UCLouvain

epl

École polytechnique de Louvain

Automating tool for fault analysis of washing machines

Towards predictive maintenance of home appliances

Author: **Mieszko Di Gregorio Botek**
Supervisors: **Jean-Pierre Raskin, Sébastien Toussaint**
Readers: **David Bol, Denis Joassin, Paul Fisette**
Academic year 2021–2022
Master [120] in Electro-mechanical Engineering

Acknowledgements

First of all, I would like to thank my advisor Jean-Pierre Raskin and my supervisor Sébastien Toussaint who gave me complete freedom in the direction of the work despite the non-standard approach of the project. I also want to deeply thank my fellow students Justine Lebrun and Antoine Percy, who performed their master thesis in the same framework as mine for their constant support, ideas, and great reviews of the work.

I want to give my thanks to Paul Fisette and David Bol to be members of my jury. A special thanks to Denis Joassin for also being part of my jury but also for the great and meaningful guidelines. The ShaRepair assembly, more particularly Jan Larosse and Tine Van Moeseke from MaakBarLeuven who helped in giving a structure and directions in the early stages of the projects.

Thanks to the Openhub and Makilab for providing me a place to work and perform my experiments with the best tools I could use. I was able to perform many tests on a washing machine without sacrificing the one used at my place thanks to the group Fnac Darty who kindly provided the washing machine used for this work.

I want to thank my family for always supporting me and always challenging me to make progress in my reflexions. Lastly, I want to deeply thank my girlfriend, Céline Dieudonné for always helping me in the best but also the worst moments and always giving me the best advice also during the long nights needed to end this thrilling project.

Abstract

The number of disposed washing machines nowadays increases at an alarming rate due to their repair complexity. Therefore, the need for efficient and automated fault analysis and predictive maintenance systems becomes crucial. For now, only a handful of projects are trying to resolve this matter but most of them are either using complex or expensive systems. Those setups are not efficient for neither the collection of large datasets nor the deployment.

This is where this master thesis provides a physical tool to facilitate such a process. Using accessible and low-cost hardware, this work aims at furnishing a tool to allow further projects to implement machine learning for failure analysis and predictive maintenance. The device developed is based on a microcontroller and an additional microcomputer acting as a server to store the data captured by two inertial measurement units. Each unit samples acceleration and gyroscopic data below 250Hz from the top and the side of the outer shell of a washing machine. The steps in building and developing the device are presented to allow replication and improvement.

The collected and stored data from the device, allowed a first manual detection of multiple failures and anomalies on a washing machine only using few processing steps such as the Fast Fourier Transform. Further guidelines on an improved version and a complete deployment of such a device via the citizen science framework are also presented.

Contents

List of Figures

List of Abbreviations

ADC .... Analog to Digital Converter

API .... Application Programming Interface

BOM .... Bill of Materials

BPFI .... Ball Pass Frequency Internal race

BPFO .... Ball Pass Frequency Outer race

BSF .... Ball Spin Frequency

CSV .... Comma-Separated Values

EU .... European Union

FFT .... Fast Fourier Transform

FTF .... Fundamental Train Frequency

GEM .... Global E-waste Monitor

GMF .... Gear Mesh Frequency

GND .... Ground

IMU .... Inertial Measurement Unit

IoT .... Internet of Things

JRC .... Joint Research Centre

LSB .... Least significant Bit

OS .... Operating System

PCA .... Principal Component Analysis

PCB .... Printed Circuit Board

PVC .... Polyvinyl Chloride

RAM .... Random-Access Memory

SRAM .... Static Random-Access Memory

RBPF .... Rotor Ball Pass Frequency

RMS .... Root Mean Square

RPM .... Rotations Per Minute

RUL .... Remaining Useful Life

SD .... Secure Digital

SSH .... Secure SHell

TRL .... Technology Readiness Level

Chapter 1

Introduction and motivation

From the Global E-waste Monitor (GEM) 2020 report [1], the world has generated more than 56.3 million metric tons (Mt) of E-waste until 2020. This is an increase of 9.2 Mt compared to the same report done in 2014. The fast growth of innovation and accessibility in the electronics domain combined with the consumer society has led to this absurd number. In this poll, 13.1 Mt are from large appliances such as fridges, washers, and dryers. A report [2] performed by the Joint Research Centre (JRC) of the European Union (EU) showed that almost 70% of washing machines are thrown away due to a defect in Germany. This report also highlighted that grouping many studies around the world, 50% of thrown washing machines are used for less than 10 years and only 20% of them are used for less than 5 years.

One way of limiting the amount of E-waste is by centering the design around the concepts of robustness, upgradability, and repairability. Unfortunately, this mindset is not completely in line with the current society. Therefore, only a few projects are put in place in order to work on fully capable and robust devices. This is the case of the washing machine *L'Increvable*[3] which is designed to last for decades, to be robust and easy to repair.

Another way of reducing that tendency is by repairing those appliances. Traditionally, diagnoses and repairs are performed from the beginning to the end by humans. A larger amount of time is spent on diagnosing the root of the failure to perform the proper repair rather than on the repair itself. In most cases, this diagnosis time is the big wall between throwing away and repairing. Even if the source of the failure is located, irreversible failure may already have occurred, leading to a broken machine and a loss of time.

Hence, there are many factors, not always logical that make consumers choose to replace a broken product rather than repair it [4].

On the side of repair centers, it is easier to target smaller appliances as they are movable and easier to repair. For larger appliances like fridges, dishwashers, and washing machines repairs are mostly done by manufacturers or third-party sellers. Nevertheless, all repair centers follow the same traditional system of manually searching the failures and their reasons. The luckiest centers may have access to the service manual and the error codes but those are only as good as the constructor designed them.

Constant monitoring for failure analysis has been in place for some time in large industries as shown in [5]. This is due to the high loss led by downtime and repair of complex systems. Nowadays, many medium-sized manufacturing companies are joining the flow.

With the rise of the Internet of Things(IoT), many appliance manufacturers stuff more electronic devices into their products. Some of them even allow better monitoring for fault diagnosis. However, the capability and control left to the user are always limited and the tools are proprietary. In the end, the main advances are in terms of the accuracy of error codes appearing when a problem occurs. Unfortunately, they only appear when it is too late. These error codes are mostly based on basic sensor measurements in the machine due to the trade-off between cost and information. It is more financially, and in terms of complexity, interesting for manufacturers to reduce the number of sensors.

This is where this master thesis aims to contribute to the concept of repairing and spotting failures in washing machines by applying failure detection and predictive maintenance.
With the help of IoT, the possibility to monitor, assess and make decisions based on proper information is more accessible than it ever was. Many open-source systems such as the ESP32 [6] or the Raspberry PI Zero 2W [7] allow great monitoring performance for low consumption and very low price. Moreover, with the expanse of big data and cloud computing, better and more complex models can be implemented to accurately locate and even predict a failure.
The point has been proven in the industry [8][5] but never introduced to the citizen sector. Of course, such concepts are not relevant for small and elementary devices such as coffee machines or blenders but on larger, more complex, and longer used tools the interest is present. The point of entry in the citizen sector is via the citizen science framework.

This master thesis will serve as a first entry point to develop such a module capable of measuring and storing vibration data of washing machines to apply failure analysis and predictive maintenance in a non-invasive way. To seek such results, Chapter 2 will introduce the literature on washing machines, fault analysis, and predictive maintenance concerning rotating machines. As the main leading thread, the vibration will be at the center of all these topics but some order signals will be introduced.

Right after, Chapter 3 will briefly introduce some fundamentals in terms of fault analysis of rotating machines via vibration analysis.

Chapter 4 will bring the entire scope of the project as well as the scope of this master thesis. An introduction to specifications of the targeted device with a first functioning diagram will also be presented.

The next portion, Chapter 5, is going to walk through the construction of the developed device, from iteration to iteration, giving the major challenges and the shortcomings until achieving a satisfactory result. A few steps will be further developed to show the reader what was or was not tried and why some choices were made. This section is intended to be a chronological view of the development to guide anyone willing to reproduce the device, improve it, or implement a similar setup with the best tracks to continue further and wiser.

Chapter 6 will describe the actual tests and measurements performed on a washing machine to study some different use cases. This chapter is meant to show the potential of such a device in an exploratory manner. A small fault analysis of the washing machines is performed to prove the point but by no mean a complete processing method of fault detection since it is an entire subject in itself.

Afterward, Chapter 7 will provide a relative retrospective insight concerning the work done to extract some guidelines for further development and testing.

Finally, Chapter 8 will display the perspective of this project with the results in mind and expose what lies next from the different roads available to the potential in use cases.

Chapter 2

State of the Art

In this section, a brief literature review of what has already been achieved in various domains similar or relative to this work is presented. This review is guided by finding how washing machines' faults can and are detected and what are the limits of those detection systems.

2.1 Fault analysis on rotatory machines

The concept of fault analysis was developed in the 60s [9] via the Fault tree analysis used in complex systems such as in the aerospace and aeronautical sector. This method was developed to quickly find any error on large systems. The industrial sector then followed the trend to reduce downtime in production lines. However, with the advancement and accessibility of electronics capable to sense data, the domestic sector is a non-negligible market for fault analysis.

The first step for fault analysis is monitoring. This is done by placing sensors in the target system. Once enough data is gathered, it is necessary to process the data to extract some tangible metrics. These metrics are then compared to some predefined threshold values allowing us to measure the state of the system.

In the case of washing machines, there have been some setups to collect data to detect faults. Recurrent inputs are often used as monitoring signals in the literature. The most common inputs for this kind of system are current, water usage, sound level, and vibration signals.

As an example, a German team [10] measured the sound level to detect faults using a smartphone microphone. Using some processing and machine learning, they developed two types of algorithms, one in the temporal domain and the other in the frequency domain they managed to show significant variation between healthy and faulty machines as shown in Figure 2.1. The main problem according to the source, is the complex calculation leading to a long processing time which is not suitable for real-time analysis. Even so, what makes this setup quite interesting is the ease of reproduction due to the straightforwardness of the microphone and the non-invasivity of the system. Those characteristics are big constraints that add more value to the results obtained. Yet, the microphone needs to be as close as possible to the rotatory part to output such results.

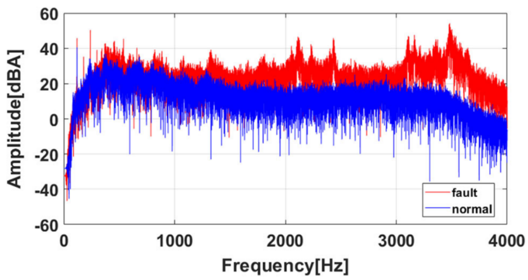


Figure 2.1: Measurement of the sound between a healthy and faulty washing machine using a smartphone microphone from [10]

Using a more complex setup, [11] used accelerometers and acoustic transducers to compare the effect of loading the machine. The results allowed to differentiate an empty running machine from a loaded one using individually both sensors as displayed in Fig 2.2. These results presented in this article show the abundance of information present in frequencies bellow 250Hz.

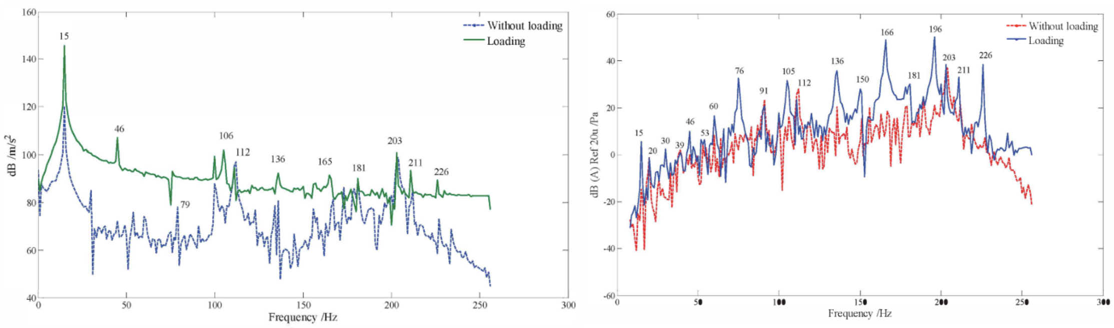


Figure 2.2: Spectral density from acceleration measurements on the left and acoustic measurements on the right with and without loading from [11]

Another example of a non-invasive technique was developed in [12] using a laser Doppler technique and a likelihood classifier. From their experience, the range of interest of the vibration with structural frequency response goes from 0Hz to 200Hz. Using various measurements and processing methods, they found out that peaks at odd harmonics classified by likelihood methods yield the best classification results. The team behind the article managed to classify four types of machine defects which are defects in the electric motor clamping screws, the release of the shock absorber, use of different types of springs, and distortion of the pulley from one another with more than 90% of accuracy. The main drawback of this example is the need for high-end gear which is not fit where a washing machine is normally placed. To obtain such performances, the machines had to follow a strict rotating pattern and speed which are non-standard for less than 20 seconds.

Continuing on the vibration signals, [13] managed to classify three types of machines (healthy, with a problem on the clamping screw at the motor, and with counter-weight distorted) with an 87% of success rate with a single point laser vibrometer at a sampling frequency of 2kHz in different points of the machine. After the measurements, the feature is extracted by a Wavelet transform[1]. As for the previous setup, the system needed is quite bulky. Contrarily to the previous case, this paper analyzes data for more than 20 seconds but at specific moments of normal washing machine cycles.

With another objective in mind, [15] and [16] managed to determine the position and weight of a load in the drum of washing machines within a good range using two different approaches. The first one uses two laser displacement sensors and two accelerometers (3 axes each) on the outer part of the drum and four load cells, one on each foot. With some processing and a trained neural network, they managed to compute the position of a mass below 2kg within a 300g deviation and within an axial accuracy of 5cm at 150 RPM. The second article only focuses on finding the angular position (compared to a rotating reference axis) and the weight using as few sensors as possible. The sensors used are a piezoelectric sensor to measure the vibration and a hall sensor with a magnet to get the reference rotating angle. From measurements at 200 and 250rpm, they got an almost linear evolution of the peak values from the sensor placed to measure the horizontal displacement with the increase of the mass.

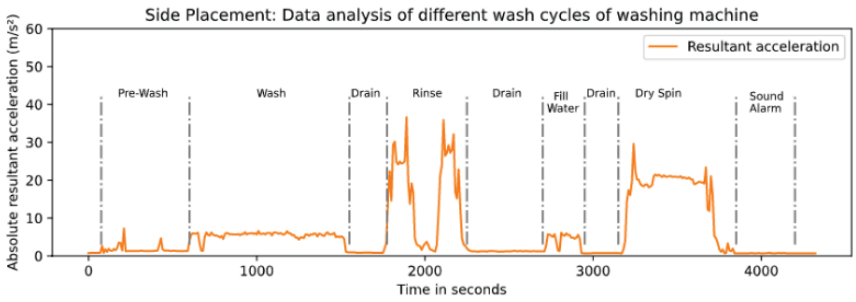
For another field of study, yet providing good vision, [17] placed a 9-axis Inertial Measurement Unit (IMU) containing an accelerometer, a gyroscope, and a magnetometer in a washing machine to measure the acceleration felt by textile. This study gives a lot of insights into the accelerations and movements inside the system as well as the different phases of the cycles of washing machines.
From all the information provided earlier, the choice of focusing the measure of vibration via an accelerometer is pertinent to detecting faults. The main problem of most of these examples is that the setups are resembling too much of something done in laboratories and are not feasible at large scales. They do not explore the possibility of wireless interfacing. In this direction, [18] managed to develop a laundry monitoring system using a low-cost accelerometer. They sure do not make any fault analysis but manage to accurately capture the different stages of the washing machine cycle as shown in Fig 2.3 using relatively accessible technology.
Due to the complexity of the system, purely physical-based models are often not enough for failure detection hence the use of machine learning is needed.

2.1.1 Data driven Failure detection

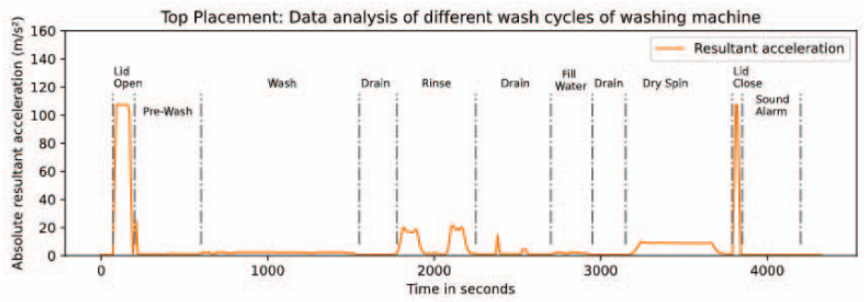
Many non-analytical methods are available for failure detection. This is the case in the popular domain of machine learning. Few algorithms are prominent in the literature as further analyzed in [19] and [20] for first-level fault analysis.

The main challenge in utilizing such methods is the lack of data. To build a robust algorithm, there must be a significant amount of data on healthy machines but also faulty ones. In this case, there is a need to recognize and label various failures, and the minimal amount of samples for each fault needed is drastically

[1]The wavelet transform is a transform similar to Fast Fourier Transform (FFT) to provide frequency information but on transient signals. More information about the Wavelet transform is available in [14].



(a)



(b)

Figure 2.3: Monitoring of the different washing cycle by [18] using two low cost accelerometers on the top and the side of top loaded washing machines

increased. Moreover, this data then needs to be separated into training and testing data as deeply explained in [21]. This is where the methods found in [22], [13] and [23] are coming up short. The prerequisite to developing a robust machine learning algorithm is designing a capable method of collecting data at a large scale. Some examples of deployments are made in [24] and [25] and will be discussed in Section 2.4.

Before proceeding further on the topic of fault analysis, it is necessary to have a common ground about the working principle of the machine. It is by knowing the target system and its principle types of failure that fault monitoring can be developed.

2.2 How, when, and what for washing machines

This section will first expose the basic mechanisms in front-loaded washing machines and present the different failure modes observed by statistical research. There will not be any discussion on top-loaded machines since they are not the target of this work.

2.2.1 How it works

Front-loaded washing machines are driven most of the time by single-phase induction motors with two poles. This motor is the main source of vibration and power consumption but is not the only motor present in a washing machine. As guested, the flow of water entering and leaving the washing machine is controlled via a pump.

To rotate the drum in the machine, a belt is attached to the motor and a wheel fixed to the drum as shown in Fig 2.4. The motor is tightened to the drum of the machines. To absorb most of the vibration, the entire system is maintained radially by 2 springs from the top and 2 dampers from the bottom. To maintain the machine stable, a block of concrete is placed on top of the drum. This gives higher inertia to the machine to avoid any lifting from the ground. Some machines have complementary springs axially to add rigidity. The last main moving mechanism in a washing machine is the locking and unlocking of the front door.

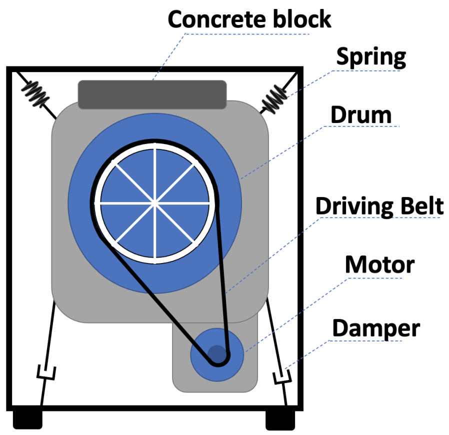


Figure 2.4: Basic schematic of the main unit in front-loaded washing machines

For each program, there is a series of defined sequences sent to the different components of the washing machines. They go from closing the front door, controlling the water pump, and managing the water heater to controlling the speed of the motor.

Many types of cycles are available in washing machines each with its own configuration. To illustrate the proportion of phases, Figure 2.5 shows roughly the amount of time the drum is spinning in a washing machine. These values depend on the machine, therefore only the ratios are to be taken into account. In practice, the amount of time where the drum is rotating at the selected maximal speed is not at all the prevalent part (less than 25%).

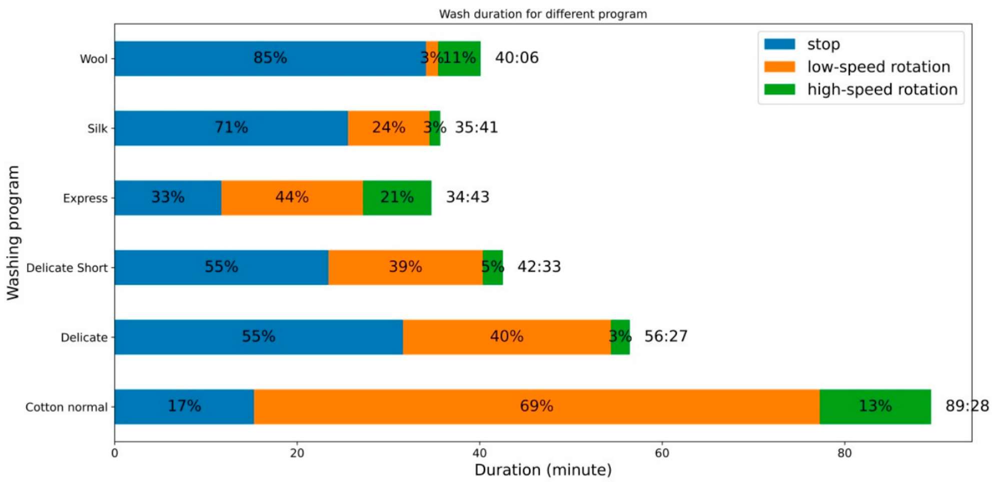


Figure 2.5: Washing machine speed distribution for different programs, from [17]

2.2.2 What to look for

From the report by the JRC [2], 50% of disposed washing machines have a lifetime below 10 years old. A factor positively influencing the lifetime of washing machines is the price. More expensive machines tend to have a longer lifetime. Among all the disposed washing machines, 70% are due to some failure. A classification of the various failures is visible in Figure 2.6.

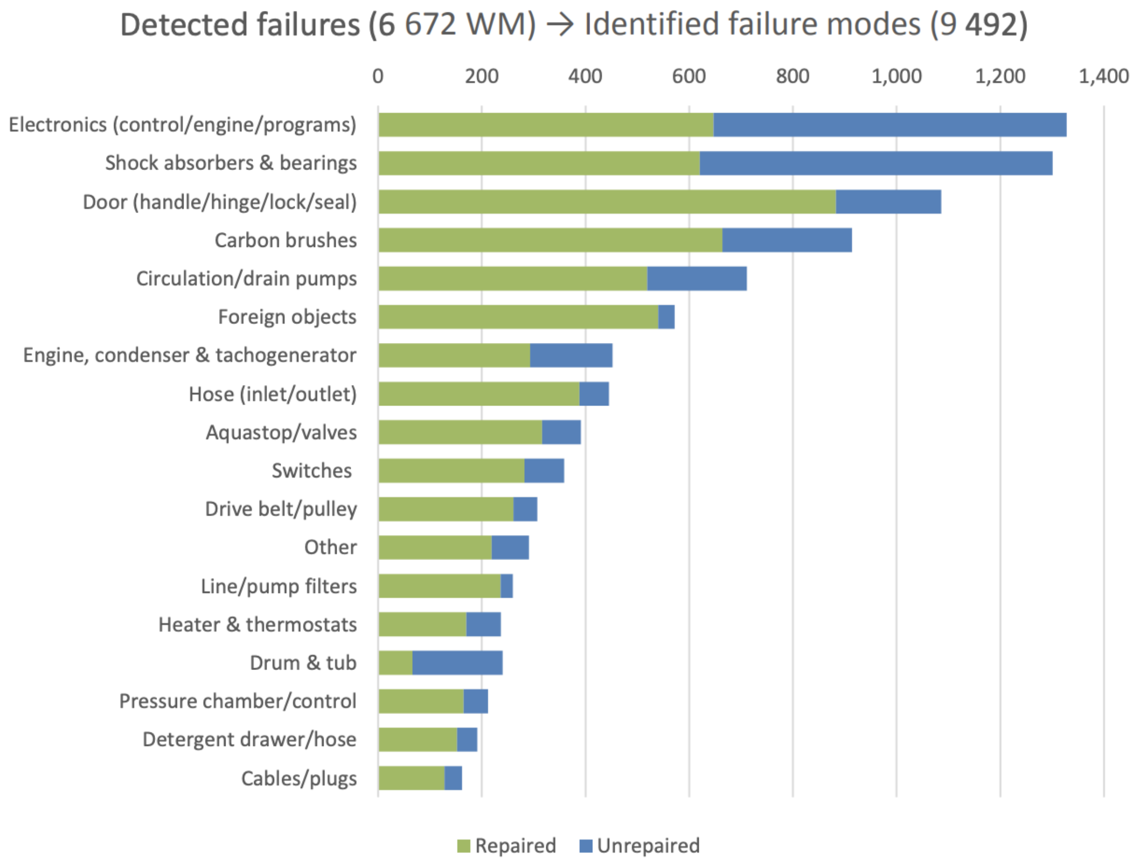


Figure 2.6: Different failure modes detected of washing machines from a survey by [26].

This figure is extracted from another report of the same group [26] . The figures permit highlighting the most common failures. Failures sources that may be visible via vibration detection systems are the shock absorbers, the bearings, the foreign object in the drum, the drive belt/pulley, and the drum. Combining those shows a significant amount of failure to detect. The discussion on the failure statistics will not be further talked about in this master thesis. For further information, the reader is recommended to browse the documents [2], [26] and [27] which go deeper into the subject.

2.2.3 When to act

A washing machine is a complex system, and therefore its failure rate follows the Bath-tub curve (see Figure 2.7) with three phases. The first one corresponds to infantile failure due to low-quality control. The second is the moment when the system is designed to be most used and contains less risk of failure considering an appropriate use. The last phase has an increase of failure rate due to wear.

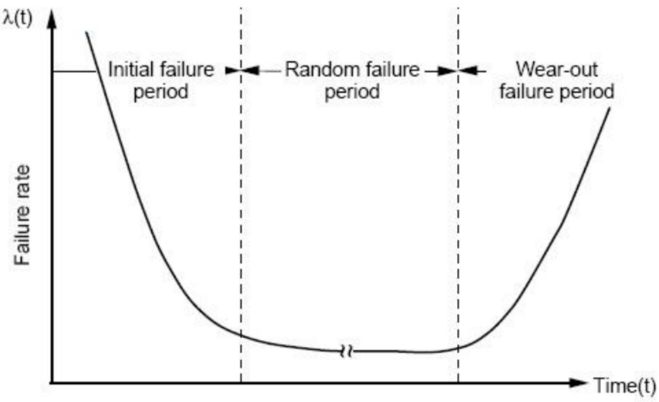


Figure 2.7: Bath-up curve representing the failure rate of systems along their lifetime from [2]

Since the stage at which washing machines are used, it is incoherent to focus on infant failure because the product is tested and certified before being sold. Therefore the range of action should be from the random failure period to the wear-out failure.

Among the random failure range, it is coherent to focus on a more general failure analysis since specific failures do not tend to appear. Given a significant amount of data and some features to look at, anomaly detection algorithms based on machine learning are the perfect tool for this part. They do not give the source of the failure but they are capable to notice early symptoms which is a good starting point.

Another approach to this random failure period would be to focus on the misuse of the machine. Few parameters are found in this category. First, there is overloading of the machine, which is considered one main case of early wear and failure on the bearings, springs, and dampers. Second is the imbalance and over-concentration of weight in the drum. This happens when objects not supposed to be washed in washing machines are put in such as non-washable shoes, wallets, keys, coins, or even phones. Lastly, the not leveled feet of washing machines are often an overlooked source of high vibration and risk of failure. All these failures can be monitored and prevented by looking at the right indicators. Some indicators are shown in the literature on rotating machines [28] but they are relevant for rotating machines performing steady-state works. Many of these metrics lose meaning when applied to data with various transient states.

The last period when wear is preeminent is the target moment for failure analysis and predictive maintenance. This is where most known failures from Figure 2.6 occur.

2.3 A step further than analyse, prediction

Until now, the discussion centered around fault analysis and detection. This is mainly used in reactive maintenance where the machine gets repaired only once a failure appears or is close to rising. There is no notion of scheduling the repair. This model is not viable in case of the system is complex to repair or where a small failure may result in huge losses (not only economically). To avoid such losses, preventive maintenance is traditionally applied. This is when there is regularly scheduled maintenance to avoid any failure. To ensure that there is no failure the scheduling must be performed regularly and therefore the amount of time and money spent is not optimal at all

Hence the need for predictive maintenance where downtime and maintenance scheduling are minimized while the use time is maximized. This is done by computing the Remaining Useful Life (RUL).

The estimation of RUL on a machine is done in four parts as shown in Figure 2.8. First, the collection of the raw data (Section 1) from the element to be monitored is performed. Second is the extraction of a metric serving as a health indicator (Section 2) which is relevant for the monitoring. The third is the setting of a threshold indicating (Section 3) a failure point. And fourth, is the development of some functions to perform the estimation of the RUL (Section 4) from the current value of the health indicator and the threshold value. In theory, these steps seem easy but in practice, it is complex to compose these values and functions due to the lack of data and the complexity of the system. As certainly noticed, the first two steps of computing the RUL are the same as the steps of failure analysis. This is why failure analysis was presented before predictive maintenance since it is an important step before predicting.

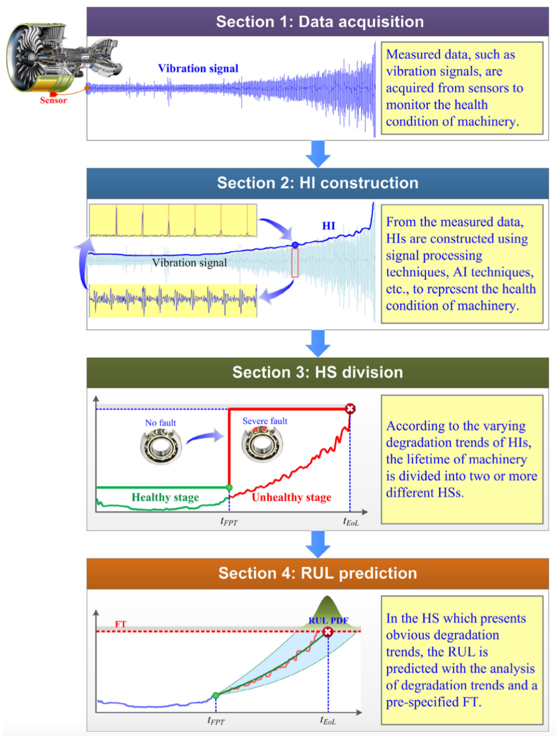


Figure 2.8: Four technical processes in a machinery health prognostic program from [29]

The document [29] gives a complete review of the state of the art in the domain of predictive maintenance. Some more insights on common cases are also given. One of them concerns bearing RUL which is, as seen previously, the main source of failure in washing machines. The document also gives insights into the different approaches to the topic.

The previous system is a model-based one, where the actual physics is the center of the algorithm. As for failure analysis, a more complex algorithm can be used to compute the state of RUL with the indicators using a machine learning-based algorithm.

To ease the task for the machine learning algorithm (meaning to reduce the training time and the model complexity), it is often necessary to perform some feature engineering on the data. This is done by preprocessing the data to extract the most relevant features. It can be performed using a Principal Component Analysis (PCA) and/or use the physics knowledge and use for example the previously build indicators. This will ensure a high-performance predictive maintenance computation and results.

For more information, a more complete review of the use of machine learning concepts in predictive maintenance is found in [20] where many important concepts are explained. A complete and easy to understand course on machine learning can be found in [21].

As the reader may have realized, the matter previously covered relies on one big assumption, the ease of data collection. This is the main reason why many fields are not using predictive maintenance or even failure analysis. However, with the revolution of big data and IoT this is no more a problem.

2.4 IoT and citizen science - a nice combo

Thanks to the IoT and the accessibility to electrical modules, it is now easier than ever to measure physical quantities and to use the data to make accurate decisions. But the system can not simply work using sensors working chaotically, this requires implementing smart architectures to be able to gather and use the data.

An example of such architecture is done in the Laundry Monitoring system [18]. This system uses one microcontroller fixed to a washing machine and connected to a Raspberry Pi acting as a router for a whole laundromat to transmit the data for online visualization. The information is sent via WiFi.

Another similar architecture is used in [24]. That work is part of the SMART-PDM European project [30] that took place between April 2019 and February 2022. This project's goal was to apply predictive maintenance solutions in various areas. In this use case, they monitored and collected data on electric consumption on large appliances to reduce the power consumption. The actual architecture of the SMART-PDM use case on washing machines is shown in Figure 2.9. It uses a smart power socket pluggable on the wall to gather the data. This data is then sent to a cloud server to be processed and used. The meaningful data is inputted in a REST API where a graphical user interface is developed to be given to everyday citizens.

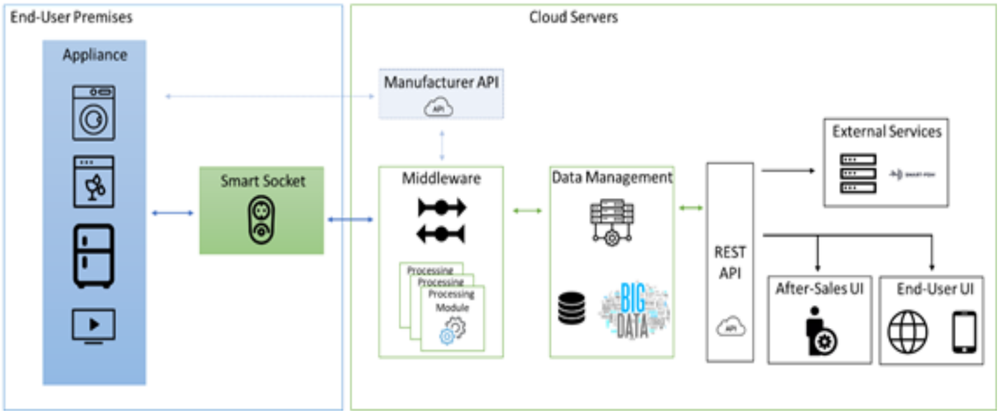


Figure 2.9: Architecture used in the project SMART-PDM for the use case of home appliances

With such setups, the amount of data needed to build and refine a machine learning algorithm for failure analysis and predictive maintenance uses is achievable. However, resources on the project are nowhere available or usable due to a lack of complete documentation.
In terms of machine learning for predictive maintenance using low-cost technology, [23] provides a clear insight into the whole system from collecting to alert notification.

All the previous examples show that the technical capability of developing a functioning architecture is present and accessible. However, there is still no environment allowing the willing deployment. This is why this project is meant to be included in the citizen science framework. This whole campaign where the citizen is not only the mean but also the actor in knowledge spreading is a key factor for the proposed project presented in the Chapter 4.
A great example of such an environment is the Sensor.Community platform [25]. This platform is a technological network at an international level to allow the development and deployment of environmental data measurement systems. Via the citizen science market, the project managed to deploy in 75 countries and has more than 14.000 active sensors worldwide.

With this last example, the literature needed to understand the project is completed. It is now time to enter the technical part of the literature susceptible to be used in this master thesis on fault analysis.

Chapter 3

Vibration of rotating machines theory

This chapter will introduce most of the mechanical-related terms and concepts to give a proper sense of the following of the work. Most of the theory comes from the following documents: [31], [32], [33], [34] and [35]. All this theory is developed around rotating machines working at a steady state or studied during one known transient. Therefore in the case of washing machines, this theory is still useful but loses some meaning in terms of physical explanation if not carefully handled.

3.1 Vibration origin

Mechanical vibrations are oscillatory deformations of a system. They are defined around an equilibrium point and are synonyms for energy transmitted in the system. The study of mechanical vibration can be done in stationary or dynamic systems. In this master thesis, we will only consider dynamic (with moving parts) mechanical vibration since the use case concerns a rotating machine.

Based on the vocabulary of the international standard ISO 2041 [36] and illustrated by [31] visible in Figure 3.1, fundamental vibrations are classified as deterministic or random (with a gray area in between). Among the deterministic vibrations, they are classified by periodicity.
A harmonic vibration is only composed of a sinusoidal wave with one frequency component. The more general periodic signal can be composed of a superposition of a multitude of sinusoidal signals with different amplitude and frequency contents. These signals correspond to a stationary regime. In terms of deterministic non-periodic signals, transient signals are also present. Lastly, at the same level of the deterministic signal lies the random signal family. All signals enter this category or are a combination of signals in these categories. This approach allows characterizing vibrations.

Another characteristic of vibrations is the source. From this parameter, vibrations can be classified as either forced vibrations or free vibrations. The free vibrations correspond to responses from not maintained sources whereas forced vibrations result from continuous excitation. In the use case of washing machines, most measured vibrations are forced vibrations.

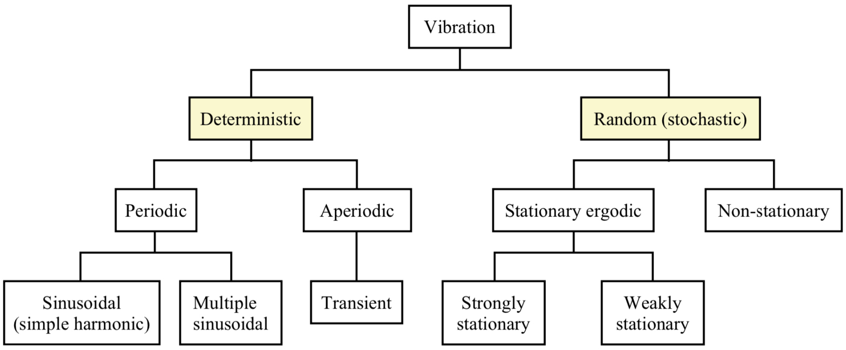


Figure 3.1: Classification of vibration performed by [36] and illustrated in [32]

There are three types of units to measure vibrations. First, there is the displacement of the physical movement [m], second the velocity [m/s] and last the acceleration[m/s^2]. These three measures each give their batch of information. Depending on the unit, different types of sensors can be used. This master thesis will only focus on acceleration measurement. Once the measure is performed, the data need to be processed in order to retrieve some indicators and metrics for interpreting the data.

3.2 Indicators and interpretation

In this subsection, a brief listing of basic different indicators used in the field of rotation machine failure analysis is presented from [33], [28] and [37].

3.2.1 Temporal indicator

Peak: The acceleration Peak is used as a threshold value. The higher the peak the bigger the force and thus the possible deformation. It is simply computed as the maximal value of the acceleration.

Peak to peak: The Peak to peak value of the acceleration is similar to the previous metric except that instead of taking from the equilibrium (no acceleration), the difference between the maximal and minimal value is taken.

Average: This quite obvious indicator is the average value of N measurements calculated as $\frac{1}{N}\Sigma_{i=0}^{N}y_i$ where is $\mathbf{y}$ is the vector containing the accelerations at each time t_i.

RMS: Another indicator is the RMS value. It is calculated as follows

$$RMS = \sqrt{\frac{1}{N}\Sigma_{i=0}^{N}y_i^2}$$

Where N is the number of measurements. We can describe the RMS as the energy content of the system. The higher the energy the more the system is capable to bend or move.

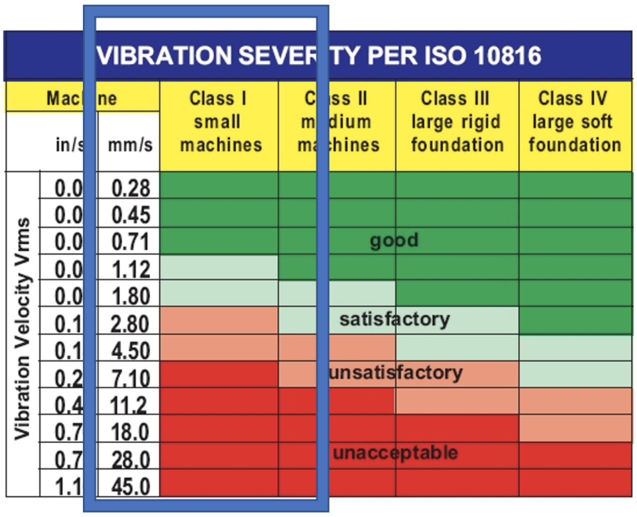


Figure 3.2: ISO 2372 norm giving the severity of vibrations of rotating machines at speed between 600 to 12,000 RPM where washing machines are categorised as Class 1

Crest factor: We use the crest factor as the factor measuring the "spikiness" of the data. It is defined as the ratio between the Peak acceleration and the RMS acceleration.

RMS Velocity: A simple yet important metric is the RMS velocity which is derived as the RMS values of the measured linear acceleration divided by the rotating speed. This metric is important due to the norm ISO 2372 displaying the acceptance of the vibration for rotating machines rotating at speeds between 600 to 12000 RPM. Table 3.2 represents the ISO norm where washing machines fall in class 1. The reason behind the use of the RMS velocity as the reference unit for the severity of the vibration is that the velocity is constant in terms of energy no matter the frequency. Whereas the acceleration favors the higher frequencies in terms of energy and the displacement of the lower frequencies.

Standard deviation: The Standard deviation is the first statistical metric used and is calculated as follows:

$$\sigma = \sqrt{\frac{1}{n-1}\Sigma_{i=1}^{N}(y_i - \mu)^2}$$

For the 2 next indicators, it is important to define the i-th central moment as

$$M_i = E[(\mathbf{y} - \mu)^i]$$

and corresponds to the higher-order expected value of the deviation of a statistical distribution from the mean. The first moment is the variance $M_1 = \sigma^2$. By standardizing these moments some interesting metrics are formed.

Skewness: The Skewness is defined as the third standardized moment and calculated as follows:

$$\gamma = \frac{M_3}{\sigma^3}$$

This metric gives a piece of information about the symmetry of the data around the mean as shown on 3.3. For large data sets, a Skewness between -2 and 2 is considered within an acceptable range.

Kurtosis: In the next order there is the Kurtosis which is the fourth standardized moment and is defined as follows:

$$\kappa = \frac{M_4}{\sigma^4}$$

This metric is used to measure the propensity to produce outliers.

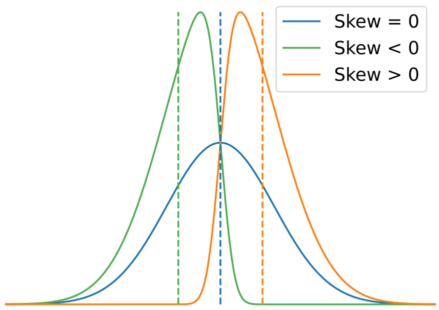


Figure 3.3: Effect of the data on the sign of skewness for a normal distribution

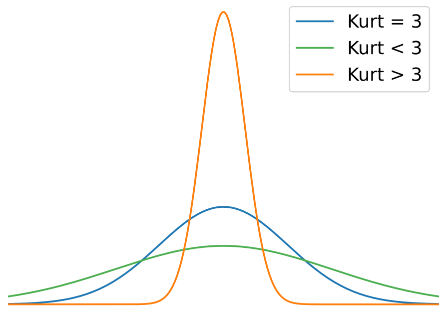


Figure 3.4: Effect of the data on value of the kurtosis for a normal distribution

All these indicators can be used with some measured data with and without prepossessing. However, with some transformations such as the Fast-Fourier transform, other more specific metrics are available.

3.2.2 Frequency indicators

FFT In order to pass from the time domain to the frequency domain, the FFT is the must-to-use tool. For the rest of the work, it will be assumed that the FFT is known as well as its limitations. More information on the topic can be found in [14].

Applying the FFT to the vibration (acceleration), data the acceleration spectrum is the result. From the spectrum, three main metrics can be retrieved: the frequency component, its amplitude, and the phase. In general, the frequency and the phase give information on the source and the amplitude on the importance of that source. Given the amount of data to analyze, there will not be any phase analysis in this work. For some information, please have a look at [38]. These three metrics have physical values as long as the FFT is performed on steady-state signals otherwise they can be used as comparison values without any physical meaning. It is also important to remind that to compare the amplitude outputted by the FFT, the number of samples must be the same.

Before using this tool, it is important to state the scope of usability. The most relevant in the scope of this work is the folding frequency. It is defined as half of the sampling frequency. The folding frequency is the maximal frequency in a spectrum that has a physical meaning, beyond lays the artificial spectrum created by the process.

Another closed related metric is the velocity spectrum and it is defined as the acceleration spectrum normalized by the frequency. This normalization allows the best picture of what lies in the lower frequencies.
The problem with the Fourier-based transformation is that it does not give information about the moment in time where those frequencies appear. In the same idea, the temporal plot of the acceleration does not give any information on the frequency content.

Power spectral density The Power Spectral Density (PSD) is derived from the Fourier Transform to obtain the energy content of a signal in terms of frequency. It can be derived as the amplitude of the FFT squared normalized by the frequency.

Spectrogram This indicator allows to display the Fourier transform over time. This is basically done by applying the Fourier transform to smaller parts of the data. Therefore there is a trade-off, as the frequency resolution increases the temporal resolution decreases and vice versa. For further information, the book [14] gives a great explanation of the concept behind the spectrum analysis.
The Fourier, PSD, and Spectrogram analyses are used in a diversity of domains aside from failure analysis of rotating machines.

Waterfall plot This graphical tool is more specific to rotatory machines. It is a Fourier analysis given a variation of speed. For this plot, it is necessary to know the exact speed of the machine at every given time. This type of tool will not be used in this master thesis since the actual rotating speed of the washing machine will not be measured for reasons explained in Chapter 6

3.2.3 Bearing related factors

Given the omnipresence and importance of bearing in rotating machines, wear and failure are deeply analyzed and documented [39]. Among all the metrics related to bearings, four bearing defect frequencies are most used to monitor faults in rotating machines.

BPFO The Ball Pass Frequency Outer Race is the frequency related to fault when the ball is on the outer race. It is calculated as follows:

$$f_{BPFO} = \frac{N_b}{2} f_r (1 - \frac{D_b}{D_p} cos(\beta))$$

BPFI The Ball Pass Frequency Internal Race is the frequency related to fault when the ball is on the inner race. It is calculated as follow:

$$f_{BPFI} = \frac{N_b}{2} f_r (1 + \frac{D_b}{D_p} cos(\beta))$$

BSF The Ball Spin Frequency is called the frequency of rotation of the rolling ball. It can be used to measure the wear on the balls of the bearing. It is calculated as follows:

$$f_{BSF} = \frac{D_p}{D_b} f_r (1 - \frac{D_b^2}{D_p^2} cos^2(\beta)).$$

FTF The Fundamental Train Frequency is the number of times the cage of the bearing does for one rotation of the shaft. It is calculated as follows:

$$f_{FTF} = \frac{1}{2} f_r (1 - \frac{D_b}{D_p} cos(\beta))$$

where N_b is the number of rolling elements, D_b the diameter of the balls, D_p the diameter of the pitch, β the contact angle, and f_r the rotor rotating frequency. The interest in having these notations is that for a given bearing (known beforehand), it is possible to know the source of the failure or near failure with the FFT.

Withing the bearing failures, there are 4 stages of failures as shown in Figure 3.5. It is only beyond the third stage that the previous frequency appears. If one were to analyse high frequency contents (which is harder the higher it gets) it is possible to notice signs of bearing failures.

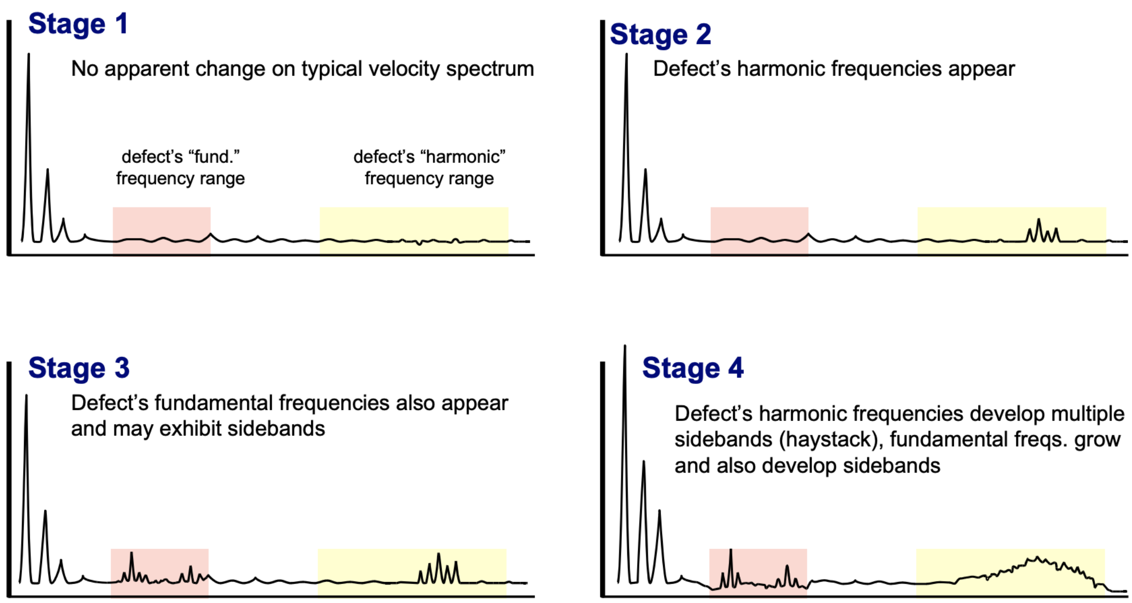


Figure 3.5: Frequency content of the 4 stages of failure of bearings from [33]

3.3 Assessment Failure severity

With the metrics and tools to measure the failure, it is meaningful to know how severe or how far from severity the machine is (close to predictive maintenance). This part is often dependent on the fault and system in question.

One possible method is to compute a basic threshold on the temporal measured acceleration, velocity, RMS, or other temporal value. This is the most obvious and easy way to work

Another method is to compute the FFT of many healthy systems and to create a frequential envelope. Once a frequency component gets higher than the envelope the alert of the failure can be triggered. The frequency can even give information about the source of the failure.

The list goes on. In the end, most conventional methods use a threshold system and compare it to the data. By measuring the distance (not only Euclidean for statistical models) the severity can be computed as explained in Section 2.3 and [29].

The best way of monitoring for failure is by analyzing known failures that have a high probability of happening as well as their symptoms. Such research has already been introduced in Section 3.2.3.

3.4 Common failures and symptoms

With all this knowledge about metrics and mechanical vibration, let's review some well-known and relevant failures of rotatory machines from [33] and [38] applicable to washing machines using acceleration measurements.

3.4.1 Misalignment

Misalignment can be parallel and/or angular. They concern systems containing any coupling, gearboxes, and belt transmissions. The effect is most noticeable on the bearings since they have to carry more load. This can be seen by a higher second harmonic component of the running speed. If the second harmonic amplitude is over 1.5 times higher than the fundamental one, then the misalignment is considered severe. Between 0.5 to 1.5 times higher than the fundamental, the coupling system is more likely to be damaged. Below 0.5 the fundamental, it is acceptable but to be monitored.

3.4.2 Unbalance

This failure is visible with a much higher fundamental of the running speed compared to normal use. There are three types of unbalance in rotating machines[38]: Static, Dynamic, and Coupled. Unbalance is overall diagnosed by a high-frequency component at the running speed without any harmonics. The amplitude of that component increases as the speed does.

3.4.3 Mechanical Looseness

The source of looseness can come from various problems. However, in washing machines, it is often due to loosened bolts and nuts or slack springs. This characteristic spectrum is given by many harmonics of the running speed up to the tenth.

3.4.4 Bearing and Gearboxes

For bearings and gearboxes, it is important to know beforehand the characteristics of each bearing and gear in the machine. In the case of bearings faults, they can be diagnosed using the frequency indicators developed in Section 3.2.3. In the case of gearboxes, the parameter to know is the number of teeth in order to compute the Gear Mesh Frequency (GMF) as follows:

$$GMF = n_{teeth} v_{shaft}$$

where n_{teeth} is the number of teeth of the gear and v_{shaft} is the rotating speed of the shaft attached to the gear. The gear failure-related spectra produce peaks at the fundamental and/or the second and third harmonics of the GMD. The third harmonics can be higher due to the three phases of gear interaction (engaged sliding, rolling, and disengaged sliding). Higher sidebands may also be present.

3.4.5 Stator problems

Eccentricity They are characterized by a high component at twice the frequency of the AC input current of the motor not modulated with the rotating speed in the radial measurement of the acceleration. This is due to stator eccentricity provoking variation of the air gap between the stator and the rotor. The factor 2 is due to the number of poles in the motor of the washing machines. In most cases, it is a two poles induction machine.

3.4.6 Rotor problems

Eccentricity Rotor eccentricity symptoms are also a high component at two times the frequency of the AC input current of the motor with pole pass frequency with sidebands and sidebands around the running speed. This is visible in the radial measurement of the acceleration.

Loose rotor bar The loose rotor bars up to open bars will induce by high components of two times the frequency line and side bands around the Rotor Bar Pass Frequency (RBPF) and sometimes its second harmonics. In the case of electric arcs forming between loose rotor bars, the component of the second harmonic will be then higher than the fundamental.

Broken, cracked rotor bar and shorting ring cracks

These types of failures will induce higher components in the fundamental and harmonics to the fifth of the rotating speed as well as high sidebands in each component.

3.4.7 Phase Problem

In case of loosened power cable, damaged or even broken connectors, one of the phases may not supply enough current. This will induce a high component at 2 times the line frequency with specific sidebands spaced of 1/3 of that frequency.

3.4.8 Multiple failures

In theory, each failure has its way to be determined with some degree of certainty but in practice, failures do not rise individually. This can lead to multiple factors appearing at the same time, generating coupling in the failure and symptoms making the individual analysis troublesome.

More information is available in the guides for machinery maintenance [38][33] and the books [31] [32].

Chapter 4

Scope of the project

4.1 Scope Definition

4.1.1 Overall project

Now that the literature has been swept, it is time to define the scope of this working thesis and the domain to explore. The main issue to address is to reduce the number of washing machines prematurely wasted because of the inefficient and costly steps of repairs. This can be done by applying predictive maintenance as shown in the previous section.

However, due to the lack of data, it is not possible to create such an algorithm at the moment. One way of proceeding would be to massively collect data with a known and powerful setup and then characterize the different failures individually. This could work but would be rather inefficient. Another path is to build a portable and reproducible module capable of capturing meaningful data. This data then can be used to train some predictive maintenance models using machine learning techniques. This later idea allows for a good trade-off between measurement capability and bigger-scale development.
The general research question over the project is defined as follows:

How to build a device capable to support a large-scale predictive maintenance algorithm on front-loaded washing machines at low cost in a non-invasive way?

The target appliance is the front-loaded washing machine given that they represent the majority of washing machines in service. Given that this project is done within the scope of a master thesis, it is impossible to start by designing the device and end with a working predictive model.
The project can be scattered into different phases. For the entire project to be completed, 4 phases are needed as shown in more detail in the planning visible in the Appendix A.

First Phase: Planning and analysis

After the general idea definition, this phase is where all the research is performed and early design features are characterized. A basic specification sheet and a first architecture (hardware and software) are to be delivered for a first prototype.

Second Phase: Failure recognition development

In this second phase, the main components in terms of hardware and basic software are to be developed to produce a prototype working in a controlled environment. At this point for simplicity, the system must be developed with only a few features and the minimal development needed for scaling. The system is meant to be capable of clearly showing differences in terms of basic indicators or trends between different states of the machine and forced failures. If the system is capable to provide tends and variation between the cases, the prototype is considered worthy to open doors for further development. This phase ends once the necessary processing is developed on the captured data to output a few metrics and values capable to serve as inputs of a machine learning model.

Third Phase: Scaling and development

The third phase starts with a retrospective analysis of the weaknesses and strengths of the system to produce an updated device (not a prototype anymore). This is the moment to perform the migration of the system from a local environment to a cloud-based system. Following that step, a first small deployment (below 10 devices) on working machines can be performed. In parallel, a larger amount of devices must be deployed in testing centers to gather efficiently accurate data on healthy machines at first. With this amount of data, a first failure detection system using machine learning can be developed. With a working failure detection system of fault capable of assessing some failures, the third phase can be closed.

Fourth Phase: Machine learning RUL

At this point, the algorithm assessing some faults is working on a few devices and for a few failures. The goal is to reinforce the algorithm by collecting a significant amount of data on failures via the test centers. The beginning of an RUL prediction algorithm can be implemented for actual predictive maintenance performances. Within this development, a middle scale (below 200 devices) is expected to be deployed among citizens. These devices should be capable of first pinpointing predefined known failures and notifying unknown anomalies.

Beyond that point, the device can be considered developed and distributed as a finished product. This does not mean that the system must not be updated regularly in terms of algorithms to incorporate more failures to assess and predict.
The development of the project is presented linearly but many backs and forths must be done between the phase to obtain a working system. Concerning the development of the algorithm, the different discussions presented are very generalized. In practice, many more steps must be performed to obtain the desired result.
This proposed plan is only in terms of basic system development and does not take into account any economical, human resources,... point of view. It can be seen as the author's general guideline for further development.

The research part of the first phase has been already mainly done. Some research has to be reviewed in retrospect but the heavy lifting is already written. The second phase is where this master thesis is at. By the end of this work, the last milestone of the second phase is to be achieved.

4.1.2 Scope of this work

This master thesis' goal is to attain the end of the second phase. Given the small amount of technical information available on faults specific related to washing machines, at the end of the second phase of the project, this master thesis is going to be an exploratory work. To end the first phase, as previously exposed a list of specifications has to be delivered. This is done by partitioning the actions that the device has to perform.

The entire process that the prototype has to provide can be simplified into 4 steps: sampling, storing, processing, and decision making.

Sampling

The first step is done via the sensors. The characteristics to take into account are the type of signal to measure, the range, the sampling rate, the resolution, and the error. It may also be useful to check for the sensibility to external signals such as temperature.

There are 2 main constraints on the design given the end goal. On one hand, the device has to be non-invasive. This is due to warranty reasons. If the device is to be used on any type of washing machine and by any untrained citizen, the machine must not be opened to install or operate. By opening the washer, the warranty lifts off. On the other hand, the device must be below low cost (around 50€maximum from a rapid survey executed on a dozen of users) in terms of hardware build to enable large-scale deployment at a low cost. Therefore industrial-grade hardware is to be put aside.

Because of these constraints, there are two options in terms of signal measurement given the literature shown in Chapter 2. Either monitor electrical signals or mechanical vibrations. The first one benefits from the ease of installation. The abundance of the data in the case of washing machines is smaller but scalability toward energy monitoring and saving are appealing. However, such a device has already been developed in [24].

This is why this master thesis is focused on measuring vibration signals via an Inertial Measurement Unit (IMU) which provides the acceleration via an accelerometer and the rotation rate via a gyroscope.
The experiments from [12], [13] and [10], show that the focus on vibrations up to 250Hz is a reasonable trade-off between the amount of data collected from the structural events and the complexity of the needed sensor. Moreover going up to 250Hz gives access to the tenth harmonics of the average maximum speed of the machine (1400RPM). As a consequence, the sampling rate should be at least every 2ms due to the folding frequency.

After some primitive tests, the sensing range can be defined within ±4g corresponding to a maximal measured acceleration of $39.24m/s^2$ It is important to find the lowest possible upper bound because there is often a trade-off between range and resolution. As the range gets bigger, the resolution decreases drastically.

As for the time of sampling, it is at first set to the entire washing machine cycle since there is no way of knowing the optimal moment to sample data. This provides further challenge since as shown in [17] and [40], the amount of time the machine is at a steady state, yielding sensible information is not significant. Therefore, tools relying on the steady state of the system must be carefully used. Furthermore, for those who calculate the amount of data, if a cycle lasts for 1 hour, there will be 3600*500 sampling. This means that the amount of data to be stored is going to be non-negligible. The reduction of data to store is left to the update performed in the third phase of the project.

Storing

The second stage is about storing. This stage is done before the processing stage in order to do the processing offline with more powerful tools. The storage is performed during the first phases by a Raspberry PI via Comma-Separated Values (CSV) files which are commonly used for storage of large data sets (order of a tenth of Megabytes by cycle) but future development Pickle files will be used [41].

Processing

The third stage is processing. This is achieved to derive some metrics and interpretable information from the data to distinguish the failures. The computation will be performed offline after the data is collected. It will be the last step done "automatically" in the scope of the thesis.

Decision making

Lastly, there is the decision-making step. The indicators and tools presented in Chapter 3 and outputted by the previous step are used to characterize the state of the machine. Once the actual device is within specification, diverse tests on actual washing machines are going to be performed. The reason behind these tests is to prove that the data collected by the device contains sensible enough information to perform later larger-scale data collection for machine learning-based algorithms to be implemented.

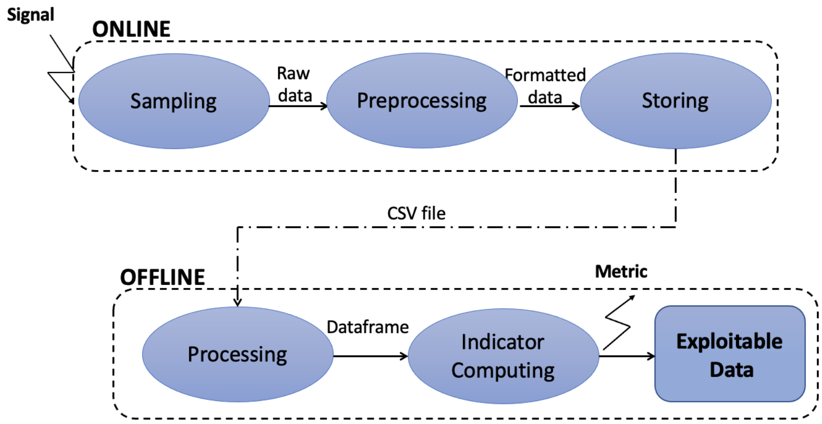


Figure 4.1: High level functional diagram of the device

4.2 First sketch and and guidelines

From the specifications presented earlier, a functioning graph can be established for the software part.
The software actions can be divided into two parts as visible in Figure 4.1. On one hand, the online part consists of real-time and time-dependent actions. This concern mostly the sensing and storing. On the other hand the offline part, the elements that can be done after the use of the machine. The interest in doing an offline part is to not be constrained by the computation capability of the mounted device. The online part concerns the physical device whereas the offline part could service.

The last major question to address before starting the development is the placement of the probing. Given the non-invasive, constrain and the information provided by [12] and [13], the top and side of the machines are ideal external places. Rather than choosing, the device is going to have two IMU, one at the top and another one on the side to assess the best positioning. A schematic view of the setup is visible in Figure 4.2.

Most the front-loaded washing machines use bent metal sheets as side panels. Thanks to this material the mounting of the main unit containing the side IMU can be performed using magnets. This feature allows easy placement and removal of the system.

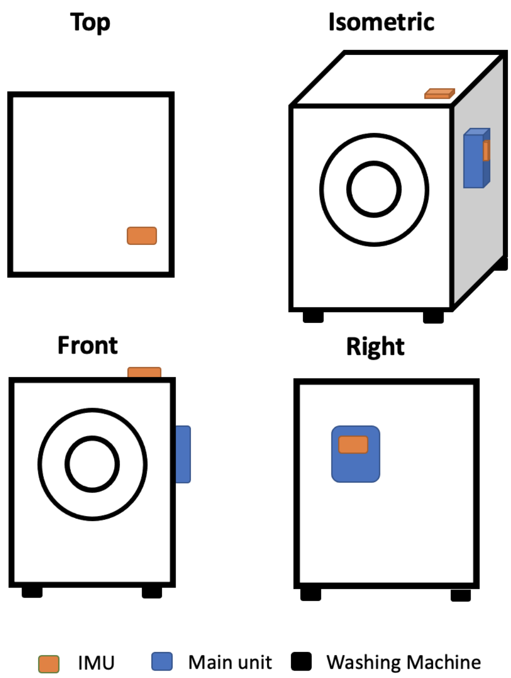


Figure 4.2: Device placement on a washing machine

Chapter 5

Device development

5.1 Different iterations

In this chapter, the different iterations are presented up to the latest iteration of the prototype. The development is presented in chronological order to procure the best guidelines for anyone trying to replicate or improve a similar setup. The development is divided into three parts related to the three main iterations.

5.1.1 V0 - The Raspberry PI Zero 2W

The first idea was simply to use a microcomputer to execute all the operations, one thread to sample the data and another to store the data in a CSV file.

Hardware Design

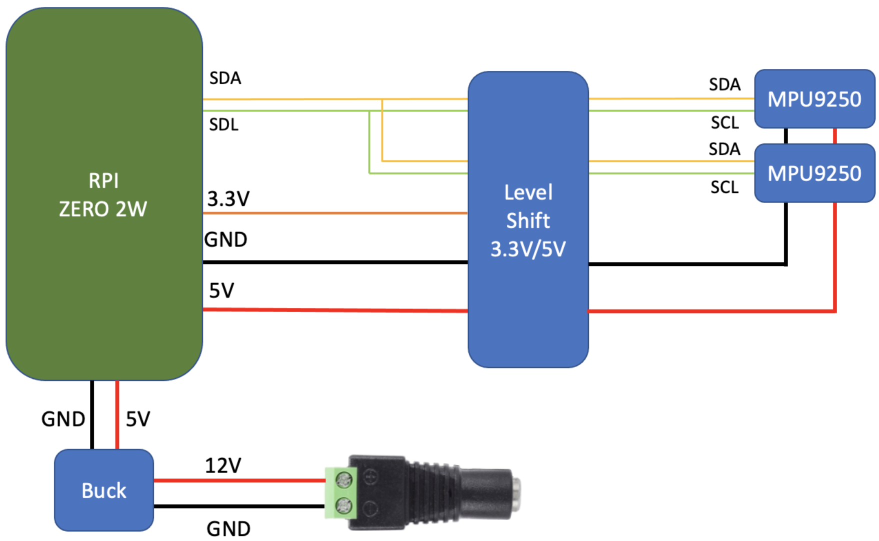


Figure 5.1: Diagram of the electrical architecture from the first iteration centered around a Raspberry Pi zero 2W.

As shown Fig 5.1, the microcomputer chosen was a Raspberry Pi Zero 2W due to its high performance for its size running a Linux-based Operating System (OS) called

Debian. This microcomputer was introduced by the Raspberry foundation in October 2021 [42]. It is almost as performant as the well-known Raspberry PI 3 B+ [43] for a third of the size.

As for the sensor, the intention was to use a ready off-the-shelf, and accessible components for the prototype. The choice went to the MPU9250 due to its high overall capability, developed libraries, and documentation. It is a 9-axis IMU. The 9 axes stand for 3 axes of acceleration (X, Y, and Z), 3 axes of rotation rate via a gyroscope (Yaw, Pitch, and Roll), and 3 axes of magnetic field sensing (X, Y, and Z). The magnetic field is a plus and will not be used in this thesis.

The MPU9250 has three main tunable factors: sample rate, full scale, and resolution [44] developed below.

Sample rate The first factor is the sample rate which is bound to the signal conditioning. For this first application, the highest constant sampling rate is required leaving almost no signal conditioning. This parameter yields an output data rate of 4kHz for the acceleration and 1kHz for the gyroscope. In this configuration, the magnetometer and digital motion processing allow wake-on-motion interrupts. However, the fast access to the data leaves a rather inconsistent data rate. Therefore a sample rate is set up to retrieve data at a more constant rate around a theoretical value of 2ms.

Full scale Second, given the small amplitude and few primitive tests, the smallest achievable scale was judged more than enough. The lowest scales are ±4g for the acceleration and $250°/s$ for the gyroscope.

Resolution Third and last, the resolution is related to the scale. The resolution is constant in terms of bits and is defined by the internal ADC of each sensor. The accelerometer and gyroscope ADCs both operate at 16bit and the magnetometer at 14bit. From the scales and these values, the Least Significant Bit (LSB) is $2^{-14}g$ for the acceleration and $7.63m°/s$ for the gyroscope. However, due to sensitivity and noise, these values are in practice higher. The lower bound of the actual precision can be at a first-order estimated with the RMS noise. For the acceleration, it is 8mg and $0.1°/s$ for the accelerometer.

To compare the vibration from the top and the side of a washing machine as presented in Section 4.2, two MPU9250s are used. Since they both use the I2C [45] communication protocol, the second one needs to have its address changed via an extra pin or via some soldering on the board as explained in the datasheet [44].

Logic level converter Since the used IMU boards work on 5V, there is a need to convert the 5V I2C data from the IMUs to 3.3V to be readable by the Raspberry and vice versa. With this information, the constrain on the logic level converter must be able to work on frequency to the order of 100kHz and is bidirectional. For this application, the BSS138 [46] is an appropriate choice.

Power The powering of the device was done via an AC to DC transformer plug on a power outlet and outputting 12V up to 5A. However, the system needs to run at 5V hence the buck converter. During a full measuring cycle, the entire device consumed on average almost 3A with power spikes up to 3.5A at 5V.

Mounting on machine The mounting of the device on the washing machine is done through magnets for the main part on the side of the machine and with tape for the top IMU. The magnets allow a strong yet removable design. The use of magnets is possible due to the metallic sheet on the side of the machine as explained in Paragraph 4.2. The bending part of that same sheet allows a constant non-moving placement of the magnets. For the IMU placed on the top, double-sided tape is used. Both mounting methods have to be rigid in order to avoid any parasitic coupling influencing the measured acceleration.

Software Design

On the software part, everything was done in Python[1]. The code was divided into two parts same as explained in Section 4.2. The sampling part reads the values of the register of the MPU9250 and stores them in a dataframe. This dataframe is then used to write to the different files.

I2C data transfer The I2C bus used for communicating with the IMU (MPU9250) is running at a speed of 100kHz on the Raspberry. This means that for a complete batch of information from the IMU, meaning 3-degree acceleration, 3-degree rotation rate both at 16-bit resolution means that the minimal total number of bits to be transmitted is 96 bits. Given the frequency of 400kHz (bits/s) and the I2C protocol (more detailed information about I2C protocol in [45]), this means that a complete set can be transmitted every 0.295 ms. Therefore the theoretical full sampling frequency is at 1kHz which is higher than the threshold defined in Section 4.1.2. This is however only for one IMU.

Data Processing The part concerning the processing and metric extraction was developed apart as described in Section 4.2. This second part is also developed in Python using the Pandas [47] module allowing fast and powerful analysis of large data sets. For the first version, the processing consisted mainly of displaying the measured values and some statistical information.

5.1.2 Results and shortcoming

Due to the many layers between the user code and the actual signal, the operation of sampling was taking on average 10ms and storing 10ms for each sample. This performance was achieved when there was no other process like Secure Shell (SSH) or

[1]not the best idea in retrospect

even a desktop graphical interface using too many resources. Leading to an overall sampling rate of 20ms. Compared to the specification in Section 4.1.2, this sampling rate is over the specified sampling rate by an order of magnitude.

After a lot of optimization (still in python...) the reading speed of one register was reduced to around 1ms and the conversion of a proper value to store around 5ms for one IMU. Even with such performances, the sampling specification is far from achieved.

Few tests were done using a C++ code and a community-developed library [48] leading to a small improvement but no difference in terms of order of magnitudes without recoding the entire library. After a lot of tweaking here and there, another architecture was needed. This change of direction was also the opportunity to think about the bigger picture and scaling. The idea was to use some network system to facilitate the data collection on machines.

5.1.3 V1 - Physical separation of the online part and the offline part

The second main iteration consisted in replacing the device performing the sampling part of the Raspberry. The replacing component was a microcontroller which is much faster for the operations of sampling and the preprocessing achieved at a lower level.

Hardware

The new general architecture is presented in Figure 5.2. The path from the first architecture to the one presented was led by 3 big questions: Which microcontroller to use? How to store the data externally of the microcontroller? How to send the data from the microcontroller to the storing system.

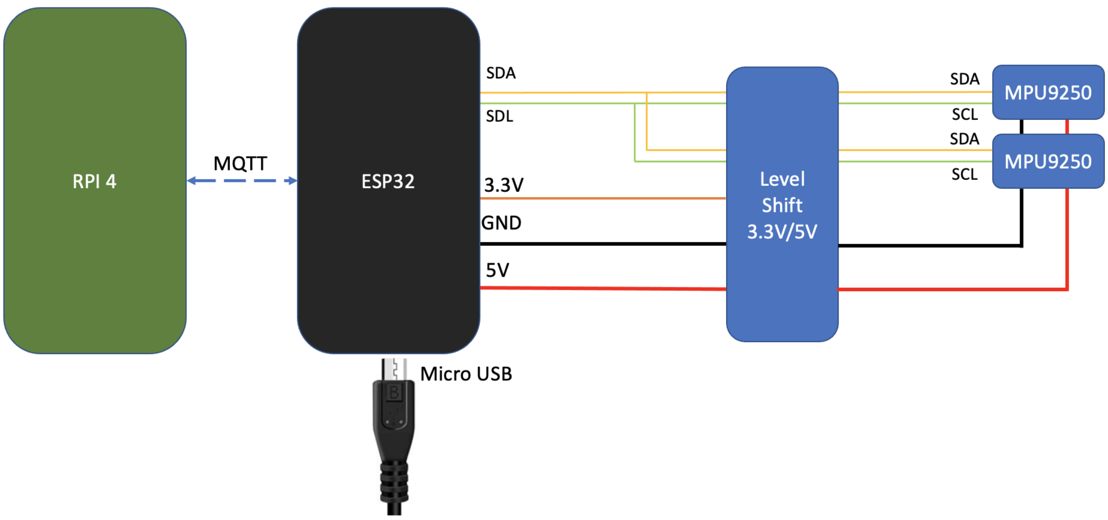


Figure 5.2: Diagram of the electrical architecture from the first iteration composed of the ESP32 microcontroller and the Raspberry Pi 4 microcomputer

Wireless communication type To start the redesign finding the right microcontroller was the first step. To simplify the prototype, the choice of wireless communication was paired with the choice of the unit. Therefore this criterion is first considered. Few options were possible: to use a radio frequency emitter and receiver like the NRF24L01 (2.4GHz)[49], Bluetooth or WiFi. The first one was rapidly put aside since closed indoor spaces are not ideal for such system. In the idea of giving the device to citizens, asking to use an external antenna to the main unit is not a good fit. This is why any device needing an additional receiver was put aside such as Bluetooth. Therefore the end choice settled on the well-known, everywhere implemented WiFi communication system.

Microcontroller Once the choice to use WiFi was made, three main paths were quickly available. The first was to use a classical Arduino board such as the Arduino Nano 33 IoT[50] which runs on a 32bit microcontroller running at 48MHz including 256kB SRAM and built-in WiFi. The second was to use an ESP family module like the latest ESP32-WROOM[6] that will be referred to as ESP32. This revolutionary 32-bit, dual-core microcontroller running at 80MHz has also built-in WiFi and is FreeRTOS [51] compatible. This device has 520kB of SRAM. The last option is to use some less known and documented microcontrollers. Due to the capability, availability and price, the ESP32 was chosen.

Power With this new configuration, the whole setup was capable to run without problem with the power supplied by the micro-USB connector. Consuming less than 150mA on average with 170mA spikes at 5V. This low power consumption is a major upgrade compared to the first version.

Software

The communication protocol to use for the whole information traffic is the most complex choice. Looking at the literature on similar IoT devices such as in [24] and [18], the use of MQTT [52] communication protocol via WiFi seemed a good idea. Other protocols were available such as HTTP [53], Websocket [54] or CoAP [55]. However, they are either slower, heavier, or harder to take a grip on as explained in [56] and [57]. From the tests in [58] and [59] the time to send a packet is below 2.5 milliseconds for publishing and subscribing. The specifications barely fit but these tests were conducted in the worst-case scenarios in terms of quantity of information and amount of data sent to the server.
Using the MQTT protocol allows for the scalability of the system. Each ESP32 needs to connect to the WiFi and connect to a broker here via the Mosquitto service which manages the entire traffic. By connecting, each ESP32 becomes a client to the system.

Short explanation on the MQTT protocol The MQTT protocol works with a publish/subscribe system. Each client can choose to subscribe to any message sent within the system via a specific topic. At the same time, any client can send any message (up to 256kB) to the system with a specific topic. This method allows for fast and lightweight communication from machine to machine without needing to change the entire system architecture while scaling. To allow such architecture

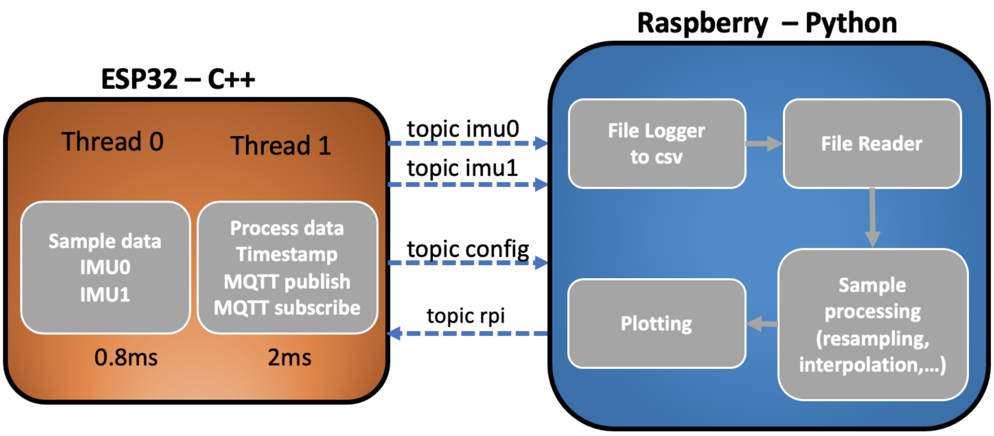


Figure 5.3: Software architecture of the second generation

a managing system called a broker needs to be installed. The broker used is called Mosquitto[2] [60]. This program will ensure that all the systems communicate properly via WiFi using the MQTT protocol.

For the current case, the message containing the information of the IMUs is split. One topic is used to send the information from the IMU0 (the one on the side) and another from the IMU1 (the one on the top) each with its own time stamp. Other topics are available to send the rest of the information as shown in Figure 5.3.

Results and shortcomings

Now that the system is entirely defined, let's sum up the new system.
An ESP32 is connected to the local WiFi to communicate with a server via MQTT. The user has to send to the ESP32 the command to start and stop the sampling as well as some information about the setup via the interface running on the server. The server would at the right moment collect the data send from the ESP32 via WiFi and store it in a file.

In practice, the local available WiFi network was configured to block the MQTT type communication (the only WiFi available where the machines were installed) an alternative way to use the Raspberry used the local server to emit its own WiFi. This was easily done using the plug and play platform RaspAP which is an easy wireless router setup for Debian based OS .

With this new setup, the values received were coming within at an average rate below 2.5ms. But another problem came, as shown in Figure 5.4, huge latency peaks of the order of seconds appeared in the system. This was mostly due to a too high load in the MQTT communication creating clogging the moment the WiFi showed any latency. These delays attaining up to 4s are unacceptable at such frequency for the application. To reduce this latency, many changes were tested. All using some kind of buffer.

[2] yes with double 't'

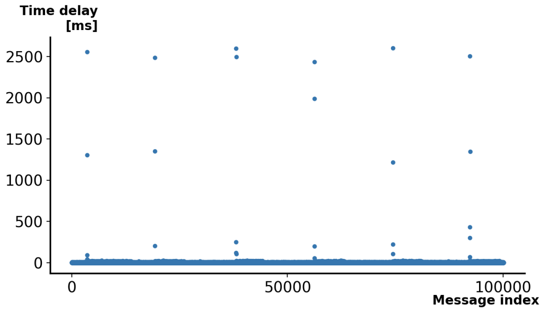


Figure 5.4: Time in milliseconds between each message for the 100000 messages of a cycle for the **first** version

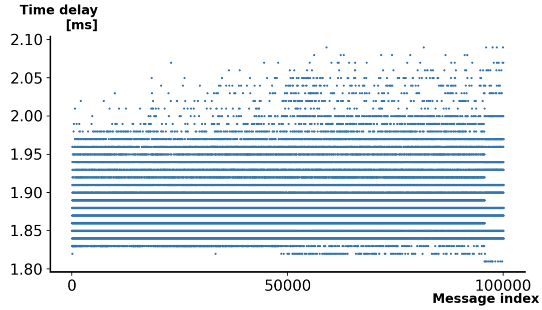


Figure 5.5: Time in milliseconds between each message for the 100000 messages of a cycle for the **Second** version

Hardware Buffer In terms of hardware buffering, the idea was to use extra storage. The easiest option given the moment was to use an external SD card. They are characterized by their size and speed. For the following test, a 16GB class 10 micro SD card was used. The 10th class is the fastest conventional SD card speed available capable of writing at 10MB\s according to [61], 10MB\s is more than enough given that each information *block* is smaller than a 120 char array so 120 bytes. This makes a writing time by *block* of $12\mu s$.

The communication of the ESP32 to the SD card can be done by 2 means, via SPI or MMC.
At first, the writing speed was barely enough for the application (even with the 40MHz SPI mode) and as the SD card was filled, the writing speed dropped way below the necessary speed.
The first system yielded writing speed up to 10ms and the latter up to 4ms per message when the SD card contained few MB.
Given the amount of work to rewrite and optimize the writing on the SD cards or to add some RAM modules, some other software alternatives were investigated beforehand.

Software buffering Among those, a queue system between the two cores on the ESP32 allowed a buffer of a few seconds before complete clogging of the WiFi MQTT traffic. This method coupled with some other software tricks ended up in the final version of the device which is explained in the next section.

5.1.4 V2 - The final prototype

With a completely new software algorithm, it was time to also update the hardware to a more reliable system. This is why a Printed Circuit Board (PCB) was designed and then manufactured. The final device and the placement are shown in the figures in Appendix D.

Hardware

In the end, a 2 sided PCB for development purposes was designed allowing more flexibility. The PCB visible in Appendix C was over-designed in case of further changes by giving the possibility to plug more sensors than the 2 IMU currently used. This is why 4 I2C entries in each 5V and 3.3V as well as their relative power sources were added. After a quick lookup on other possible sensors, ADC connections were added to the board. Thus 4 more ADC entries were added in both 5V and 3.3V. These ADC are directly connected to the internal ADC of the ESP32 so they required no additional hardware. For the 5V to 3.3V communication (ADC and I2C) level converter, the BSS138 was used as explained in the paragraph 5.1.1. The complete Bill Of Materials is visible in Appendix B.

Final code architecture

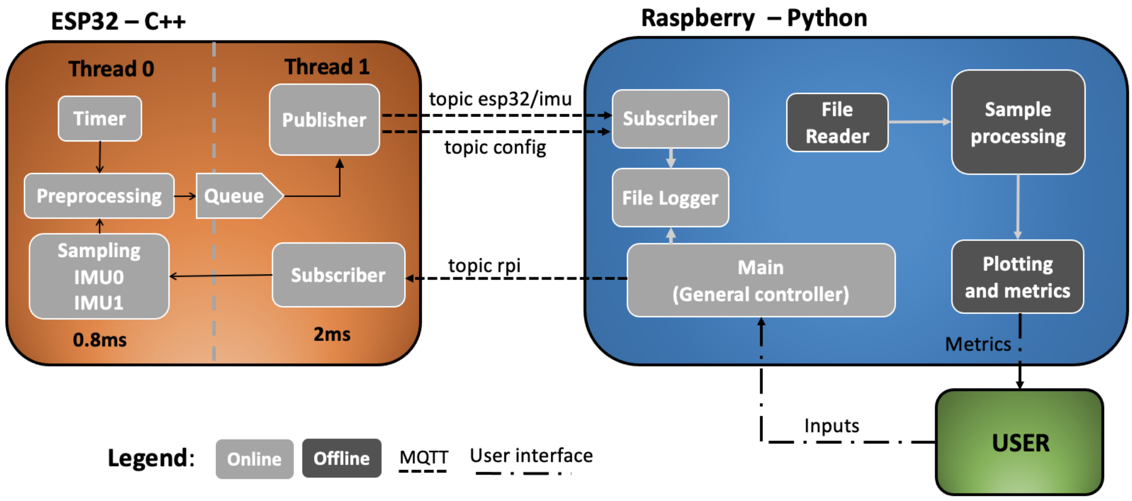


Figure 5.6: Final software architecture using only one topic for both IMUs

The final software architecture is displayed in Figure 5.6. From higher to lower level there is the Python code that controls the continuously running server. This server is used as the control room via the broker explained in Section 5.1.3. This is managed by the Mosquitto service running on the MQTT communication protocol as developed in 5.1.3. The Mosquitto service is running in the background on the Raspberry PI/ In parallel, a subscriber program running on the Raspberry will receive any message published with a certain predefined topic. In this case, every message is sent by the ESP32. To ease the code every topic has the layout *dev/elem* where *dev* corresponds to the source device that publishes the message and *elem* is the precise topic of the message. Therefore the subscriber program on the Raspberry Pi listens to every message with the topic starting by *esp32_X* corresponding to the message sent by the ESP assigned to the X value.

Once a user wants to record a washing cycle, the program cycle.py containing the main program is launched. This program is responsible to set up everything needed to monitor the cycle.

First, some basic information(cycle type, rotating speed, ...) about the cycle needs to be inputted by the user, then this information and the actual date are passed to a logging code which sets up the file where all the information is stored. The

main program is charged to tell the ESP32 when to start and stop the sampling depending on the cycle duration via the *rpi* MQTT topic.
Once the cycle program asks the ESP32 to start to sample by publishing a message of sampling to the topic *rpi*, the ESP32 samples the data and publishes the data to the topic *esp32_0/imu* which is received by the Raspberry subscriber.

On the side of the ESP32, there are a few more tricks. The ESP32 is subscribed to every message published by the Raspberry. Once the sampling is requested, it will not simply read the data, preprocess, and send them. The ESP32 stores the data in a Queue containing 455 items of 200 bytes long each which serves as a buffer to avoid the MQTT server to clog. The limit on the size of the published item is set by the MQTT protocol at 256 bytes. The Queue is bounded by the amount of volatile memory of the ESP32. Within each item, 2 samples from each IMU containing the 3-axis acceleration and 3 axes gyroscopic rate are stored. The reason for having an item containing 4 samples is because the MQTT protocol handles better larger messages (up to a limit) at a slower rate rather than smaller messages at a higher rate as shown in the measured delay of the samples in the figure 5.4 for the previous version and 5.5 for the current version. Since the noise on the acceleration is at the order of mg, the number of digits under the unit transmitted to the server by acceleration value is 3.

Given a worst-case scenario where each item of the Queue is sent every 2ms containing 2 samples of each IMU, the Queue allows for at least 0.9s buffer in case of clogging. The sampling and storing of the Queue are done on one Core of the ESP32 and the extraction from the Queue and the publishing on another.
Using this tricks the average time between samples for each IMU is below 2ms while never exceeding 2.15ms as visible in Figure 5.5.

Due to the non-negligible amount of code developed during this work, all the code developed in this work is only available upon request.

With this last performance, all the specifications listed in Section 4.2 are achieved. The device is therefore ready to be used and tested on washing machines to perform some measurements.

Chapter 6

Results

6.0 Disclaimer

In this chapter, the device previously developed is used to capture data on various use cases. The studied cases are variations of loading, speed, and leveling of the machines. The reason for the variety of the cases is to observe a noticeable difference in the measured data. This part is done with as little processing as possible since the processing of the data is a topic on its own. All the data and the processing related is available upon request.
Due to the amount of data, the main topic of discussion is based on the norm of the total acceleration measured by the horizontal IMU. Some further analysis on the acceleration captured by the vertical IMU is at the end as a comparison. There will not be any discussion whereas the use of the data related to the gyroscope.

6.1 Use case machine

6.1.1 General setup

To perform the tests, a machine was provided by the group *Fnac Darty*. The model used for the experiment is an AEG prosense 6000 series.
All the cases are studied using the same type of program called *Pompen/Centrifugeren* on the machine performing the shortest cycle. Depending on the speed, the overall cycle does not follow the same trend.
The variability of the cases is performed via four factors: the rotating speed, the balance of the machine, the type and placement of the load.

First, the studied speeds are 400, 600, 800, 1000, 1200 and 1400 Rotations Per Minute (RPM). The experiments on the fastest rotation speed are not always performed due to the high energy content at high speed (Energy $\propto \omega^2$). Those speeds are selected according to the choice of the program. This program is gives three characteristics speed variations. Those are a slow and a fast acceleration toward the maximum inputted speed and a long steady state at that speed.

Second, the balance of the machine is manipulated via the leveling of the feet as a binary parameter (balanced/unbalanced). This is studied since it is often a source of acoustic noise in washing machines and is fairly easy to test. The leveled (and also

Experiment\Speed[RPM]	400	600	800	1000	1200	1400
Empty - balanced	X	X	X	X	X	X
Empty - unbalanced	X	X	X	X	X	X
Centered 1kg - balanced	X	X	X	X	X	X
Centered 1kg - unbalanced	X	X	X	X	X	X
Centered 2kg - balanced	X	X	X	X	X	X
Wet towels - balanced	X	X	X	X	X	X
Shoes - balanced	X	X	X	X		
Loose towels 1.5kg - balanced	X	X				
Joined towels 1.5kg - balanced	X	X				

Table 6.1: List of all the experiment performed

called balanced in this work) machine is always leveled via a level indicator using the *Phyphox*[62] phone app within 0.3° from a completely flat position. The unleveled (or unbalanced) machine is capable of doing a frontal and lateral movement of 2°.

Lastly, the weight and compactness of the load are studied by inserting different types of loads in the drum of the machine. There are 6 use cases: empty, 1kg centered weight, 2kg centered weight, wet towels, loosen towels, joined towels, and shoes.

Not all the combinations were performed due to the lack of interesting data, time, and security of the setup. All the experiments performed are visible in the table 6.1.1.

Centered load experiments

To maintain a constant load almost centered a setup was designed as shown in Figure 6.1. First, the 3 drum paddles were removed to give access to the holes in the drum where steel wires are attached as mounting points. From each mounting point, a hook is attached to the main load placed at the center of the drum.

The centered load is composed of 4 parts as shown in Figure 6.1: a PVC pipe, 6 acrylic pieces, 2 plastic covers and a plastic separator. The PVC pipe serves as a container to increase the mass of the load and the plastic covers to avoid the interior of the pipe to escape while maintaining the acrylic pieces. Those pieces are there as a mounting point on the load to attach a band linked to the hook. To impede the acrylics piece to snap together a plastic separator is placed in between. While placing the setup, the centering was achieved within 2.5mm.

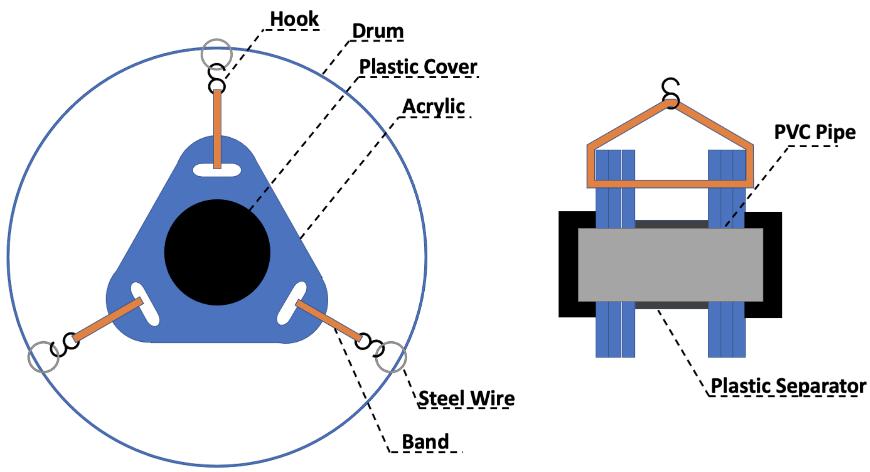


Figure 6.1: Schematic view of the setup for the centered weights

Without any additional weight inside the pipe the whole load system weights around 1kg. An additional mass of 1kg inside the pipe can be added. Using this setup a centered mass of 1kg and 2kg is simulated. The actual setup is visible on Figures 6.2 and 6.3.

Figure 6.2: Front view of the centered weight setup in the washing machine

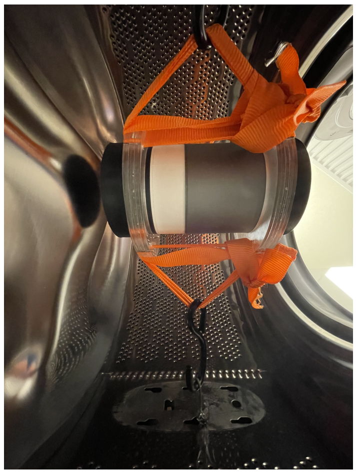


Figure 6.3: Lateral view of the centered weight setup in the washing machine

Dry towels experiments

For these use cases, two beach towels are used, each weighing around 750g. Two cases are tested, the first is when the towels are put in the washing machine normally meaning completely unconstrained. In the second, each towel is folded as much as possible and they are attached together as shown in figure 6.4. These 2 towels are not attached to the drum and are capable of moving around in the drum. To maintain consistency in the experiences, the joint towels are always placed and maintained with their center axis parallel to the center axis of the drum during the cycles. The experiment with the towels is only performed for rotation up to 600RPM since the joint towels already display noticeable effects on the machines. Moreover, the testing environment is not secure enough to perform dangerous tests.

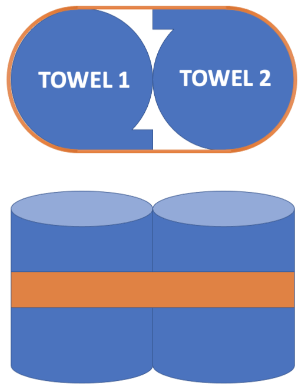


Figure 6.4: Attachment of the 2 joint towels in the joint towels experiments

Shoes experiment

Lastly, to simulate a completely realistic and yet noticeable case, two running shoes are placed in the machine. The two shoes weigh approximately 800g together and are inserted without the laces as instructed in the washing machine manual. They are placed without fixation in the machine leaving any kind of movement possible.

6.1.2 First look at the measurement

As an entry to the analysis, a global preview of the raw measurement is mandatory to have a first look at the scope of the measurements.

Temporal analysis

The Figure 6.5 displays the raw measurement of the norm of the total acceleration at 1000RPM on an empty and leveled machine captured by the horizontal IMU. This is why the mean value of the measurement is near $9.81 m/s^2$. Before any case analysis, few elements can be extracted for this figure.
First, the moment when the start button is pressed and the front door gets locked is noticeable in the acceleration plot shown by the LOCK zoom. This signature response is characteristic and varies little across all the tests.

Next, 3 small-amplitude batches of vibration can be observed in Figure 6.5. Each of these batches corresponds to one push sequence of the water pump to drain the remaining water in the drum. The water pump runs many times during the cycle but is often not noticeable due to the vibration of the rotating drum. Given such responses, the device can be expected to highlight pump failures if this is developed further.

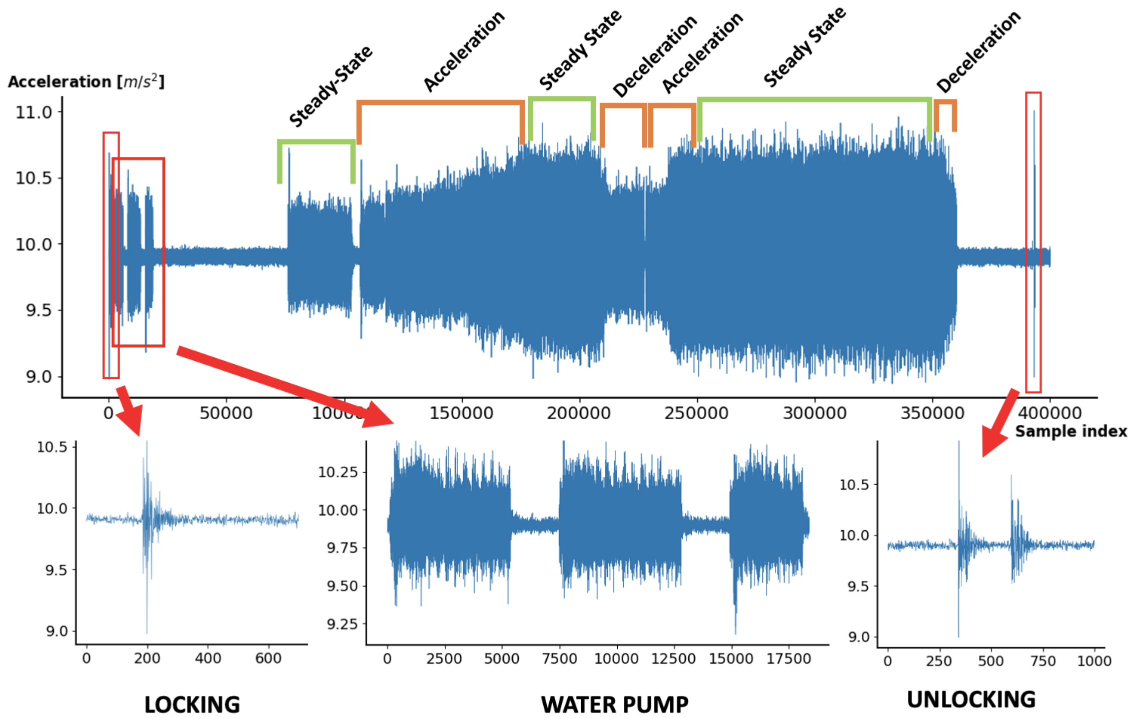


Figure 6.5: Norm of acceleration from a full cycle of a washing machine with no load at 1000 RPM from the horizontal IMU with more details on 3 specific responses

The next three characteristics of higher vibration blocks are visible. The first block corresponds to a first rotation at around 400RPM independently of the input speed with almost no noticeable transient (except in high speed and heavyweight). The second block starts with a slow acceleration towards the input speed, maintains a relatively short steady-state and ends with a slow deceleration. The last block is composed of a fast acceleration to the max speed followed by a long steady-state and finishes by a rapid deceleration. The transient acceleration amplitudes of the last block vary considerably among the different cases. During the steady-state of the last block, the spectral analysis is performed.

A similar trend is visible from the vertically attached IMU.

At the end of the cycle, the unlocking of the door is visible by 2 signals similar to the locking system. These signals are due to the movement of the switch used to lock and unlock the door. Since these locking and unlocking events are unique in time and response, it can be viable to perform some sleep system for the device. This sleep system could use the sharp locking response to wake up the device to start the sampling and use the characteristic response of the unlocking system to stop. This sleeping system could reduce the power consumption by a factor of magnitude according to [6].

With the temporal approach presented, a spectral analysis of the same measurements is then presented.

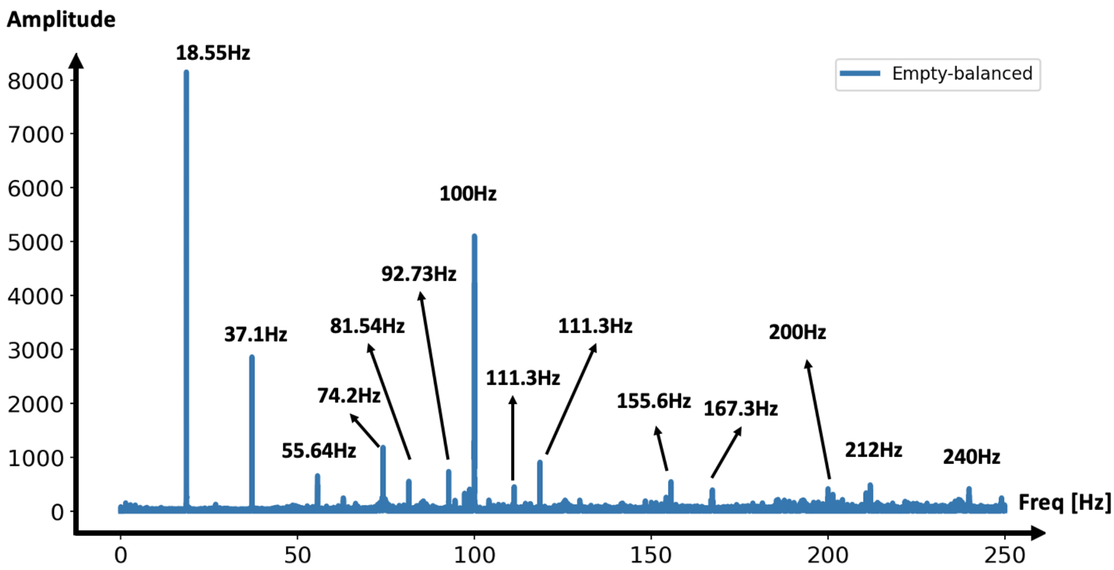


Figure 6.6: Spectrum of the acceleration norm from a full cycle of a washing machine with no load at 1000 RPM from the horizontal IMU

Spectral analysis

From the measurement at 1000 RPM previously presented, an FFT transform can be applied in the steady-state region leading to Figure 6.6. This spectral representation gives more interpretable information on the system concerning the source of vibrations.

This FFT is performed from a set of the data presented in the longest steady-state part of Fig 6.5. Since the data rate is not completely consistent as shown in Figure 5.5, resampling and interpolation are performed on the temporal data to achieve a 2ms constant data rate. This allows the spectral response up to 250Hz shown in Fig 6.6.
First, the highest peak is at around 18.56 Hz (the first peak on the left). This peak corresponds to the rotation speed of the drum which is in this case 18.56 Hz meaning around 1114 RPM. This is also confirmed in [40] where the speed profiles are not maintained at the exact inputted speed.

The second highest peak is at 100Hz and is accompanied by some sidebands. This 100Hz peak is visible in every spectral plot on this machine but the sidebands are not always present. From the common failures explained in Section 3.4, this is mostly due to some stator eccentricity since the 100Hz is 2 times the AC input current frequency and there is no sign of modulation with the speed.

Given that the machine is a repaired one, other few non-standard peaks are visible. It is however possible to characterize some of them, especially at lower frequencies. For example, those at 37.1, 55.65, 74.2, 81.5 and 92.75 Hz correspond to harmonics of the rotating speed and will be further explained in the section 6.2 and are signs of slackening springs (which was later confirmed). Those are visible in different cases of 1000 RPM cycles. Other peaks are visible such as 81.5 and 118.55 Hz. Those 2

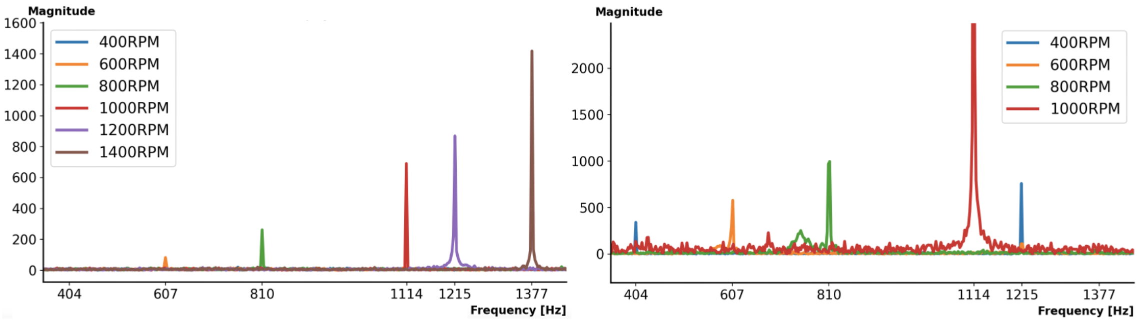


Figure 6.7: Spectral responses below 1400RPM of the norm of the acceleration of the case of the centered load on the left and the shoes to the right from the horizontal IMU for various rotating speeds

due to their values and their modulation with the rotating speed of the machine are suspected to be caused by the axial oscillation of the drum or the movement of the heating element of the incoming water that was not correctly tightened after some inspection.

The observation of one case allows a first analysis and some guesswork but to unfold the capabilities of the device, it is necessary to perform a cross-analysis of the different cases.

6.2 Cross analysis

In this section, the different cases presented in Figure 6.1.1 are compared. First in terms of spectral analysis since it is the most visual way to analyze and later few temporal figures similar to Figure 6.5 are shown to give some feelings about the data.

6.2.1 Cross spectral analysis of the speed

For this section, the influence of the speed is developed for two cases. The centered load of 1kg and the shoes are presented in Fig 6.7.
These two figures are shown because they illustrate well the visualization of the rotating speed of the washing machine for different cycles. The horizontal scale was converted in RPM and relabeled to highlight the close relation to the rotating speed of the washing machine. The frequency of vibrations is not exactly the same as the rotating speed displayed on the machine. However, from [40] and the consistency of these frequencies across the multitude of tests, those frequencies are related (if not equal) to the rotating speed of the machine. This relationship is even enhanced by the apparition of harmonics of this peak in Figure 6.6. Knowing this, it could be considered that this device is capable of capturing the rotating speed of the machine and detect any linked failure.

From this analysis, we can conclude that the device is capable of measuring the running speed of the washing as long as there is enough speed or load. This conclusion is to be further displayed as the superposition of the different use cases is done.

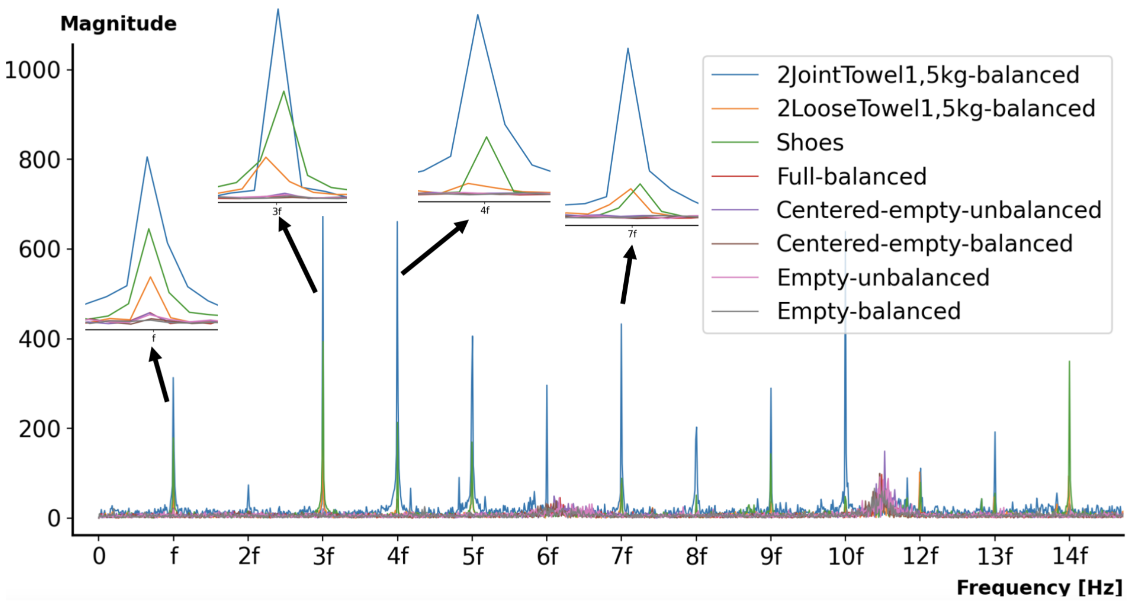


Figure 6.8: Spectral response of the various cases studied at **400RPM** for the side IMU

6.2.2 Cross spectral analysis between use cases

To compare the different use cases, a similar time span is chosen between the experiments for the same speed. This time span is within the steady-state period of the cycle as presented in Section 6.1.2. All the time spans contain the same number of samples. This allows comparing the frequency content as well as the amplitude of the spectral peaks.
The result for 400RPM and 600RPM are visible on Figures 6.8 and 6.9 where the different cases where sorted toward the worst case scenario. A close look on the X-axis again shows conversion of the frequency in terms of multiple of the rotating speed *f*. This is done intentionally since it displays strongly all the visible harmonics across the spectrum and the use cases.

Not all the harmonics present have the same distribution within the same rotating speed and between the rotating speeds. At 400RPM, many harmonics are dominated by the worst-case scenario, the joined towels followed by the shoe case and the case of the loose towel. The lack of presence of the other cases at 400RPM is due to the low speed and low effect of the loading.
Regarding the distribution of the spectrum at 600RPM, the increased speed rotating permits a different response depending on the other factors. For example, the previously dominant case of the 2 joined towels is not anymore the higher amplitude in several harmonics. Some other modes start to appear. It can be pointed up that the first harmonic (2f) is now way more important in the worst-case scenario whereas in the second harmonic (3f) there is no resonance at that frequency for that scenario. There is even the rising of peaks from the previously flat cases of the centered load.

The 100Hz peak due to the stator eccentricity is visible on every curve as explained in Section 6.1.2 and is only surpassed by another peak only for the case of 1.5kg joint towels are placed which is an extreme case.

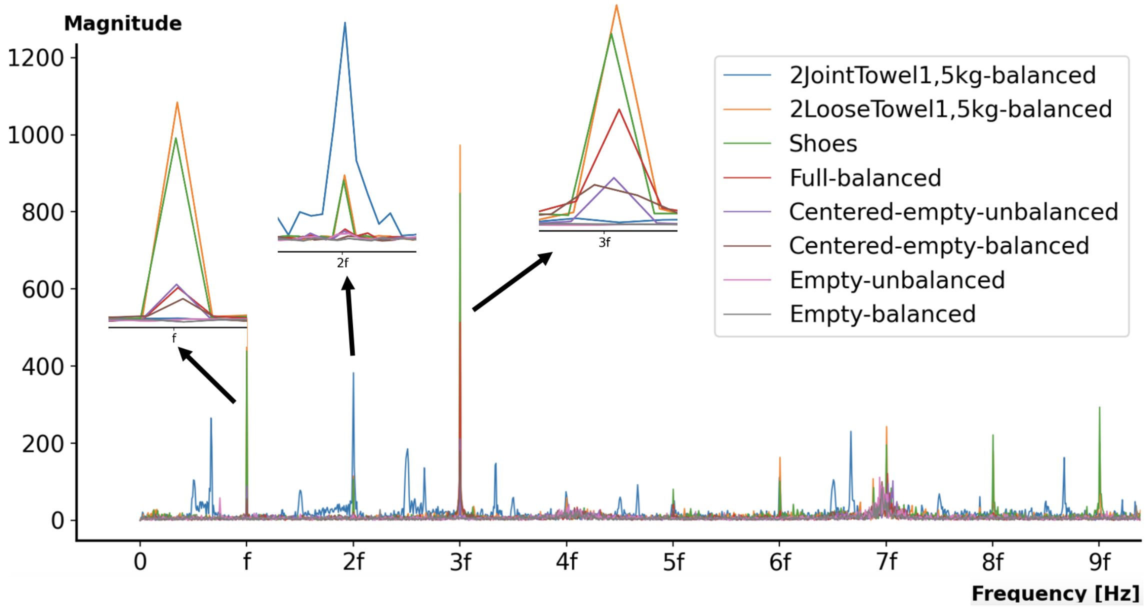


Figure 6.9: Spectral response of the various cases studied at **600RPM** for the side IMU

From these spectral results of the side IMU at low speed, the effect of leveling the washing machine is almost negligible. For the loosen towel case a component at twice the rotating speed appears and this component becomes prominent for the 2kg centered load and the joint towel cases. These off-centered mass cases share a sub 100Hz spectral response. This kind of pattern could be used to detect wrong placed mass if developed further.
Lastly, the presence of many harmonics at low speeds across the cases reinforces the observation of slackened springs.

6.2.3 Variation of the temporal signal between the cases

Rather than displaying all the data collected, few temporal graphs are presented to show the noticeable variation of the data for the different use cases.
A relative pertinent variation is visible on the the acceleration for the machine containing the shoes and rotating at 1000RPM shown visible in Figure 6.10.

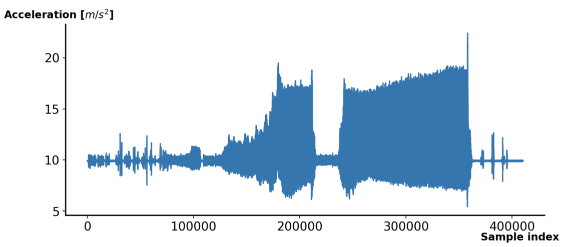


Figure 6.10: Shoes case

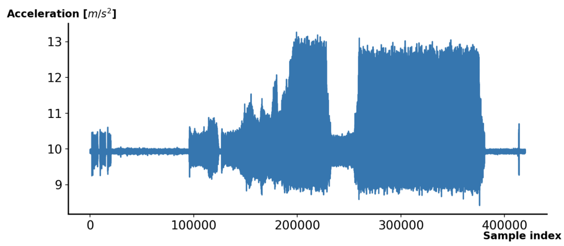


Figure 6.11: Wet towel case

Figure 6.12: Temporal acceleration measurement of a full cycle of a washing machine at **1000RPM** by the side IMU for two cases

There are 4 main trends that deviate from the nominal case visible in Figure 6.5.

First, between the water pumping and the first steady-state, peaks are visible whereas previously only noise was visible. Those peaks in the acceleration are due to the movement of the shoes in the machine. Second, the maximal as visible on the vertical axes, the acceleration is overall higher in magnitude than on an empty machine. The asymmetry of the vibration around the rest value is also to keep on the table. Third, the acceleration and deceleration parts are the phases where the acceleration attains the higher peaks and where the envelope is the more irregular. Lastly, the long steady-state at 1000RPM present in Figure 6.5 is not as visible via the vibrations in the new figure. Instead an almost constant increase of the acceleration is visible.
The second, third and a bit of the fourth points are visible on the data from the wet towels case at 1000RPM displayed in Figure 6.11. But the intermittent oscillation due to the movement of the towels within the drum is at first glance not noticeable due to the low density of the towels compared to shoes. Once the towels are joined together, those same oscillations are visible.

6.2.4 Comparison of the data coming from the horizontal and vertical IMU

Until now only the data from the side IMU is considered. However, the top IMU also has its own collection of information to reveal.

Few differences can be spotted with a fast look at the superposed curves in Figure 6.13. From the batch of measurements performed, at lower rotating speeds the IMU1 on the top of the machine displays higher variation for a given speed than the IMU0 on the side. But this tendency switches as the rotating speed goes beyond 1000RPM. Once the machine is leveled, this effect is drastically reduced. Therefore, to study fault on the washing machine at a lower rotating speed, the top IMU could be capable to give more data whereas, at higher speeds, the side IMU would be better. This extrapolation is done from the measured data of different cases on one machine.

In addition, during the accelerations and deceleration, the top IMU is more sensitive to variations. This effect is visible in Figure 6.13.
But this sensibility brings some drawbacks. For example, the locking, unlocking and pump signature vary a lot with the content of the load as well as within a cycle.
With a closer look at the comparison Figure 6.13, it is noticeable that the mean values are not exactly superposed. This is due to the lack of calibration on both IMUs. Without it, there is some offset and scaling but this does not change the trends.

6.2.5 Metric analysis between speed

As presented in Section 3.2, few metrics can be computed to perform a manual analysis of the failures. These metrics were applied on the whole measured cycle similar to Figure 6.5 for each case. However, to keep the a coherent comparison, the metrics were computed by speed as shown in the Appendix E and were always normalized to observe trends rather than values.

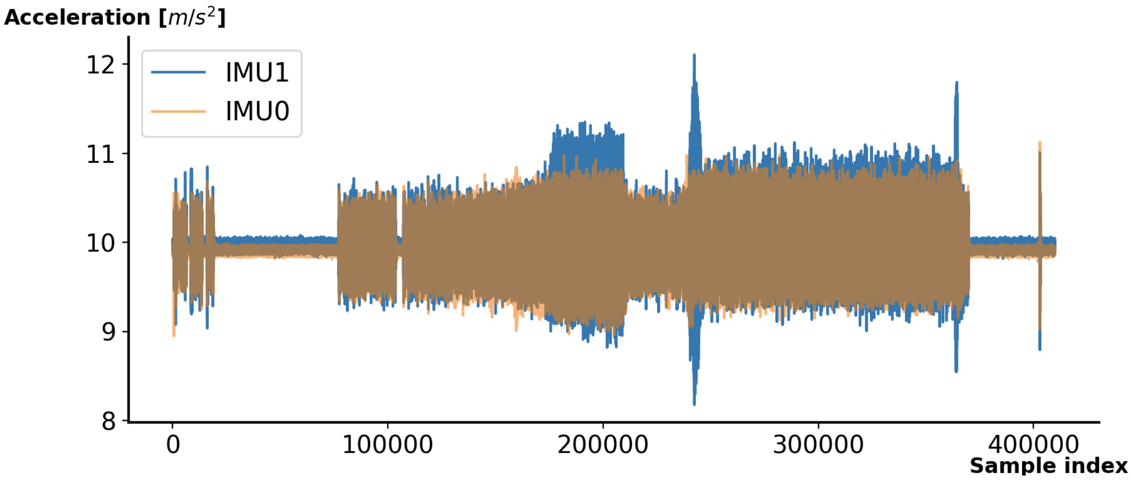


Figure 6.13: Comparison of the acceleration measured by both IMU at 1400RPM for an empty and leveled washing machine. IMU0 is on the side and IMU1 on top of the washing machine

However no additional trend can be extrapolated we some confidence due to the lack of processing of the data (cropping and noise filtering). Since the cycle contains a considerable number of different transients, these metrics are not as significant as the presented in the literature. To obtain better metrics to compare, a first cropping on the locking and unlocking signal shown on Figure 6.5 should be performed to consider the true Peak and Peak to Peak value. The RMS value, Kurtosis and Skewness have little meaning in this type of machine as they work on a multitude of transient signals at each cycle. The metrics tables are available for the different experiments in the Appendix E. Other useful metrics could be developed on the envelope of the signal. The definition of those metrics and the processing of the data to implement in a machine learning algorithm are the next mandatory steps for this project.

With this last short-term perspective on the processing and metrics, this master thesis results are now closed.

Chapter 7

Discussions

Even if the developed device shows promising results, there are still some elements to be highlighted to perform a complete assessment of the capability of the device.

7.1 Comments on the device development

First, in terms of speed transmission. The system samples and stores the data at a rate below 2ms on average for each IMU. However, the transmission and storing of the data between a local server and the cloud have quite some differences. The first and main problem is the number of layers. In this laboratory setup, the information only had a few layers to pass whereas in a realistic environment, the information should travel from the ESP32 to a cloud server passing many more layers. To completely lever this uncertainty, adding more storage capacity to act as a buffer by adding RAM to the microcontroller is a solution to invest further. This would ensure that even if the information is lost during transmission, the data can still be sent by the device until there is an acknowledgment of the reception of the data by the non-local server. But this requires a significant amount of fast writing memory and may increase the cost. Another path consists of regressing to the first iteration and completely rewriting the driving software of the IMU in C++ on the Raspberry PI Zero 2W.

Another point where improvement is achievable is in the storing of the gyroscopic values. In the latest microcontroller software, 3 bytes are allocated to digits under the decimal point of each rotation rate where the noise is up to the unit. This means that for each sampling point of each IMU, 9 bytes are wasted.

The last main shortcoming in terms of device development is the lack of calibration. For now, the data collected is stored as captured by the IMUs. Although, as visible in Figure 6.13 comparing the measurements of both sensors, their measured values suffer from non-idealities of the sensors. This can be reduced drastically, by scaling and offsetting the data via calibration factors. Performing such preprocessing would bring the meaning of the data closer to physical reality.

7.2 Experiments and result comments

As the validation of the device was performed using only one machine, the validity of such a test is not statistically pertinent. Nevertheless, the exploratory part of the capability of the device was achieved. In case of further development, similar experiences must be performed on a variety of other machines. Variability is to be expected, this is why a data-based algorithm is more viable to perform the fault analysis and predictive maintenance rather than a physics-based one. Another complementary approach would be to perform measurements on machines out of the factory to gather nominal data of different cycles. This could be used as a first step training model or to perform some higher-level comparison with non-nominal behaviors.

Given the amount of transient in a washing machine cycle, it may be better to use a Wavelet-based analysis on the signal rather than the FFT on a portion of it since the Wavelet analysis is more suited to transient signals. In a real-life situation, the amount of time where the machine is at a steady-state is not the most frequent. This is why this fault analysis is not as trivial as for industrial rotating machines used to work on a steady-state regime.

This last reflection is even intensified as the moment of collecting the data and extracting the tendencies is supposed to be automated. This is why a data-driven approach using machine learning is bound to be more promising. However, it is not accurate to think that the data can be simply inputted into the algorithm and results will magically appear. The inputs need to be filtered, processed and formatted for such an application. This last part is yet to be performed in the framework of this project. Some possible early processing could use PCA algorithm capable of extracting the main features of large samples of data.

Finally, in terms of results, the results shown in the previous chapter were only done via the norm of the total acceleration captured by the IMUs. Some more precise information can lay in the individual measurement on each axis of the acceleration, such as the axial oscillation of the drum on the Y-axis of the IMUs. Moreover, no data concerning the gyroscope was used. This opens the door for sensor fusion at various levels. The fusion could be performed from the stage of the raw data to the fusion of the metrics for each sensor.

7.3 Commentary about the overall concept

In terms of placement, the actual device supports two IMUs, one on the top and another on the side within the main device as explained in Paragraph 4.2. Even if the top IMU has proved to display some useful information about the state of the machine, its placement is not conceivable. After a discussion with an actor in the washing machine business, the idea of having an external device with two mounting points will not be adhered to by the market. Moreover, the top side IMU is mounted using double-sided tape which is rather unreliable in the long term since there is no mounting point. This is why in terms of realization, only the side inertial measurement can be considered. Alternatively, with the number of bytes freed by this action during transmission, a more accurate IMU could be used to capture a finer range of vibrations at a higher sampling rate.

Talking about the placement of the device, it is important to highlight that this device will not work on constrained machines. In the case of laundromats, machines are often stacked one to another and on top of one another. The vibration response is different due to a lack of possible movement, dissipation, or resonance if there is any object on top of the machine. This could be useful to detect some non-ideal cases of small objects placed on top of the machine but could turn into a constraint if the object is too heavy.
In terms of data storing, the current software captures the data for the whole cycle. In the case of some deployment, the amount of data to store is going to be way too big. Therefore some type of condensed feature must be performed to optimize the ratio between useful features and the amount of data.
Another issue concerning data with such a device is security. The collected information could with more development provide an estimation of the number of people living in a house or even the presence of people in a house. This is why despite the overall protection provided by the WiFi protocols it is mandatory to avoid any possible correlation of the data and the user for a third party.

Lastly, using citizen science, there is no ultimate guarantee of the exactitude of the data. For now, the user needs to input some information such as the maximum speed of the cycle and the type of cycle. If a user selects the wrong parameter the information collected is bound to be inaccurate for the model. This also applies if the user touches the machine or places an object on top of the washing machine. This further supports the use of centers to collect data and to provide a fully autonomous device

Chapter 8

Conclusion and Perspectives

To conclude this master thesis, it has been shown how it is possible to develop a device and the system behind it capable to measure sensitive data to perform a fault analysis. The device was developed almost within the constraining of low cost and non-invasiveness while respecting the requirements sets in the scope of the project. The device is capable to measure 3 degrees of acceleration and 3 degrees of rotations rate on the side and the top of the washing machine. The technical scalability of the system is achievable since most of the design choices were done to allow it.

The results of different use cases showed that the device was capable to detect failures from the motor, the load in the drum, and the springs while being attached externally to the washing machine and only looking at the norm of the acceleration. Moreover, some unexpected signals sources were also captured such as the vibration of the water pump and the movement of the front door switch during the locking and unlocking of the washing machine.

From these observations and studies on the type of failures of washing machines, it is estimated that the device, with further development, can be capable to collect data containing the symptoms of one-third of identified failure modes.

To achieve a device capable to perform by itself fault analysis, a few steps are still to be performed.
First, more data must be captured on other washing machines to ensure that the trends measures are not only visible in the tested washing machine.
Second, the development of some processing on the data has yet to be performed to reduce the amount of data and to extract meaningful metrics needed for an automated fault analysis model.
Third, an updated version of the device in terms of hardware and software must be developed following the guidelines presented in the discussion combined with a migration to a cloud-based system. Once this is done, a first deployment can be performed with a controlled number of devices to collect a significant amount of data and benefit the citizen science framework to develop a machine learning-based fault analysis algorithm.
Following these steps, the device needed to develop a predictive maintenance model can be considered using machine learning to reduce the number of washing machines thrown away due to the difficulties of repairing such large appliances.

Appendix A

Planning of the project

	Task	WorkStream	Domain	Planning																					
				M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15	M16	M17	M18	M19	M20	M21	M22
Phase 1	**Research and Concepts**																								
	Failure Analysis and Repertoriation	Analysis	Mechanics																						
	Specification requirement	Analysis	Mechanics - Electronics																						
	Market Oulook and Research of potential actor	Analysis	Management																						
	Basic architecture Design	Design	Electronics - Software																						
	Database Format Design	Design	Software																						
	Analysis of indicators	Analysis	Mechanics																						
Deliverable	Concept proof design																								
Phase 2	**Failure Recognition**																								
	Developement early hardware design (early sampling)	Development	Electronics																						
	Developement local server system (early storage)	Development	Sofware																						
	Developement link sampling to storage	Development	Software																						
	Developement early processing of the data	Development	Software																						
	Develop local server computation (early analysis)	Development	Software																						
	Design attachment to machine	Design-Development	Mechanics																						
	Select indicator for fault detection	Analysis	Mechanics																						
	Deployment on 1-2 machines	Testing																							
	Test various Failure	Testing	Mechanics																						
	Benchmarking of the device	Testing	Electronics - Mechanics																						
Deliverable	Manual Failure detection on a machine																								
Phase 3	**Scaling and First Deployment**																								
	Update from prototype to actual device	Development	Electronics																						
	Multi device scaling		Software																						
	Cloud storing migration		Software																						
	Cloud computing migration		Software																						
	User interface first concept	Design -Development	Software																						
	Citizen deployment research	Analysis																							
	Deployement < 50 machines to test center	Testing																							
	Deployement < 10 machines to the citizen	Testing																							
	Machine Learning failure detection	Development	Software																						
	Data gathering of failures (forced ones)	Development	Mechanics																						
	Retreive User experience																								
	Optimization of the data storing	Development	Software																						
	Processing of the data for ML algorithm for failure analysis	Development	Software																						
Deliverable	Automatical Failure detection on a machine																								
Phase 4	**Predictive maintenance**																								
	Machine learning RUL development	Development	Software																						
	User interface second iteration	Design -Development	Software																						
	Massive deployement																								
Deliverable	Automatics RUL estimation on machines																								

Figure A.1: Planning of the entire project to perform predictive maintenance on washing machines

Appendix B

BOM of the device

Component	**Quantity**	**Unitary price**	**Total price [€]**
3D printed casing	84g	20€/kg	1.68
Heatsets M2	2	0.08€	0.16
Heatsets M2.5	3	0.08€	0.24
PCB (with BSS138)	1	9.50€	9.5
ESP32	1	8.67€	8.67
Connector Male	4	0.02€	0.08
Connector Female	4	0.02€	0.08
Pin header (1x20)	2	0.25€	0.5
Cable	1.5m	0.48€/m	0.72
MPU9250 module	2	9€	18
Magnets	4	2€	8
USB micro cable	1	6€	6
Screws	5	0.05€	0.25
Total			**53.88**

Appendix C

PCB of the device

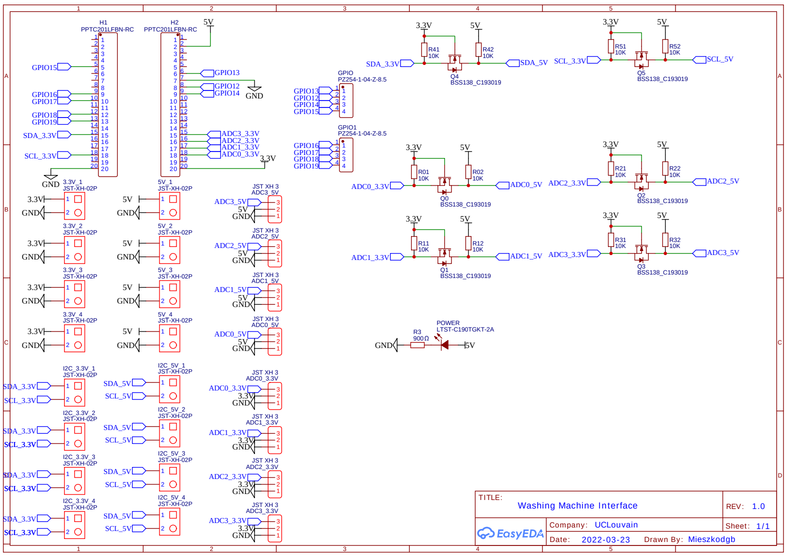


Figure C.1: Schematics of the latest version of the device

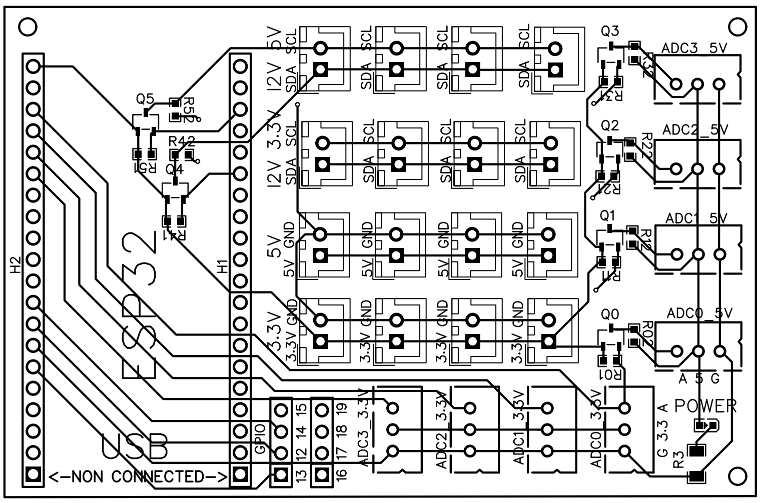


Figure C.2: PCB of the latest version of the device not to the scale

Appendix D

Device and placement

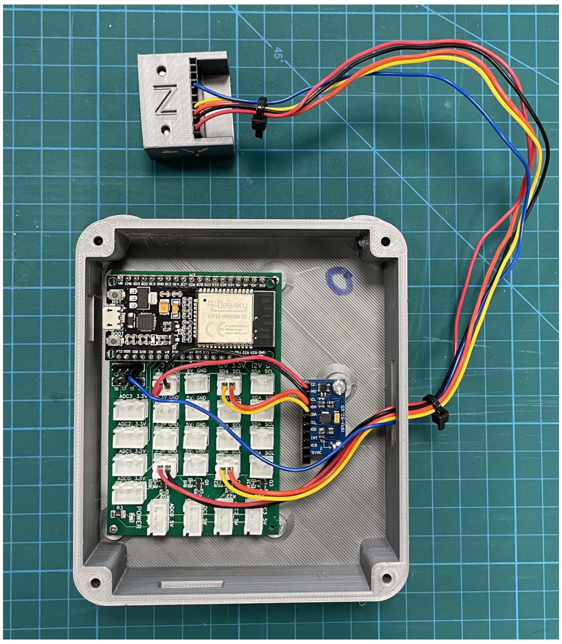


Figure D.1: Front view on the device

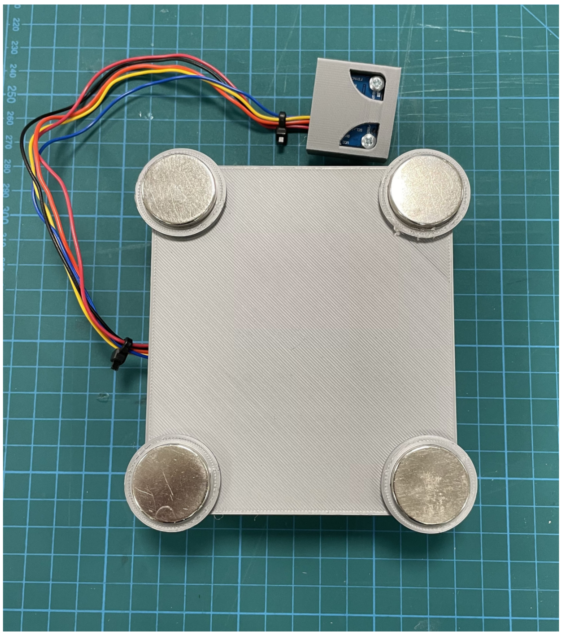


Figure D.2: Back view on the device

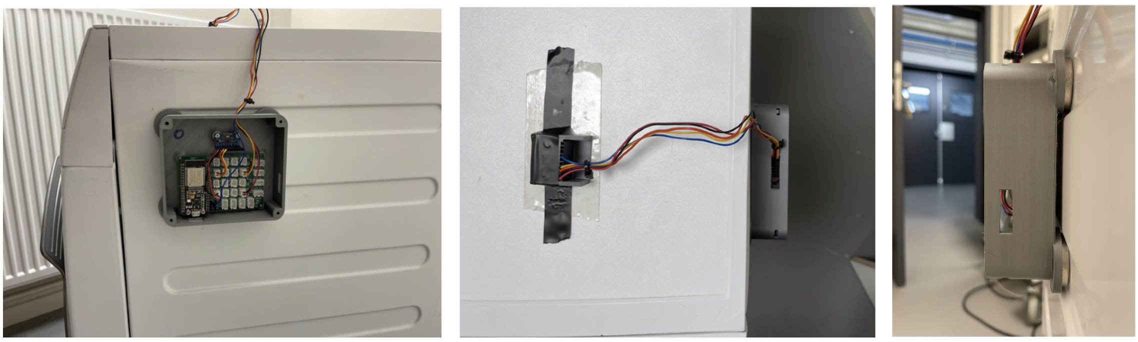


Figure D.3: Placement of the device front the front, top and side view

Appendix E

Normalized metrics from the measurements

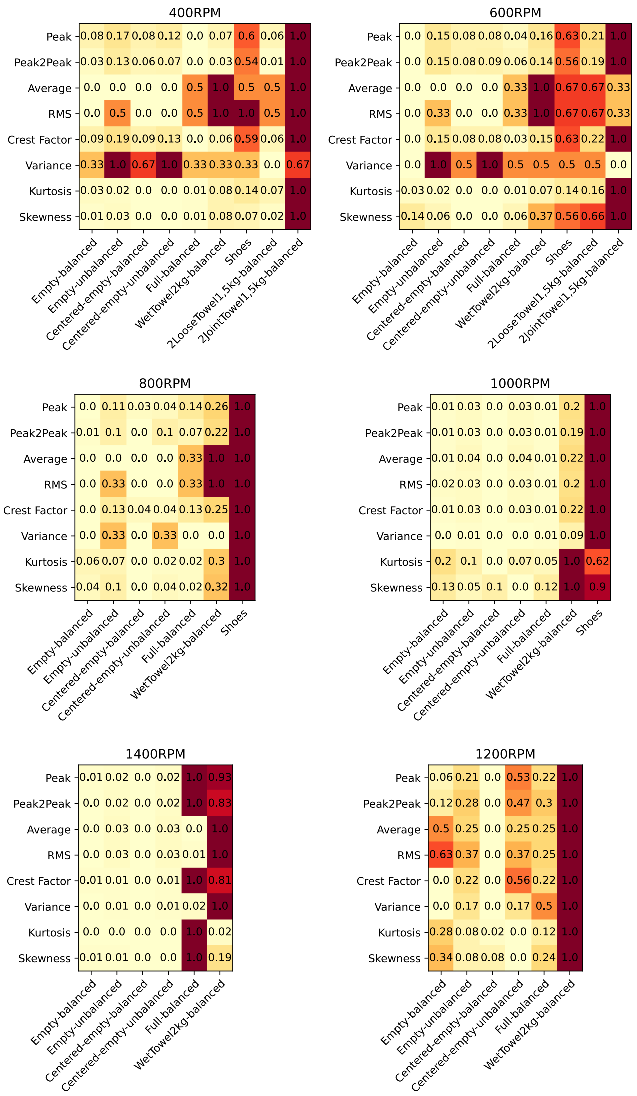


Figure E.1: Normalized metrics of the different use cases at each rotating speed of the washing machine

Bibliography

[1] Vanessa Forti et al. *The Global E-waste Monitor 2020. Quantities, flows, and the circular economy potential.* July 2020. ISBN: 978-92-808-9114-0.

[2] Felice Alfieri et al. "Durability assessment of products: analysis and testing of washing machines". In: (Dec. 2018). DOI: 10.2760/115684.

[3] RWTH Aachen University. *L'Increvable*. http://www.lincrevable.com/en/. Accessed on 2022-05-14.

[4] Melanie Jaeger-Erben, Vivian Frick, and Tamina Hipp. "Why do users (not) repair their devices? A study of the predictors of repair practices". In: *Journal of Cleaner Production* 286 (2021), p. 125382. ISSN: 0959-6526. DOI: https://doi.org/10.1016/j.jclepro.2020.125382. URL: https://www.sciencedirect.com/science/article/pii/S0959652620354287.

[5] Plant Engineering. *Facilities Maintenance 2020*. https://www.advancedtech.com/wp-content/uploads/2020/03/2020-ATS_Plant-Engineering-Maintenance-Study-Results.pdf. Accessed on 2022-05-19.

[6] Espressif Systems. *ESP32 Hardware Reference*. https://docs.espressif.com/projects/esp-idf/en/latest/esp32/hw-reference/index.html. Accessed on 2022-05-19.

[7] Raspberry Pi. *Raspberry PI Website*. https://www.raspberrypi.com/. Accessed on 2022-05-14.

[8] Plant Engineering. *Facilities Maintenance 2019*. https://www.plantengineering.com/wp-content/uploads/sites/4/2019/02/Plant-Engineering-2019-Maintenance-Report.pdf. Accessed on 2022-05-19.

[9] Wikipedia contributors. *Fault tree analysis — Wikipedia, The Free Encyclopedia*. https://en.wikipedia.org/w/index.php?title=Fault_tree_analysis&oldid=1091159956. [Online; accessed 5-June-2022]. 2022.

[10] Ji Min Baek, Sang Hoon Ji, and Ja Choon Koo. "A cost effective on-site fault diagnosis method for home appliance rotor failures". In: *Microsystem Technologies* 26.11 (2020), pp. 3389–3394. ISSN: 1432-1858. DOI: 10.1007/s00542-020-04892-9. URL: https://doi.org/10.1007/s00542-020-04892-9.

[11] Xiaojuan Sun et al. "Measurement and analysis of a horizontal-axis washing machine for low-frequency abnormal noise". In: *2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence, URAI 2016* (2016), pp. 735–739. DOI: 10.1109/URAI.2016.7734103.

[12] Nicola Paone et al. "Fault detection for quality control of household appliances by non-invasive laser Doppler technique and likelihood classifier". In: *Measurement* (June 1999), pp. 237–247. DOI: 10.1016/S0263-2241(99)00007-X.

[13] "Classification of washing machines vibration signals using discrete wavelet analysis for feature extraction". In: *IEEE Transactions on Instrumentation and Measurement* 51.3 (2002), pp. 497–508. ISSN: 00189456. DOI: 10.1109/TIM.2002.1017721.

[14] Brunton Steven L. and J. Nathan Kutz. *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. Cambridge University Press, 2019. DOI: 10.1017/9781108380690.

[15] G. Pittner, R. Barczewski, and J. Buśkiewicz. "Numerical and experimental vibration analysis of domestic washing machine drum". In: *International Journal of Applied Mechanics and Engineering* 17.3 (2012), pp. 765–777. ISSN: 1425-1655.

[16] Ahmet Yörükoğlu and Erdinç Altuğ. "Determining the mass and angular position of the unbalanced load in horizontal washing machines". In: *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM* (2009), pp. 118–123. DOI: 10.1109/AIM.2009.5230027.

[17] Shahood Uz Zaman et al. "E-textile systems reliability assessment-a miniaturized accelerometer used to investigate damage during their washing". In: *Sensors (Switzerland)* 21.2 (2021), pp. 1–16. ISSN: 14248220. DOI: 10.3390/s21020605.

[18] Sanjeev Shakya et al. "An Internet of Things System for a Laundry Monitoring Service". In: *2021 Fifth World Conference on Smart Trends in Systems Security and Sustainability (WorldS4)*. 2021, pp. 235–240. DOI: 10.1109/WorldS451998.2021.9514042.

[19] Yaguo Lei et al. "Applications of machine learning to machine fault diagnosis: A review and roadmap". In: *Mechanical Systems and Signal Processing* 138 (2020), p. 106587. ISSN: 0888-3270. DOI: https://doi.org/10.1016/j.ymssp.2019.106587. URL: https://www.sciencedirect.com/science/article/pii/S0888327019308088.

[20] Luis P. Silvestrin, Mark Hoogendoorn, and Ger Koole. "A Comparative Study of State-of-the-Art Machine Learning Algorithms for Predictive Maintenance". In: *2019 IEEE Symposium Series on Computational Intelligence, SSCI 2019* (2019), pp. 760–767. DOI: 10.1109/SSCI44817.2019.9003044.

[21] A. Géron. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, 2019. ISBN: 9781492032595. URL: https://books.google.be/books?id=HnetDwAAQBAJ.

[22] Ayaz Kafeel et al. "An expert system for rotating machine fault detection using vibration signal analysis". In: *Sensors* 21.22 (2021), pp. 1–15. ISSN: 14248220. DOI: 10.3390/s21227587.

[23] Mustafa Cakir, Mehmet Ali Guvenc, and Selcuk Mistikoglu. "The experimental application of popular machine learning algorithms on predictive maintenance and the design of IIoT based condition monitoring system". In: *Computers Industrial Engineering* 151 (2021), p. 106948. ISSN: 0360-8352. DOI: https://doi.org/10.1016/j.cie.2020.106948. URL: https://www.sciencedirect.com/science/article/pii/S0360835220306252.

[24] Luis Lino Ferreira et al. "Predictive Maintenance of home appliances: Focus on Washing Machines". In: *IECON 2021 – 47th Annual Conference of the IEEE Industrial Electronics Society*. 2021, pp. 1–6. DOI: 10.1109/IECON48115.2021.9589642.

[25] Sensor.Community. *Sensor.Community*. https://sensor.community/en/. Accessed on 2022-05-19.

[26] Paolo Tecchio, Fulvio Ardente, and Fabrice Mathieux. *Analysis of durability, reusability and reparability - Application to washing machines and dishwashers*. November. 2016. ISBN: 9789279607905. DOI: 10.2788/630157.

[27] Paolo Tecchio, Fulvio Ardente, and Fabrice Mathieux. "Understanding lifetimes and failure modes of defective washing machines and dishwashers". In: *Journal of Cleaner Production* 215 (2019), pp. 1112–1122. ISSN: 0959-6526. DOI: https://doi.org/10.1016/j.jclepro.2019.01.044. URL: https://www.sciencedirect.com/science/article/pii/S0959652619300514.

[28] Hongyu Yang, Joseph Mathew, and Lin Ma. "Vibration Feature Extraction and Analysis for Fault Diagnosis of Rotating Machinery-A Literature Survey". In: *Asia Pacific Journal of Multidisciplinary Research* November (2017), pp. 12–14.

[29] Yaguo Lei et al. "Machinery health prognostics: A systematic review from data acquisition to RUL prediction". In: *Mechanical Systems and Signal Processing* 104 (2018), pp. 799–834. ISSN: 0888-3270. DOI: https://doi.org/10.1016/j.ymssp.2017.11.016. URL: https://www.sciencedirect.com/science/article/pii/S0888327017305988.

[30] ENFORMA. *SMART-PDM*. https://smart-pdm.eu/. Accessed on 2022-05-14.

[31] Hiroshi Kanki et al. *Vibrations of rotating machinery: Volume 1. Basic Rotordynamics: Introduction to Practical Vibration Analysis*. Springer, 2017.

[32] Hiroshi Kanki et al. *Vibrations of rotating machinery: Volume 2. advanced rotordynamics: Applications of analysis, troubleshooting and diagnosis*. Springer, 2019.

[33] FES Systems Inc. "Vibration Analysis Services". In: (). Accessed on 2022-05-08, pp. 1–64. URL: https://www.irc.wisc.edu/export.php?ID=158.

[34] M. P. Norton and D. G. Karczub. *Fundamentals of Noise and Vibration Analysis for Engineers*. 2nd ed. Cambridge University Press, 2003. DOI: 10.1017/CBO9781139163927.

[35] Jack D. Peters. *Introduction to Basic Vibration Analysis*. http://www.microtech.co.th/pdf/Vibration_pdf_file/IntroductiontoBasicVibrationAnalysis. Accessed on 2022-04-14.

[36] *Mechanical vibration, shock and condition monitoring — Vocabulary*. Standard. International Organization for Standardization, Oct. 2018.

[37] C. Scheffer and Paresh Girdhar. *Practical machinery vibration analysis and predictive maintenance*. 2008.

[38] SKF USA Inc. *Spectrum Analysis-The key features of analyzing spectra*. `https://www.skf.com/binaries/pub12/Images/0901d1968024acef-CM5118-EN-Spectrum-Analysis_tcm_12-113997.pdf`. Accessed on 2022-05-10.

[39] Aurelien Prudhom et al. "Time-frequency vibration analysis for the detection of motor damages caused by bearing currents". In: *Mechanical Systems and Signal Processing* 84 (2017), pp. 747–762. ISSN: 10961216. DOI: `10.1016/j.ymssp.2015.12.008`. URL: `http://dx.doi.org/10.1016/j.ymssp.2015.12.008`.

[40] Rainer Stamminger et al. "Durability testing procedure for washing machines – approach and first learnings". In: *eceee Summer Study Proceedings* (2017), pp. 1573–1581.

[41] Dario Radečić. *Stop Using CSVs for Storage — Pickle is an 80 Times Faster Alternative*. `https://towardsdatascience.com/stop-using-csvs-for-storage-pickle-is-an-80-times-faster-alternative-832041bbc199`. Accessed on 2022-05-17.

[42] Raspberry Pi. *Raspberry Pi Zero 2 W*. `https://www.raspberrypi.com/products/raspberry-pi-zero-2-w/`. Accessed on 2022-05-14.

[43] Raspberry Pi. *Raspberry Pi 3 Model B+*. `https://www.raspberrypi.com/products/raspberry-pi-3-model-b-plus/`. Accessed on 2022-05-14.

[44] InvenSense. *MPU-9250 Product Specification*. `https://www.digikey.be/en/products/detail/tdk-invensense/MPU-9250/4626449?s=N4IgTCBcDaILYAcCuBOMBWADCAu`
Accessed on 2022-05-19.

[45] Circuit Basics. *BASICS OF THE I2C COMMUNICATION PROTOCOL*. `https://www.circuitbasics.com/basics-of-the-i2c-communication-protocol/`. Accessed on 2022-05-19. Feb. 2016.

[46] Circuit Basics. *BSS138 datasheet*. `https://cdn-shop.adafruit.com/datasheets/BSS138.pdf`. Accessed on 2022-05-19. Oct. 2005.

[47] Pandas Community. *pandas*. `https://pandas.pydata.org/`. Accessed on 2022-05-10.

[48] Alex Albino. *FemtoIO library for the InvenSense MPU9250 chip*. `https://github.com/femtoio/MPU9250`. 2015.

[49] Nordic Semiconductor ASA. *Single chip 2.4 GHz Transceiver NRF24L01*. `https://www.sparkfun.com/datasheets/Components/nRF24L01_prelim_prod_spec_1_2.pdf`. Accessed on 2022-04-11.

[50] Arduino. *Nano 33 IoT*. `https://docs.arduino.cc/hardware/nano-33-iot`. Accessed on 2022-05-19.

[51] MIT Licence. *FreeRTOS™ - Real-time operating system for microcontrollers*. `https://freertos.org/`. Accessed on 2022-05-01.

[52] MQTT.org. *MQTT: The Standard for IoT Messaging.* `https://mqtt.org/`. Accessed on 2022-04-14.

[53] Wikipedia contributors. *Hypertext Transfer Protocol — Wikipedia, The Free Encyclopedia.* `https://en.wikipedia.org/w/index.php?title=Hypertext_Transfer_Protocol&oldid=1091188660`. [Online; accessed 5-June-2022]. 2022.

[54] Wikipedia contributors. *WebSocket — Wikipedia, The Free Encyclopedia.* `https://en.wikipedia.org/w/index.php?title=WebSocket&oldid=1088237509`. [Online; accessed 24-May-2022]. 2022.

[55] Carsten Bormann. *CoAP - RFC 7252 Constrained Application Protocol.* `https://coap.technology/`. Accessed on 2022-05-14.

[56] CARSTEN GREGERSEN. *Websocket vs. MQTT vs. CoAP: Which is the Best Protocol?* `https://www.nabto.com/websocket-vs-mqtt-vs-coap/`. Accessed on 2022-05-19.

[57] Jan Bartnitsky. *HTTP vs MQTT performance tests.* `https://flespi.com/blog/http-vs-mqtt-performance-tests`. Accessed on 2022-05-19.

[58] Bernhard K Aichernig and Richard Schumi. "How Fast Is MQTT?" In: *Quantitative Evaluation of Systems.* Ed. by Annabelle McIver and Andras Horvath. Cham: Springer International Publishing, 2018, pp. 36–52. ISBN: 978-3-319-99154-2.

[59] MQTT.org. *Benchmark of MQTT servers.* `http://www.scalagent.com/IMG/pdf/Benchmark_MQTT_servers-v1-1.pdf`. Accessed on 2022-05-19.

[60] Eclipse Foundation. *Eclipse Mosquitto™ An open source MQTT broker.* `https://mosquitto.org/`. Accessed on 2022-04-14.

[61] Kingston. *A guide to speed classes for SD and microSD Cards.* `https://www.kingston.com/en/blog/personal-storage/memory-card-speed-classes`. Accessed on 2022-05-08.

[62] RWTH Aachen University. *Phyphox.* `https://phyphox.org/`. Accessed on 2022-05-14.

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
École polytechnique de Louvain
Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/epl**