 mA, as was shown in Figure 2.3b. N is related to the sampling frequency as will be shown in the next section. In this example N equals 4, which results in a reasonable sampling frequency. The realizations are then compared with the analytic results under the assumption that Y was normally distributed. The top graph of Figure 3.5 shows the histograms, based on the measured realizations, and the normal probability density functions based on the mean and variances found in the previous section. The bottom graph shows the estimated and analytical values of P_D and P_{FA} . Based on the figure it is clear that for this particular example the expressions for P_D and P_{FA} can be well approximated under the assumption that Y is normally distributed.

Using the approximated normal distribution of Y , γ can finally be determined for a chosen P_D or P_{FA} . Different from Section 2.3 we propose to use two thresholds, $\gamma_{\min}$ and $\gamma_{\max}$ where $\gamma_{\min}$ is closer to μ_0 and $\gamma_{\max}$ is closer to μ_1 . When sample $Y_n < \gamma_{\min}$ it will be classified as H_0 , and if $Y_n > \gamma_{\max}$ it will be classified as H_1 . In the region between $\gamma_{\min}$ and $\gamma_{\max}$ no classification is made, which prevents unwanted fast switching between the two hypotheses, also known as hysteresis.

Figure 3.6 shows an example where $\gamma_{\min} = \tilde{\mu}_0$ and $\gamma_{\max}$ is related to $P_{\text{FA}} = 0.02\%$. $\gamma_{\min}$ is chosen to be exactly at the center of the noise as can be seen from Figure 3.7. This figure represents the signal corresponding to Figure 3.6. Note that $\gamma_{\max}$ is removed from the noise, which is the result of a low value for P_{FA} .

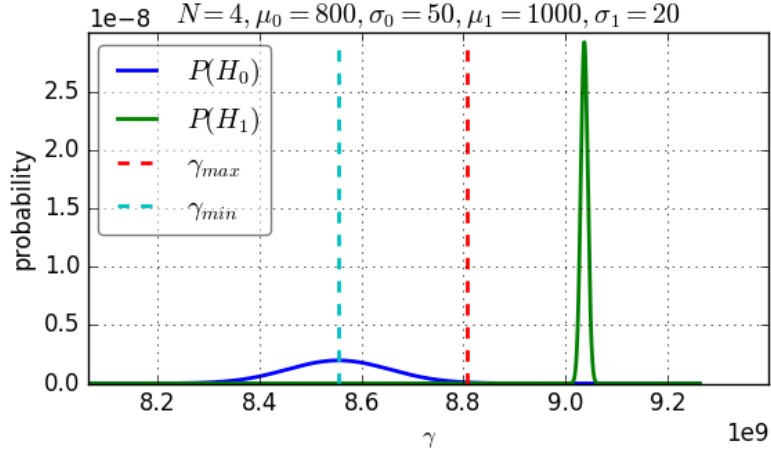


Figure 3.6: Distributions of Y with $\gamma_{\min} = \tilde{\mu}_0$ and $\gamma_{\max} = Q(0.02)\tilde{\sigma}_0 + \tilde{\mu}_0$.

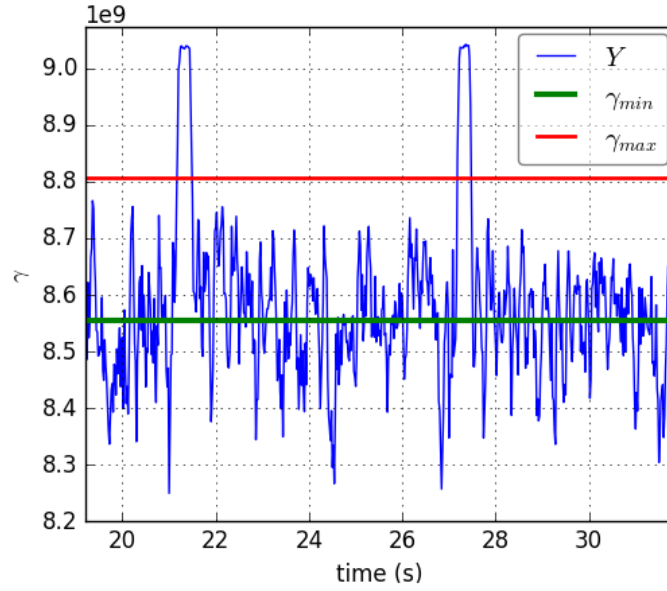


Figure 3.7: Simulated signal corresponding to the distributions shown in Figure 3.6.

3.4 Pulse to energy consumption

The previous section discussed how pulses are detected. The output of the algorithm needs to report the instantaneous power consumption and total energy consumption. Since the C-value indicates the number of revolutions per kWh, and each detected pulse indicates a revolution, the energy consumption can be calculated by counting all detected pulses divided by the C-value. This can be expressed in an equation as follows:

$$E = \frac{N_{\text{pulse}}}{C}, \quad (3.19)$$

where, E is the energy in kWh, N_{pulse} the total number of detected pulses and C is the C-value.

To estimate instantaneous power consumption the time between two pulses need to be examined. Given the C-value and the interval between two pulses the instantaneous power can be calculated by the following equation:

$$P_{\text{avg}} = \frac{3600}{t_{\text{pp}} \cdot C}, \quad (3.20)$$

where P_{avg} is the power in kW and t_{pp} the time between two pulses in seconds. Note that between two detected pulses it is only possible to calculate the average power consumption, because t_{pp} is dependent on the power consumption, which can vary over time.

3.5 Sampling frequency and duty cycle

The minimum sampling frequency depends on the maximum rotation speed of the disk and the number of samples required in the area of the stripe to successfully classify the stripe as a pulse. As discussed in Section 1.1.3 the rotation speed of the disk is proportional to the measured power consumption and depends on the C-value. Since the stripe covers 5% of the total disk surface, the sampling frequency can be calculated with the following equation:

$$f_s = \frac{\frac{P}{3600} \cdot C}{0.05} \cdot (N + 1), \quad (3.21)$$

where f_s is the sample frequency in Hz, P is the power consumption in kW, C is the C-value in rev/kWh and N is the number of samples required for the classifier. Considering the worst case scenario: the fastest rotation speed will occur with a meter having the highest encountered C-value of 800 rev/kWh, and the power consumption at maximum just below the main fuse. For the highest possible power consumption in a consumer household, we assume a three-phase grid with each a main fuse of 25A at 230V, resulting in an overall maximum power consumption of $75\text{A} \cdot 230\text{V} = 17.25\text{kW}$. Assuming the

Table 3.2: Measured rise and fall times of the photo-reflector.

	LED	Phototransistor
t_r	200 ns	374 μs
t_f	800 ns	426 μs

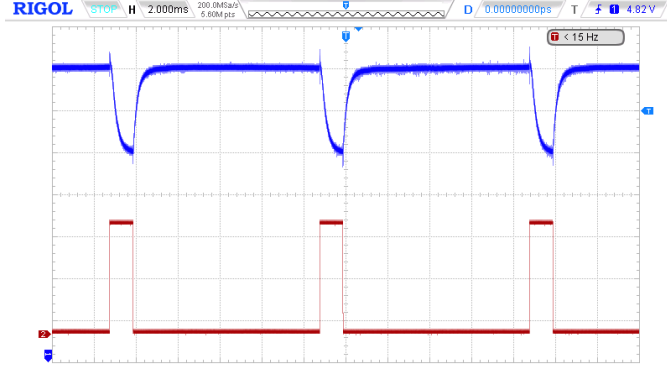


Figure 3.8: Phototransistor response (in blue) to enabling the LED (in red). Measured with the oscilloscope.

likelihood ratio test needs at least 4 samples from the stripe, the minimal sampling frequency will be: $\frac{17.25}{\frac{3600}{0.05}} \cdot 800 \cdot 5 \approx 400\text{Hz}$. This implies the sampling frequency is reduced by at least a factor 25 with respect to the fixed 10 kHz used by the algorithm developed by Quby.

The next step will be to duty cycle the LED to preserve energy. This could be achieved with the following approach: enable the LED, sample the phototransistor, and disable the LED afterwards. In this approach the minimal on-time of the LED needs to be obtained by investigating the time required to take a valid sample after the LED is enabled. To get more insight in the timings of the LED and phototransistor the rise and fall times need to be examined. Since the datasheet does not provide this information for low currents applied to the photo-reflector, the rise and fall times are measured with an oscilloscope. Table 3.2 shows the measured rise and fall times.

Figure 3.8 shows that when the LED is enabled and so light falls onto the phototransistor, the voltage over the phototransistor drops. When the LED is turned off the voltage over the phototransistor increases again. To calculate the minimum time the LED needs to be on, the fall time of the phototransistor needs to be inspected. The rise time of the phototransistor can be ignored since we are only interested in the on-time of the LED. To calculate the minimal on-time the following equation is used:

$$t_{\text{LED,on}} = t_{r,\text{LED}} + t_{f,\text{phototransistor}} + t_{\text{ADC}} + t_{f,\text{LED}}, \quad (3.22)$$

where t_r stands for rise time, t_f for the fall time and t_{ADC} for the sample and hold time of the ADC. The sample and hold time of the examined microcontroller is defined as $2\ \mu\text{s}$. Using Table 3.2 and Equation 3.22 the minimal on-time of the LED will be $0.2\mu\text{s} + 426\mu\text{s} + 2\mu\text{s} + 0.8\mu\text{s} = 429\mu\text{s}$.

Assuming a sampling frequency of 400 Hz, the LED can be duty cycled around 20% while still provide valid readings.

Chapter 4

Evaluation

To evaluate the presented algorithm of Chapter 3, a Python implementation was created running on a PC. Supplying the photo-reflector signal as an input to the Python implementation gives the estimated power consumption (Watt) and energy consumption (kWh) over time as output. These predicted electricity consumption values will be compared with a ground-truth reference and more insights will be given in the algorithmic behavior. For this work the proposed algorithm was tested in a lab environment where extreme conditions, such as fast switching power consumptions and changes in ambient light, can be simulated. Next to the lab experiments, measurements were performed at two households with a different analog meter. First, the lab experiments will be discussed and thereupon the experiment performed at the households.

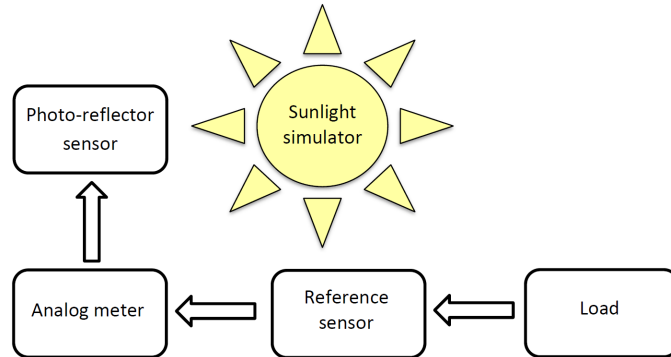


Figure 4.1: Schematic overview of the lab test setup.

Table 4.1: Meter properties lab setup.

	C-value	Stripe color	Disk type
Meter-750	750	black	serrated
Meter-600	600	red	smooth
Meter-350	350	black	smooth

4.1 Lab experiments

4.1.1 Test setup

Figure 4.1 shows a schematic overview of the test setup in the lab environment. An electric heater with different heating levels is used as the load. Between the analog meter and the heater a Fibaro wall plug [20] is placed, which measures the power and energy consumption of the heater. These measurements are logged and used as a reference for the output of the algorithm. The phototransistor is logged with an Arduino, while the sensor is attached to the analog meter. A specifically-designed light source is used to simulate sunlight. Below the mentioned components are described in more detail.

Analog meters

The experiments were performed with three different analog meters. The meters differ in C-value, stripe color and disk type. Table 4.1 gives an overview of the different meters.

Load

The electric heater has four different heating levels, which can be changed manually with a switch. The different levels correspond to the following power consumptions: 0 W, 700 W, 1250 W and 2000 W. Note that the heater has no thermostat, meaning this power is consumed continuously.

Photo-reflector logger

Since the meter module is not able to store the photo-reflector sensor data to a storage device, an Arduino UNO with the photo-reflector sensor, a real-time clock (RTC), and an SD-card extension board is used. At a sampling frequency of 1 kHz, the photo-reflector sensor is sampled with the Arduino's internal 10 bit ADC and stored to an SD-card. The RTC makes sure that the stored photo-reflector signal is provided with a timestamp such that it can be easily compared with the reference signal. The sampling frequency of 1 kHz is the highest possible sampling frequency found at which we are still able to store the samples to an SD card. However, in the Python model this

frequency can be downsampled to the required sampling frequency according to Equation 3.21. The LED of the photo-reflector is limited to a constant current flow of 1 mA with the help of a resistor.

Reference sensor

To measure the energy and power consumption of the load, a Fibaro wall plug is used. The wall plug can measure power consumptions up to 3 kW and has an accuracy of $\pm 0.5\%$ between 0 and 1 kW, $\pm 1.5\%$ between 1 kW to 2 kW, and $\pm 2\%$ above 2 kW [21]. It compensates for its own consumption and has a resolution of 0.1 W and 0.01 kWh for power and energy consumption, respectively. To interface with the Fibaro wall plug, the wireless Z-wave protocol is used. Using a Raspberry Pi and a Z-wave dongle, the wall plug is configured to send an update of the power and energy consumption at a frequency of 1 Hz. A Python script running on the Raspberry Pi logs these consumptions together with a timestamp to the SD-card.

Sunlight simulator

To simulate different levels of sunlight a GTI gle-m4/32 light source is used [22]. This device can emit light in the same relative spectral-power distribution as sunlight. The measured intensity of the light source is 2200 lux at a distance of 50 cm. As a reference, our office lighting is found to be around 300 lux. When measured outside at an overcast day an intensity of 1400 lux was found. However, on sunny days the intensities can vary between 10,000 and 25,000 lux [23]. Since the GTI gle-m4/32 has only two settings, on and off, we decided to use a set of neutral density filters. These filters reduce the light intensity with a fixed percentage equally over the full spectrum, meaning the spectral power distribution corresponding to sunlight will remain. The lowest filter available was a neutral density filter that transmits 70% of the incoming light intensity. Stacking these filters will imply a transmission rate of 0.70^N , where N is the number of filters.

4.1.2 Switching power consumption

In this experiment the load is varied between the four different power consumptions for a period of 25 minutes while the ambient light is kept at an intensity of 200 lux. The first 20 minutes the load is varied slowly and the last 5 minutes the load is switched rapidly. A rapidly switching load is for example common in the case when a coffee machine is used. Figure 4.2 shows an example output of the algorithm during the experiment performed with Meter-750. The top graph shows the recorded signal of the photo-reflector sensor. A ‘+’ sign indicates when a pulse is detected. This location corresponds to the first sample that crosses the $\gamma_{\max}$ threshold in the likelihood block. The second graph shows the calculated power consumption based on

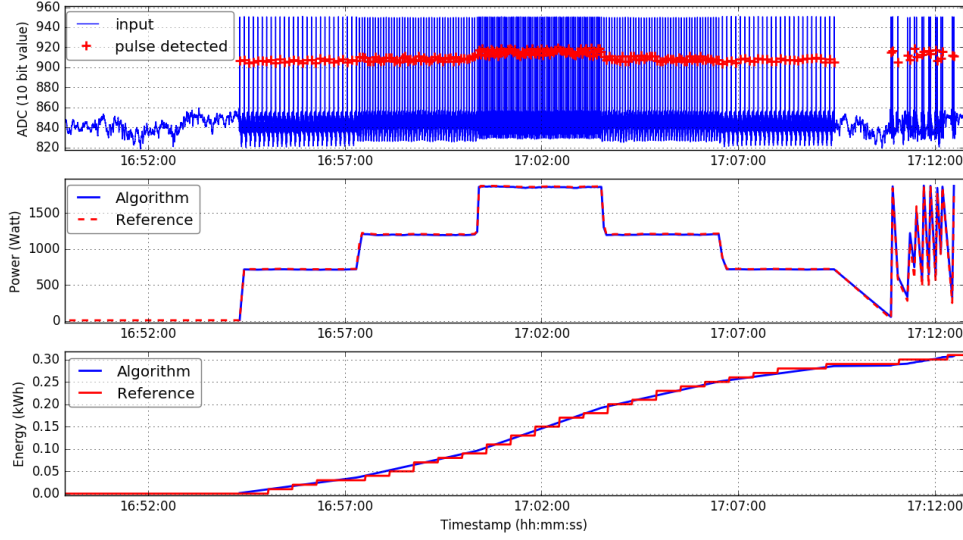


Figure 4.2: Switching power consumption experiment. The first 20 minutes the load is varied slowly and the last 5 minutes the load is switched rapidly.

the detected pulses, compared to the reference sensor. As can be seen from this graph the algorithm closely follows the reference. The bottom graph shows the energy calculated by the algorithm and the energy consumption measured by the reference sensor. Note that the reference sensor only updates the energy after a 0.01 kWh increment. To indicate the performance of the algorithm the Root-mean-square error (RMSE) of the power consumption is calculated. For the case of the energy consumption, the difference between the signals is examined at the end of the experiment when the last pulse is detected. This difference will be expressed as a percentage error. Table 4.2 shows the results.

Table 4.2: Results switching power consumption experiment.

	RMSE (Watt)	Algorithm (kWh)	Reference (kWh)	Error
Meter-750	24.9	0.307	0.310	-0.97%
Meter-600	33	0.218	0.220	-0.91%
Meter-350	37	0.268	0.270	-0.74%

Examining the recorded signals more closely, showed that all pulses for all meters were detected. This means the calculated errors are most likely caused by the inaccuracy of the Fibaro wall plug.

Table 4.3: Results ambient light experiment.

	RMSE (Watt)	Algorithm (kWh)	Reference (kWh)	Error
Meter-750	64	1.318	1.33	-0.90%
Meter-600	81	1.357	1.37	-0.95%
Meter-350	75	1.058	1.07	-1.12%

4.1.3 Ambient light influences

In the following experiment, the load is fixed at 2 kW while ambient light is increased from 30 to 2200 lux. Twelve density filters are stacked and placed between the photo-reflector and the sunlight simulator. Then, every 4 minutes two density filters are removed, resulting in a doubling of the light intensity. Figure 4.3 shows the data of the experiment performed with Meter-350. The signal statistics are updated after every pulse because the matched filter in the ‘estimate signal statistics’ part of the algorithm is dimensioned to a predefined power consumption of 2 kW, which for this experiment equals the input signal. The ‘likelihood ratio test’ part of the algorithm uses these statistics and detects the actual pulses in the signal. However, the moment the last two filters were removed, three consecutive pulses were missed. As a consequence, the calculated power is off by 1400 watt, indicated in Figure 4.3 by the drop in power. The cause of these pulse misses is that μ_0 and μ_1 are updated with a factor 0.7. Since the step from 1100 to 2200 lux introduced a rather large change in offset, updating μ_0 and μ_1 to the new value takes three pulses. Figure 4.4 shows the estimated μ_0 and μ_1 over time. The top graph in this figure shows that it takes several steps before the statistics are completely adjusted. The bottom graph shows the estimated amplitude of the recorded signal. Note that, as described in Section 2.2, the amplitude increases when more light is added. Table 4.3 shows the calculated errors. In all test cases, only two or three pulses were missed during the last transition from 1100 to 2200 lux.

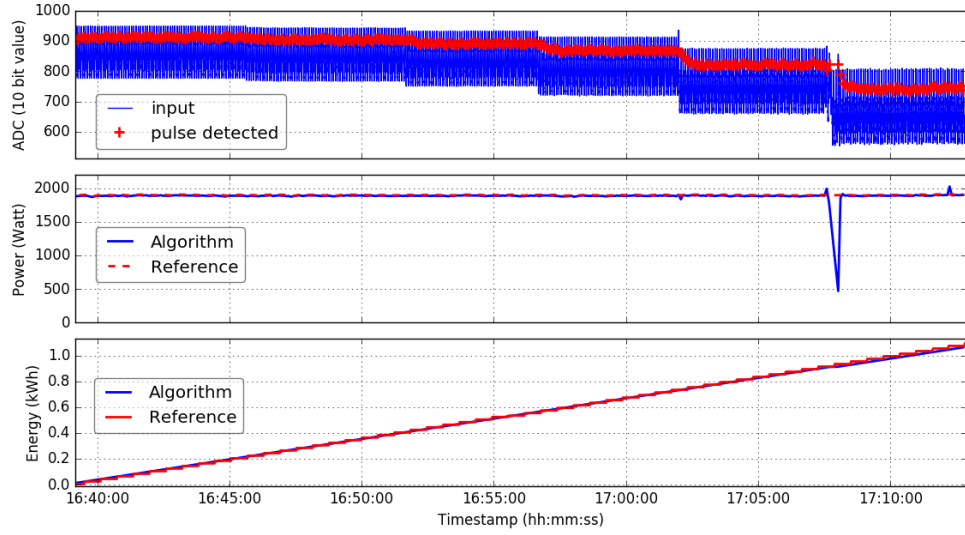


Figure 4.3: Increasing ambient light from 30 to 2200 lux while the load is fixed at 2 kW. Every 4 minutes two density filters are removed, resulting in a doubling of the light intensity.

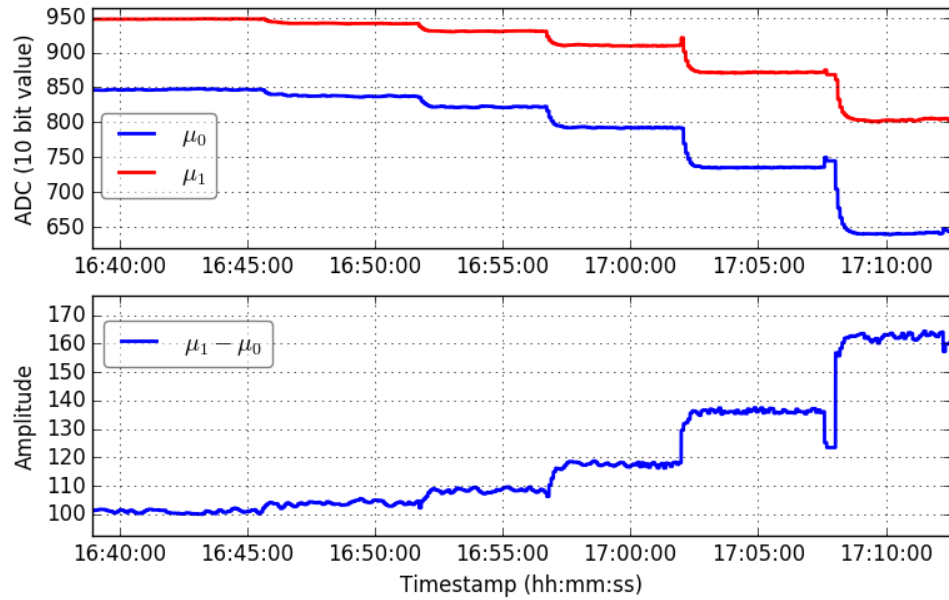


Figure 4.4: Estimated μ_0 and μ_1 corresponding to Figure 4.3.

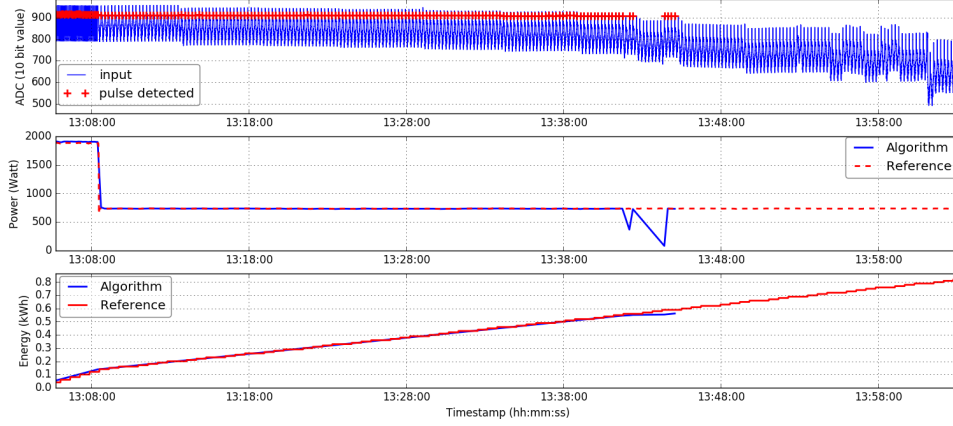


Figure 4.5: After updating the signal statistics at 2 kW in the first four minutes, the signal is set to 750 W, while the light intensity is varied between 30 and 2200 lux. Every 4 minutes one density filter is removed resulting in an increase in light intensity by approximately 43%.

In the next experiment the load is set to consume 2 kW for the first two minutes and then to 750 W. During the consumption of 750 W, the light intensity is varied between 30 and 2200 lux. Every 4 minutes one density filter is removed resulting in an increase in light intensity by approximately 43%. The matched filter is dimensioned at a power consumption of 2 kW such that only the first two minutes the signal statistics are updated. Figure 4.5 shows the algorithm output of the experiment performed with Meter-600. After removing the eighth filter, between timestamps 13:38 and 13:48, pulses are no longer detected. This is due to the fact that the old threshold values $\gamma_{\min}$ and $\gamma_{\max}$ determined at the first two minutes of the experiment are still used because the signal statistics are not updated during a power consumption of 750 W. Only if the power consumption is set back to 2 kW for a short period, the signal statistics will be updated, and pulses will be detected again. However, in practice the template used to estimate the signal statistics is chosen to match a wide range of powers that occur regularly, such that this situation will be prevented.

4.2 Household experiments

4.2.1 Test setup

In two households the Arduino based photo-reflector logger is placed in the meter closet, where an analog meter is located. In this particular setup we cannot use the Fibaro wall plug as a reference due to the fact that we do not have access to a plug conducting the main power signal. As an alternative, the household residents wrote down the energy consumption displayed on

Table 4.4: Meter properties households.

	C-value	Stripe color	Disk type	Sunlight
Household A	120	red	smooth	No
Household B	600	black	serrated	Yes

the analog meter every 4 hours during daytime for one week. Table 4.4 shows the meter properties used in the households. In household A the meter closet is closed by default and only opened when the owner wrote down the meter readings. In household B the meter closet door was completely removed to examine the influence of sunlight, which enters the room through a window.

4.2.2 Evaluation

Both household owners were asked to retain their usual electricity consumption pattern when the Arduino logger was placed. To validate the algorithm, the meter readings written down by the household owners are compared to the energy consumption calculated by the algorithm. Different from the lab experiments the matched filter, to update the signal statistics, is dimensioned to a power consumption of 300 W. This power consumption was found to occur regularly and therefore chosen. As an example, Figure 4.6 shows two graphs with the power and energy consumption over a period of 36 hours for household B. The top graph indicates the power consumption calculated by the algorithm, while the bottom graph shows the calculated energy consumption in comparison to the actual meter readings. The solid points in the graph indicate the moment in time a meter reading is performed by one of the household owners. Note that the power consumption lacks a reference since this information cannot be extracted from the meter readings. However, from Figure 4.6 it can be concluded that for this recording the algorithm follows the actual energy consumption quite well.

Table 4.5 shows the error in the measured energy consumption for each 24 hours of recorded data in both households and the total error over the full week. All errors are within the 5% error range, mentioned in Section 1.3, except the error of 10.4% at day four Household A. After examining the raw signal, it was concluded that no false pulses were detected, which could have lead to an overestimated energy consumption. The reason for this error is most likely the fact that the meter readings of Household A are written down without a fractional part. This makes the percentage error of Household A less valuable and should therefore only be seen as an indication of the performance of the algorithm. Household B wrote down the meter readings with one digit behind the dot, making the percentage error more reliable. Examining the raw signals revealed that no false pulses were detected over all recorded data of household A and B. However, some pulses were missed in the case of household A and B. For household A, around 2 pulses were missed

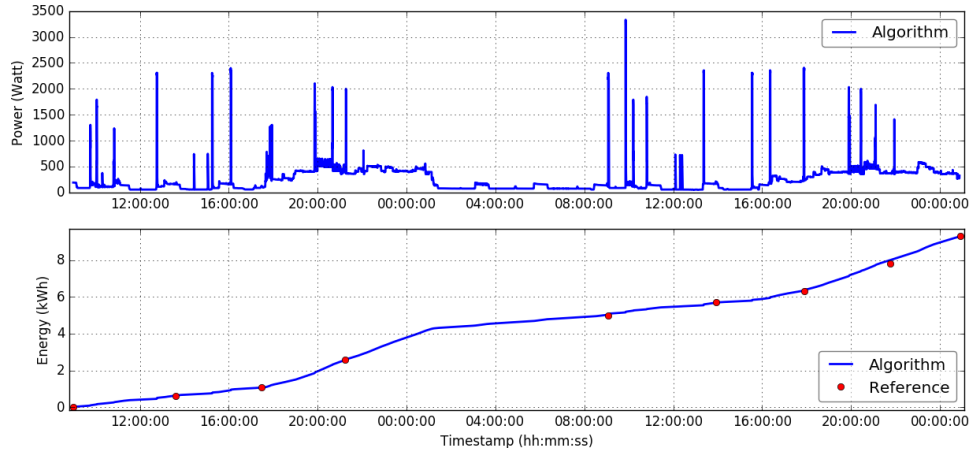


Figure 4.6: Estimated energy consumption of Household B.

in the short period that the household owner wrote down the meter readings and aimed a flashlight directly onto the electricity meter. The pulse misses in household B were introduced because the signal statistics were sometimes not updated for a couple of hours while the ambient light increased due to the sun rise. This corresponds exactly to the previous explained experiment performed in the lab. However, the influences of the changing ambient light happen so slowly that only 5 pulses were missed before the signal statistics were already updated. Looking at the total error after one week, it can be concluded that the algorithm meets the requirement of a percentage error less than 5%.

Table 4.5: Calculated energy-consumption error of both households.

	Household A			Household B		
	Algorithm (kWh)	Reference (kWh)	Error	Algorithm (kWh)	Reference (kWh)	Error
Day 1	7.90	8	-1.25%	4.30	4.3	0%
Day 2	12.17	12	1.41%	5.75	5.9	-2.54%
Day 3	6.96	7	-0.57%	5.27	5.3	-0.69%
Day 4	5.52	5	10.4%	5.33	5.4	-1.59%
Day 5	12.22	12	1.83%	5.23	5.2	0.71%
Day 6	8.33	8	4.12%	1.05	1.1	-4.55%
Day 7	8.91	9	-1.0%	5.03	5.0	0.6%
Total	62.01	61	1.66%	31.96	32.2	-0.75%

Chapter 5

Conclusion and Future Work

5.1 Conclusions

In this work, an energy-efficient pulse-detection algorithm is proposed that monitors the consumed energy (kWh) and power (W) indicated by analog electricity meters. Analog electricity meters have a disk that has a rotation speed proportional to the energy consumption. By observing this disk with the help of a photo-reflector sensor the energy consumption and power can be determined. Because of the physical properties of the disk, each captured revolution can be represented as a pulse signal that is 5% duty cycled. The company Quby developed a device (referred to as the *mains-powered meter module*) that is attached to an analog electricity meter and uses a photo-reflector to track the energy consumption and power. In order to increase their target customer group, Quby is interested in a battery-powered meter module. With these battery-powered meter modules electricity meters that do not have a power outlet nearby can be monitored. Two main aspects of the mains-powered meter module were revealed that have a big impact on the total power consumption. This includes the LED current, which is currently set to 10 mA, and the sampling rate of the optical signal, which is set to 10 kHz. One of the requirements for a future battery-powered meter module is that it runs for at least one year with an energy budget of 4200 mAh (roughly four AA batteries). The error percentage in the derived energy consumption should be less than 5%. To achieve energy efficiency a new pulse-detection algorithm is proposed, which is more robust to noise. As a result, the LED current can be reduced from 10 mA to 1 mA. Moreover, the sample frequency of the phototransistor is reduced from 10 kHz to a maximum of 400 Hz. In addition, the LED was duty cycled to 20% in order to save more energy. Note that, due to the reduced LED light, the sensor is more sensitive to ambient light. As a result, the signal statistics of the recorded signal vary significantly over time when electricity meters are exposed to ambient light. Our proposed algorithm continuously estimates the signal statistics and is

therefore robust to these changing conditions. Based on an approach where the sampled signal is continuously matched to a signal corresponding to a predefined frequently occurring power consumption, the signal statistics are estimated. These estimated signal statistics are then used in a multiple-sample likelihood-ratio test classifier to detect pulses in all possible power consumption ranges.

To evaluate the proposed algorithm, measurements are performed in a lab and at two households. The maximum error found in the lab test was -1.12%, and the highest mean error at the households after one week was 1.66%. In total, the algorithm was evaluated with 5 different meters, for a total of 350 hours. Both lab and household experiments have errors below the required 5% and show that the proposed method gives stable and reliable results in different realistic circumstances. There was one case where the signal statistics were not updated properly when the ambient light was changed abruptly in a lab environment. A possible solution to update the signal statistics more frequently, resulting in fewer pulse misses, will be discussed below.

Due to the reduction of the LED current, the most important part of the algorithm is the estimation of the signal statistics. This component gives the algorithm the ability to detect pulses while the signal statistics are varying over time. Without the correct signal statistics the multiple-sample likelihood-ratio test classifier will not be able to detect any pulses. The purpose of the classifier is to remove the impact of the noise by examining multiple samples, resulting in a higher probability of detecting a pulse.

The proposed pulse-detection algorithm is at least 50 times more energy efficient than the current pulse-detection algorithm developed by Quby. This factor 50 is based on the worst-case C-value of 800 rev/kWh. For lower C-values the improvement in energy consumption will be even higher since the sampling frequency of the phototransistor and the duty cycle of the LED can be further reduced. This implies that the battery-powered meter-module will be able to last -at least- one year with an energy budget of 4200 mAh.

5.2 Future work

Section 4.1.3 showed the limitation of the proposed algorithm when the template does not match for a long period. To cope with this limitation, research should be done to use a set of filters instead of one filter, where each filter is dimensioned to a different frequently-occurring power consumption. This would result in a more frequent update rate of the signal statistics.

A second area for improvement is that the algorithm is evaluated with an LED current set to 1 mA while 20% duty cycled. Most likely the LED current can be reduced even further, i.e. looking to Figure 4.2 pulses are still clearly present in the signal.

Another possible research area is to dynamically adjust the LED current during run time. The sensitivity index, in Equation 2.13, can be used to determine the distance between two normal distribution functions. This sensitivity index should be applied to the two approximated normal distribution functions used in the ‘multiple sample likelihood ratio test’ component, presented in Section 3.3. A small value for the sensitivity index implies the distribution functions are close to each other, meaning it will be hard to differentiate pulses from noise. Increasing the LED current will increase the SNR as shown in Section 2.2. This most likely also implies the distribution function will shift apart, which results in an increased value of the sensitivity index. Therefore, we propose to relate the LED current to the sensitivity index as follows: a small sensitivity index means the LED current needs to be increased, a high sensitivity index indicates the LED current can be decreased. For example, using this relation the LED current could be automatically set to zero in the cases where ambient light introduces enough reflected light from the disk to differentiate between pulses and noise.

A fourth research area is that during our experiments, the LED was duty cycled at 20% between zero and 1 mA. Ongoing research at the Embedded Software Group at the University of Delft shows that when an LED is duty cycled between the on and off state, it could be more energy efficient to keep the LED at a low current level close to zero instead of fully disabling it. As a result, the rise and fall time could be shorter, meaning an even lower duty cycle could perhaps be applied.

Finally, before the algorithm goes to market, more field tests need to be performed, since only two different households were examined. Also, the lab test can be expanded to a test setup performed in an outside environment influenced by real sunlight instead of a simulator.

Bibliography

- [1] European Commission, “Smart grids and meters.” <https://ec.europa.eu/energy/en/topics/markets-and-consumers/smart-grids-and-meters>, May 2017. [Online; accessed 3-May-2017].
- [2] C. F. Covrig, M. Ardelean, J. Vasiljevska, A. Mengolini, G. Fulli, E. Amoiralis, M. Jiménez, and C. Filiou, “Smart grid projects outlook 2014,” *Joint Research Centre of the European Commission*, 2014.
- [3] European Commission, “Cost-benefit analyses and state of play of smart metering deployment in the EU-27 - accompanying the document report from the commission benchmarking smart metering deployment in the EU-27 with a focus on electricity,” 2014.
- [4] “Smart meter explained - smart meter roll-out.” <http://www.which.co.uk/reviews/smart-meters/article/smart-meters-explained/smart-meter-roll-out>. [Online; accessed 3-May-2017].
- [5] F. Leferink, C. Keyer, and A. Melentjev, “Static energy meter errors caused by conducted electromagnetic interference,” *IEEE Electromagnetic Compatibility Magazine*, vol. 5, no. 4, pp. 49–55, 2016.
- [6] ECN, Energie-Nederland and Netbeheer Nederland, “Energietrends 2016.” <http://energietrends.info>, September 2016.
- [7] Centraal Bureau voor de Statistiek, “Huishoudens; samenstelling, grootte, regio, 1 januari.” [http://statline.cbs.nl/Statweb/publication/?VW=T&DM=SLNL&PA=71486ned&D1=0-2,23-26&D2=0&D3=0,5-16&D4=\(1-1\)-1&HD=170504-0916&HDR=T,G3&STB=G1,G2](http://statline.cbs.nl/Statweb/publication/?VW=T&DM=SLNL&PA=71486ned&D1=0-2,23-26&D2=0&D3=0,5-16&D4=(1-1)-1&HD=170504-0916&HDR=T,G3&STB=G1,G2), August 2016. [Online; accessed 3-May-2017].
- [8] Department for Business, Energy and Industrial Strategy, “Smart meters, quarterly report to end december 2016, great britain,” *Smart Meters statistics, Low carbon technologies, and Household energy*, March 2017.
- [9] Netbeheer Nederland, *P1 Companion Standard - Dutch Smart Meter Requirements*, 5.0.2 final ed., February 2016.

- [10] W. Gillberry, "System for remote meter viewing and reporting," Feb. 9 1999. US Patent 5,870,140.
- [11] S. V. Gubbi and B. Amrutur, "Adaptive pulse width control and sampling for low power pulse oximetry," *IEEE transactions on bio-medical circuits and systems*, vol. 9, no. 2, pp. 272–283, 2015.
- [12] Z. Yuan, J. H. Kim, and J. D. Cho, "Adaptive sampling for ECG detection based on compression dictionary," *Journal of Semiconductor Technology and Science*, vol. 13, no. 6, pp. 608–616, 2013.
- [13] K. H. Ng, C. F. Yeong, E. L. M. Su, T. Y. Lim, Y. Subramaniam, and R. S. Teng, "Adaptive phototransistor sensor for line finding," *Procedia Engineering*, vol. 41, pp. 237–243, 2012.
- [14] A. V. Oppenheim and G. C. Verghese, "Signals, systems, and inference - class notes for 6.011: Introduction to communication, control and signal processing," 2010. MIT OpenCourseWare.
- [15] "Wikipedia: Location scale family." https://en.wikipedia.org/wiki/Location%E2%80%93scale_family. [Online; accessed 24-May-2017].
- [16] N. A. Macmillan and C. D. Creelman, *Detection Theory: A User's Guide*. Lawrence Erlbaum Associates, 2nd ed., 2005.
- [17] "Pennstate eberly college of science - probability theory and mathematical statistics." <https://onlinecourses.science.psu.edu/stat414/node/167>. [Online; accessed 3-April-2017].
- [18] S. W. Smith, *The Scientist and Engineer's Guide to Digital Signal Processing*. California Technical Pub, 1st ed., 1997.
- [19] "Wikipedia: Normal distribution, moments." https://en.wikipedia.org/wiki/Normal_distribution#Moments. [Online; accessed 26-April-2017].
- [20] Fibaro Systems. <https://www.fibaro.com/en/products/wall-plug>.
- [21] Fibaro Systems, *Fibaro Wall Plug - Operating Manual*, v2.0 ed., January 2016.
- [22] GTI Graphic Technology Inc. <http://www.gtilite.com/products/gle-m-color-matching-luminaires/>.
- [23] "Wikipedia: Lux." <https://en.wikipedia.org/wiki/Lux>. [Online; accessed 29-May-2017].

Appendix A

Different C-values

In this work over 300 pictures of electricity meters are examined. The pictures were made by volunteers living across Europe. By manually inspecting these pictures the C-values are determined. The following table shows the number of occurrences for each C-value from the inspected pictures and Figure A.1 shows the corresponding histogram.

Table A.1: Occurrences of C-values in Europe, derived from over 300 pictures.

C-value (rev/kWh)	Occurrence	C-value (rev/kWh)	Occurrence
50	2	250	16
70	1	278	6
75	108	300	1
96	40	375	30
100	1	384,5	1
120	3	400	36
125	2	416	7
143	1	500	2
150	8	556	13
166	5	600	7
187,5	1	625	3
200	8	800	4
240	1		

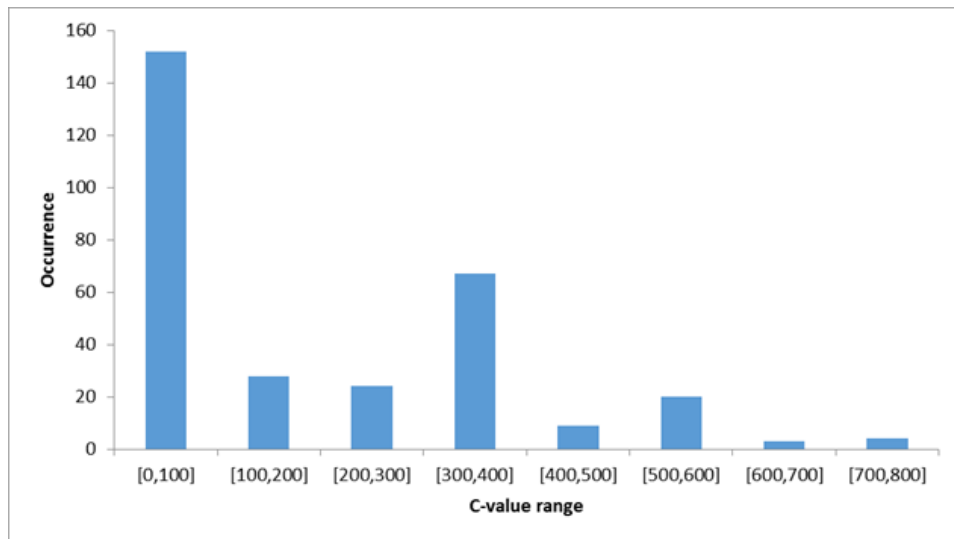


Figure A.1: Histogram of the occurring C-values in Europe, based on Table A.1.