# Digital Twin Based Condition Monitoring System for a Cold Stamping Machine

**Pedro de Almeida Pecora - 41177**

Dissertation presented to the School of Technology and Management of Bragança to obtain the Master Degree in Industrial Engineering. Work developed during the double degree exchange program between the Polytechnic Institute of Bragança (IPB) and the Federal Technological University of Paraná (UTFPR).

Work oriented by:

Prof. Dr. Paulo Jorge Pinto Leitão

Prof. Dr. Cristiano Marcos Agulhari

Bragança

2021-2022

# Digital Twin Based Condition Monitoring System for a Cold Stamping Machine

**Pedro de Almeida Pecora - 41177**

Dissertation presented to the School of Technology and Management of Bragança to obtain the Master Degree in Industrial Engineering. Work developed during the double degree exchange program between the Polytechnic Institute of Bragança (IPB) and the Federal Technological University of Paraná (UTFPR).

Work oriented by:

Prof. Dr. Paulo Jorge Pinto Leitão

Prof. Dr. Cristiano Marcos Agulhari

Bragança

2021-2022

# Dedication

I dedicate this work to everybody that helped me being able to be here and work on this project, in special to my grandmothers, Lourdes, my parents, Benedito and Denise, my sibling, Julia, professors and friends. To my uncle Luiz, godmother Cecira, and cousin Ricardo, for they will be always in my memories.

# Acknowledgement

Polytechnic Institute of Bragança (IPB) and Federal Technological University of Paraná (UTFPR) for the opportunity given, allowing me to gain greater academic knowledge.

To IPB for providing the necessary equipment used to develop this work.

Prof. Dr. Paulo Jorge Pinto Leitão, and Prof. Dr. Cristiano Marcos Agulhari for all the help, support and guidance provided throughout this work.

To Catraport's general manager, Umberto Pellegri, and Catraport as a whole, for the opportunity and for him always being open-minded about improvements in the factory.

To my friends from UTFPR, Heitor, Edean, and Caroline for the support and the good laughs.

To Gregório and Henrique for everything.

To Fernando, for the support with everything ML-related.

To Rebeca, for motivating me and always being positive about the outcome, even in times that I was not.

To Victória and Gustavo, for always being ready to help.

To each and everyone that helped me throughout my thesis, in one way or another.

# Abstract

The fourth industrial revolution aims to connect and digitalize industrial assets. In an industrial environment, energy efficiency and machine health are important aspects of daily life. Energy monitoring systems, used alongside sensory data gathered from machines on the shop floor, can be powerful tools for monitoring the overall machine health. In this work, a data acquisition and monitoring system was implemented in a cold molding machine owned by Catraport, aiming to monitor electrical and vibrational data gathered during the molding process. The sensing equipment used communicates over Wi-Fi and, due to network problems and inconsistency, the factory's computer network had to be modified, allowing for a better connection. The collected data was stored into InfluxDB, with timestamps for each measured input. With the collected data, five different dashboards were created using Grafana, one giving an overall view of the measured parameters, and the others containing more specific information of each parameter, namely current intensity, power, power factor and vibration. From the real-time data, out of control condition testing is carried out using Nelson Rules, imposing that whenever a parameter triggers one or more of the implemented rules, an alarm is shown on the dashboard and an email is sent to the maintenance technician. From the collected data a Machine Learning algorithm, named EECP-CBL, was implemented aiming to predict the next 5 minutes of current intensity, the forecasts also generate alerts for the maintenance team.

**Keywords:** Energy monitoring system; Data acquisition; IoT.

# Resumo

A quarta revolução industrial tem como objetivo conectar e digitalizar ativos industriais. Em um ambiente industrial, a eficiência energética e a saúde das máquinas são aspectos importantes da vida cotidiana. Sistemas de monitoramento energético, utilizados juntamente com dados sensoriais coletados de máquinas no chão de fábrica, podem ser ferramentas poderosas para monitorar a saúde geral das máquinas. Neste trabalho, um sistema de aquisição e monitoramento de dados foi implementado em uma máquina de moldagem a frio de propriedade da Catraport, visando monitorar dados elétricos e vibracionais coletados durante o processo de moldagem. O equipamento de sensoriamento utilizado comunica-se através de Wi-Fi e, devido a problemas e inconsistências, a rede de TI da fábrica teve que ser modificada, para permitir uma conexão melhor. Os dados coletados foram armazenados no InfluxDB, com registros de tempo para cada entrada medida. Com os dados coletados, cinco painéis diferentes foram criados usando Grafana, um dando uma visão geral dos parâmetros medidos, e os outros contendo informações mais específicas de cada parâmetro, nomeadamente corrente, potência, fator de potência e vibração. A partir dos dados em tempo real, são realizados testes de valores fora de controle utilizando as Regras Nelson, implementando um sistema de alerta onde, se um parâmetro acionar uma ou mais das regras implementadas, um alarme é mostrado no painel e um e-mail é enviado ao técnico de manutenção. A partir dos dados coletados, foi implementado um algoritmo de aprendizado de máquina, chamado EECP-CBL, com o objetivo de prever os próximos 5 minutos de intensidade atual, as previsões também geram alertas para a equipe de manutenção.

**Palavras-chave:** Sistema de monitoramento energético; Aquisição de dados; IoT.

# Contents

# List of Tables

# List of Figures

# Acronyms

**AI** Artificial Inteligence.

**ANN** Artificial Neural Network.

**API** Application Programming Interface.

**ARIMA** Autoregressive Integrated Moving-Average.

**CPS** Cyber-Physical Systems.

**CT** Current Transformer.

**DT** Digital Twin.

**I4.0** Industry 4.0.

**IoT** Internet of Things.

**IPB** Polytechnic Institute of Bragança.

**IT** Information Technology.

**KPI** Key Performance Indicator.

**LCL** Lower Control Limit.

**LSTM** Long Short-term Memory.

**ML** Machine Learning.

**RNN** Recurrent Neural Network.

**SSH** Secure Shell.

**UCL** Upper Control Limit.

**UTFPR** Federal Technological University of Paraná.

**VM** Virtual Machine.

**VNC** Virtual Network Computing.

**VT** Voltage Transformer.

# Chapter 1

# Introduction

Industries have constantly been advancing to increase production, get better results, and always try to find a way to improve the actual manufacturing used. Industry 4.0 (I4.0) aligns the automated systems from the third industrial revolution with the available technologies that can collect and share data from different type of sensors and use them to find the best approach for companies.

One of the main components of this revolution is the use of Internet of Things (IoT) devices that provide real-time data [1]. The changes brought by the advent of I4.0 "*led to data-oriented operations that require thoughtful implementation of visualization techniques and software to realize the full value of the data*" [2]. Visualization can be done in any data visualization tool, e.g. Grafana, Tableau, Infogram [3]–[5].

Given the concept of I4.0, the amount of data generated is massive, however, companies are not exploiting this potential to it's full capacity [6]. Machine learning techniques can offer to the I4.0 concept the ability to extract this potential to it's fullest, generating added value to the industry [7].

The use of data monitoring, in conjunction with Artificial Inteligence (AI) algorithms for failure prediction and data forecast, is becoming a powerful weapon when it comes to better understanding the shop floor and energy efficiency.

Along this scenario, Catraport, a company located in Mós, Bragança, Portugal, produces components for the automotive industry using presses, through a cold molding process. The company is always searching for methods to improve its process, e.g., through the continuous monitoring of the electrical and vibrational parameters of their machines, which bring the capability to adapt and optimize the production efficiency. Aiming to a more energy efficient and connected press, the Company wants to have an electrical and physical monitoring system, helping the maintenance team to analyse and even predict failures.

## 1.1 Objectives

The main objective of this thesis is to develop a digital twin based monitoring system of electrical and vibrational parameters of a cold molding machine, using two IoT nodes. Creating dashboards along the way to visualize the data. The collected data is then monitored using an out of control condition test, alarming the maintenance crew if a value goes out of control. Another objective is to apply machine learning techniques to find correlation between failures and the collected data.

This work aims to expand the knowledge about the machine's electrical and vibrational parameters, and to help the maintenance team in case of failures that may occur. With this objectives in mind, the problem will be divided in the following specific assignments:

- Study of the existing IoT energy monitoring devices and IT infrastructures.
- Develop a system architecture based on DT to solution the problem.
- Installation of the sensing components.
- Creation of a platform to visualize gathered data, using dashboards.
- Develop and implement the monitoring service using the selected Nelson's rules.
- Understand and develop the needed knowledge in data analysis.

- Study Machine Learning techniques for failure prediction and data forecasting.
- Install, test, and validate the system architecture and algorithms at the plant.
- Develop a manual to help the user get familiarized with the developed solution.

## 1.2 Document structure

In addition to this introductory chapter, this document is structured as follows, Chapter 2 presents the related work and the state of the art in the field. Chapter 3 presents a description of the case study, its requisites and needs. Chapter 4 describes the adopted system architecture used for the problem, along with a description of the components and technologies used in each step. Chapter 5 implements the system architecture, describing results and problems faced during the implementation. Chapter 6 does an exploratory analysis on the machine data, and presents different algorithms to forecast current intensity to be detected by out of control sample tests. Finally Chapter 7 rounds up the masters thesis with the conclusions and points out future works.

# Chapter 2

# Related work

This chapter briefly describes the related work in the field of Industry 4.0 (I4.0), Internet of Things (IoT), Digital Twin (DT), condition-based maintenance, energy monitoring, and Machine Learning (ML) techniques required for developing the solution.

## 2.1 Industry 4.0

The concept of I4.0 was first introduced in Germany, in 2011. It was a part of a government sponsored initiative to increase the national industry competitiveness. Quickly becoming a hit rather than a hype [8], [9]. While the first three industrial revolutions are based on mechanisation, electricity and Information Technology (IT), the forth industrial revolution brings the use of IoT and Services [10], and Cyber-Physical Systems (CPS) to the manufacturing environment.

Nakagawa, *et al.* [11], presented the main goals of I4.0 as: multidimensional cross-integration of IT systems; consistency of engineering over the lifecycle; creation of lot size one to small lots of customized products; and new work-related social infrastructures.

Following this concept main goals, Kahn, *et al.* [12] states that there is no clear definition of I4.0, but the concept is based on six design principles, as follows:

- *Interoperability*: Ability to communicate CPS, IoT devices, factories and human using IoT technologies;

- *Virtualization*: Transformation of physical processes monitored by CPS to digital models and simulations;
- *Decentralization*: CPS's have the ability to make decisions without the need of a central command;
- *Real-Time Capability*: Collection and analysis of data in a real-time for failure prevention and higher production efficiency;
- *Service Orientation*: utilization of CPS, factory and human services in a Service Oriented Architecture environment to facilitate decision making for managers, operators and customers;
- *Modularity*: Easier scalability of systems and modules by the addition of new machines, without changing existing modules or configurations.

Figure 2.1 illustrates the evolution steps and great technologies breakthroughs that made possible each industrial revolution, where for I4.0, it connects the automation level of the third revolution with the emergent with IoT, CPS and smart technologies, such as cybersecurity, cloud-computing, addictive manufacturing, augmented reality, big data and analytics, artificial intelligence, Blockchain, simulation and modelling, and collaborative robots [13].

The concept of Industry brings a quality-of-life improvement to the industries. In an everyday more connected world, being able to have a live description of the the overall factory health, production and expenses anywhere, along with the ability to change operational parameters dynamically can drastically increase a industry performance. Digitalization of assets, both data-wise as structure-wise enables the creation of said asset in a simulated environment, allowing for riskier approaches and layouts in search of a more efficient industry with no detriment to the actual layout.

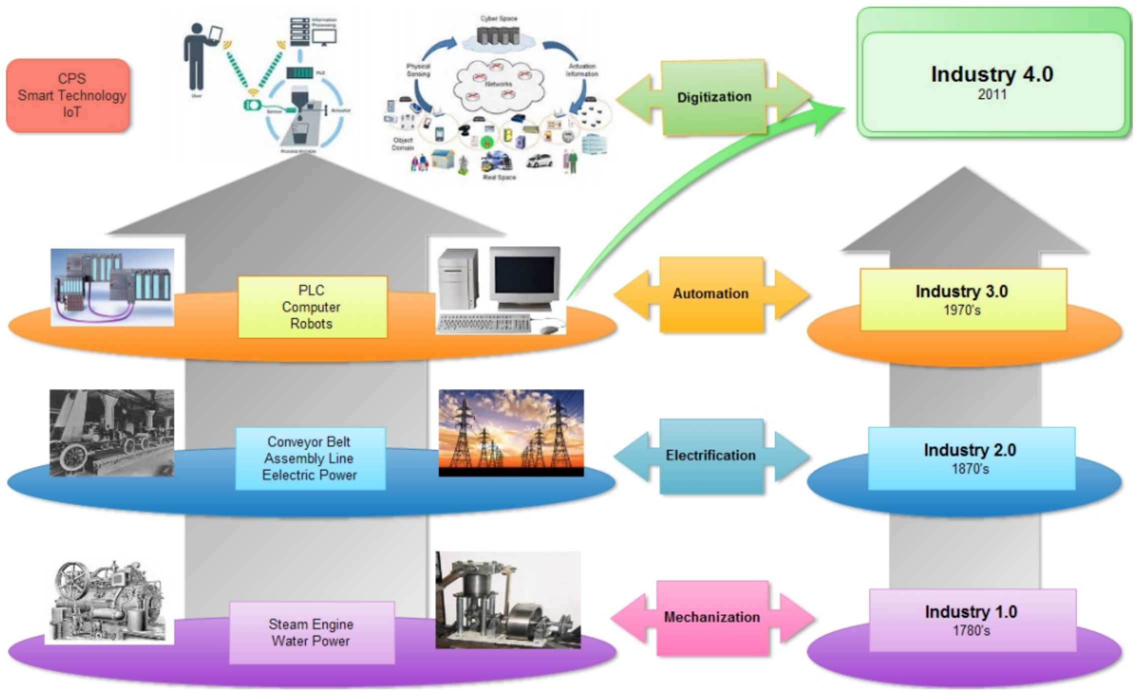


Figure 2.1: Industrial revolutions and their technologies [12].

## 2.2 Internet of Things

Internet of Things is the technology that allows for the connection of everyday objects to the internet. It can connect from household appliances to vehicles, and other electronic devices together on the network, which brings humans to a more connected and intelligent life [14]. The main strength of the IoT idea is the high impact it will have on various aspects of everyday life and the behavior of potential users [15]. Basically, anything is a "Thing" for the IoT concept, e.g. buildings, human beings, machines.

The IoT concept is experiencing exponential growth both in industrial and academic areas, representing one of the most disruptive technologies of this century [16], and fuelling the most important industrial revolution so far of the 21st century. IoT solutions can provide to more than just industrial environments, such as e-health, comprising of monitoring and home care applications, smart cities, smart fleets [17].

There already is a lot of economic growth with IoT-based services. According to

Statista [18], by the year 2030, the number of connected devices worldwide will be around 25.4 billion, that is around 5.8 devices per capita, Figure 2.2 illustrates the growth from 2019. The US National Intelligence Council stated that "*by 2025 Internet nodes may reside in everyday things – food packages, furniture, paper documents, and more*" [19]. Philipp Wegner [20] forecasts that by 2025, the global expending on enterprise IoT technologies will be 361 billion U.S. dollars. The IoT technology is estimated to create an annual economic impact in range the of 2.7 trillion to 6.2 trillion U.S dollars by 2025 [21].

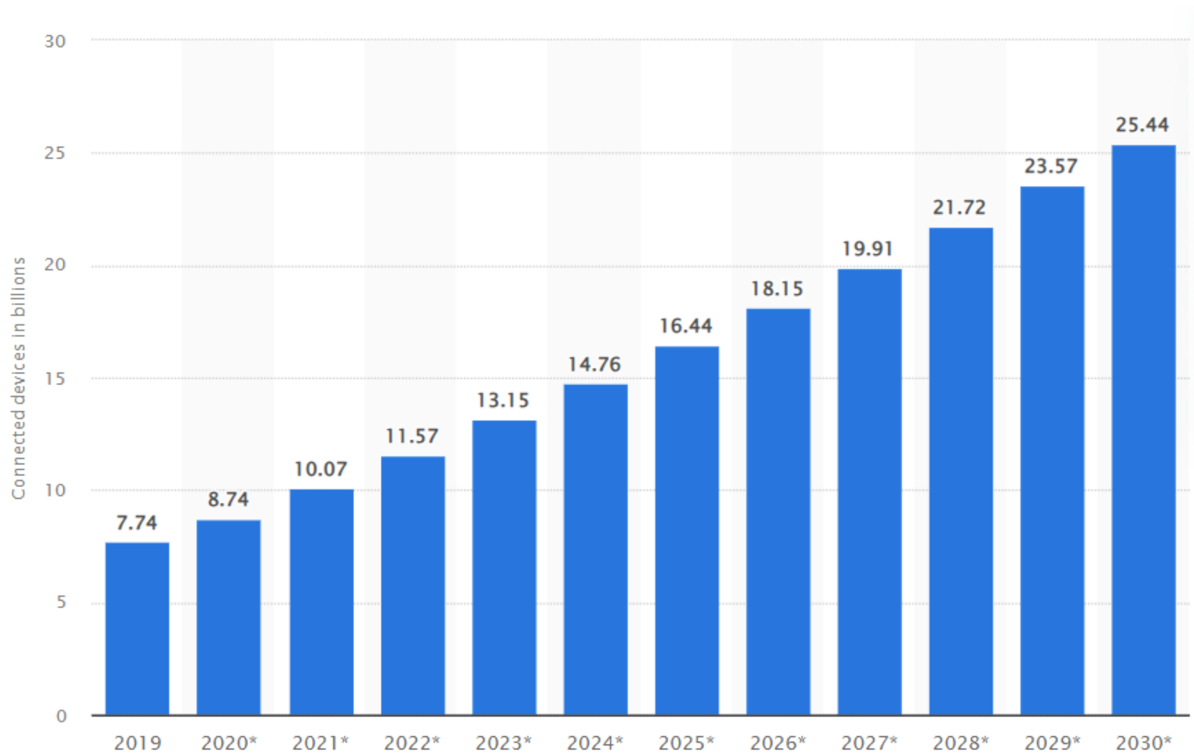


Figure 2.2: Number of IoT connected devices worldwide [18].

The IoT concept can be divided into six main elements [22]:

1. Identification: identification methods to provide clear identity for each object within a network;

2. Sensing: Being able to sense data from the environment and send back to a database or cloud is crucial for the IoT technology. Single board computers with sensing accessories are the most utilized form of IoT products;

3. Communication: The utilization of communication protocols such as Wi-Fi, Bluetooth, RFID, HTTP enables the connection of multi heterogeneous devices together;

4. Computation: Hosts all the computational power of the IoT device, serves as the brain behind all the gathered data;

5. Services: Serves as a bridge between the real world objects and the virtual representation of them, centralizing raw measurements to be processed by the IoT application;

6. Semantics: Smart extraction of knowledge from data for decision making.

The IoT is a huge contribution to mankind and may serve as a fundamental pillar for new technologies and services that have yet to emerge, as everything that exists is now becoming smart [23]. Aiming to narrow the gap between the real and connected worlds.

## 2.3 Digital Twin

The I4.0 revolution shook the industrial world and triggered a digital transformation world-wide. Upon relying on the I4.0's principle of virtualization, the DT concept began gaining momentum.

It evolved from a industry presentation in 2002 by Dr. Michael Grieves [24] of a basic conceptual model of a product's lifecycle management, named Mirrored Spaces Model, to a "*physical product in real space, virtual product in virtual space and the connection of data and information that ties the two spaces together*" [25]. According to Grieves, the digital twin concept have different manifestations, namely Digital Twin Prototype (DTP), Digital Twin Instance (DTI), and Digital Twin Aggregate (DTA), all being operated in a Digital Twin Environment (DTE) [24]:

Digital Twin Prototype - the prototypical physical asset, having the necessary information to describe and create a virtual version of said asset, such as a fully 3D representation of the asset, bill of processes, bill of materials, bill of services and bill of disposal.

Digital Twin Instance - corresponds to the link of the DT and the physical product, throughout the life of the physical product. The Digital Twin Instance may contain the same information as the Digital Twin Prototype, if not even more information.

Digital Twin Aggregate - an aggregation of all the DTI's, not needing to be an independent data structure. For example, the DTA can be a computational structure that can monitor the DTIs.

Digital Twin Environment - integrated, physics based application space for operating on DTs for multiple purposes, namely predictive and interrogative. For predictive purposes, the DTE would make the DT in charge of predicting future behaviour and performance of the product. The interrogative purpose of a DTE is correlated to the DTA manifestation, where a DT would constantly be asked about their status related to any characteristics that the physical asset may have.

However, many other definitions of DT can be found in the literature, even that sometimes incorrectly [26]. As an example, [27] defines Digital Twin as mirrored products to be as well subjects of virtual space reproduction to gain the same benefits.

Technologies associated with Digital Twin may vary widely. Pires, *et al.* [28] reviewed over 200 papers containing the term DT and from the dataset, the most common technologies are: CPS, AI, ML, Deep Learning, Virtual and Augmented Reality, Big Data, IoT, Simulation, Data Analytics and Cloud. These technologies have a great intersection with the technologies used in I4.0, revealing that DT and I4.0 have a symbiotic-like relationship.

In manufacturing, Digital Twin offers simulation and optimization of the production system, from a single component to a whole production line and its logistical aspects [29]. While in the industry subject, DT and DTA's can aid maintenance strategies such as Condition-based Maintenance and Predictive maintenance [30] by monitoring the assets' condition, using statistical analysis and AI algorithms for more detailed information and even intelligent decision making. Identifying the impact of state changes on processes, evaluating anticipatory maintenance measures and machine conditions, and processing machine data during its different life cycles [31]–[33], are also some ways that DT can effectively support and revolutionize the maintenance sector, including and not limited to reducing production downtime, rescheduling and speeding up preventive maintenance stops.

As an example application of Digital Twin towards predictive maintenance, [34] developed a DT copy of a CNC tooling machine with sensory data, such as acceleration, applied force, and acoustic emission during the CNC process; the predictive maintenance revolve around forecasting the remaining useful life of cutting tool.

## 2.4 Condition-based maintenance

Condition-based maintenance is an intriguing field that saves the industry from heavy losses occurring during machine breakdowns, by determining the optimal time to perform maintenance of a certain asset [35], [36]. The determination of the exact time where a failure started and predicting the time until a complete failure requires a great amount of data and effective algorithms [37].

The use of sensoring equipments, historical data, and supervised learning processes may increase the chance of a Condition-based maintenance system to be assertive on its purpose [38]. The MIMOSA organization [39] proposed an Open System Architecture for Condition-Based Maintenance (OSA-CBM), divided into seven modules [40]:

- Sensor Module: provides the system with digitized sensor or transducer data;
- Signal Processing Module: performs basic signal operations and feature extractions. Alongside the inputs from the sensor module, can extract frequency, the filtered data, among other features;
- Condition Monitoring Module: receives data from the modules and compares the data with expected values, if necessary it generates alerts;
- Health Assessment Module: receives data from the previous modules and determines if the health of the monitored asset, system, or sub-system has degraded. The module may even suggest fault possibilities;

- Prognostic Module: active prediction of the future system condition. It does this by being connected with all the previous modules;
- Decision Support Module: provides recommended maintenance actions or alternatives on how to run the system, sub-system, or asset;
- Presentation Module: from the connection to all previous modules, the presentation module is the way the suggestions and features extracted from the data to be presented.

Condition-based maintenance techniques can be used to monitor several equipment conditions, such as vibration monitoring, sound or acoustic monitoring, temperature monitoring, electrical monitoring, and physical condition monitoring [41]. Galar *et al.* [42] presented a vibration monitoring system to analyse the overall health of a paper press machine. For this purpose, 544 sensors were installed to monitor the vibration of the rollers' bearings, with all the data being stored on-line and shown via a visualization software.

## 2.5 Energy monitoring

Energy monitoring, whether it is aimed at electrical consumption or energy efficiency, is an area of high interest for industry. The industrial sector uses more energy than any other end-user sectors, consuming roughly 37% of the world's energy production [43]. Where, depending on the type of industry, induction motors can take up to 70% of all electrical loads [44], this tends to plummet down the energy efficiency if the loads are not balanced, since induction motors generate a great amount of reactive power. With a rapid growth in energy demand and the most being generated via nonrenewable fossil fuel, fuel prices tends to increase.

Figure 2.3 compares the industrial energy consumption percentage by fuel type in 2020 and the forecast value for 2050 [45]. The annual growth rate is expected to be 1.4%, the energy consumption in 2050 is estimated to be around 1.5 times higher than in 2020.

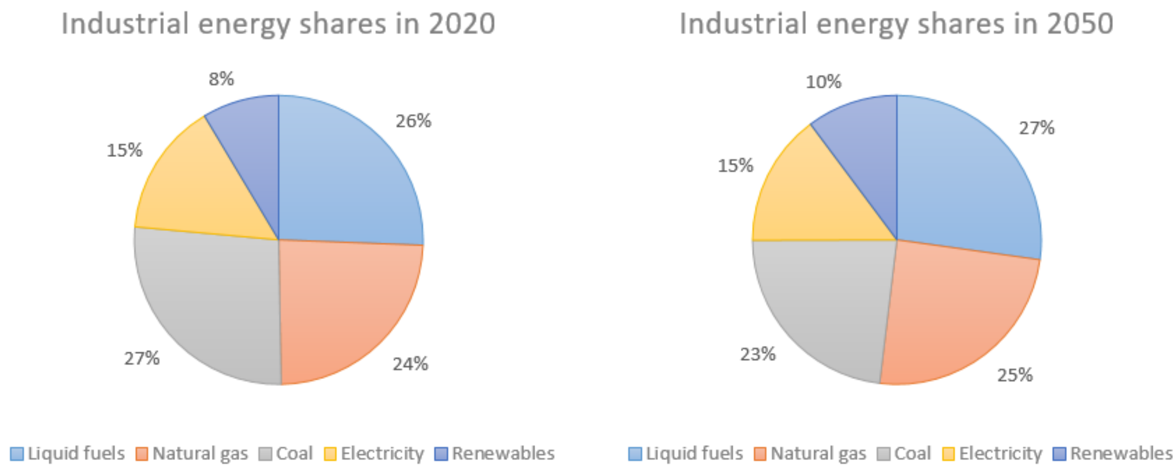


Figure 2.3: Industrial energy consumption share in 2020 and 2050, adapted from [45].

Due to electricity's nature, being an abstract and invisible force, it can be defined in many different ways, i.e. strategic material, commodity, and social necessity [46], therefore, the final consumer doesn't have a concrete way of knowing it's expenditures until a bill is filled. This implies on the accurate reading of the energy provider and nothing else. The gap presented by this approach is where a private energy monitoring system may come in hand, allowing for the consumer to make decisions for reducing operational costs and improving the energy efficiency, since providing an energy feedback can reduce energy consumption by 20% [47].

The main component of a energy monitoring system is the measurement device, however these so called smart energy meters can be high cost. For these devices to do their calculations and communications, they need two electrical quantities: current and tension. Bandarra *et al.* [48] states that is possible to separate monitoring systems in two groups:

- Non-Intrusive Monitoring: Aims to minimize the number of metering equipment while requiring advanced algorithms for allowing the user to know global and individual consumption;

- Intrusive Monitoring: The more conventional method on the marked. With a higher number of metering equipment, this monitoring system can be expensive, while providing a greater data reliability.

Current and tension transformers (CT and VT) are the most common metering equipments used, as they allow for calculations of every other electrical quantity. By reading the changes in the magnetic field, the devices can effectively calculate the current and tension of the measured equipment. Mudaliar and Sivakumar [49], developed an IoT based energy monitoring system using Raspberry Pi in a company that manufactures both high and low tension panels. The work described in [50] monitored heating and cooling machines voltage, current, and power factor in a vehicle testing facility, to reduce the upkeep cost of said machines by shifting the operation time from 8am until 4pm to 1am until 9am.

Many other examples of successful applications of energy monitoring to cut energy costs, give a detailed insight on energy consumption, measure energy efficiency [51]–[53]. These applications contributes to the overall idea that in an industrial environment, custom energy monitoring systems may aggregate more value to a whole production process, e.g. by minimizing costs or rescheduling activities to low cost energy hours, rather than the monthly readings given by the energy provider that are often times few and lacking information.

## 2.6 Machine Learning techniques

Arthur Samuel, considered one of the pioneers in the field of ML, defined ML in 1959 as the "field of study that gives computers the ability to learn without being explicitly programmed." [54]. The desire of constant learning is natural to human beings, so it is logical for it to be made an essential aspect of machines as well [55]. This field revolutionized the way data is explored by providing concise insights about data through the extraction of important snippets of information that humans can interpret more intelligently. Many industries applies machine learning to extract relevant data [56]. Shinde and Shah [55] divides the application of ML into five distinct domains:

- Computer Vision: Comprises of object recognition, detection, and processing;

- Prediction: Many sub-domains, such as classification, analysis, time series, and recommendation;
- Semantic Analysis: Process of relating syntactic structures from words, sentences, paragraphs to the level of writing as a whole;
- Natural Language Processing: How can computers process natural language, being able to identify contextual nuances;
- Information Retrieval: Science of searching information in a document, searching documents and metadata for the desired information.

While being divided into five domains, the ML concept is wide and spans over various types of algorithms, that may have uses across many domains, Figure 2.4 presents the branches of the ML concept. From each subdivision, examples of algorithms that have a common denominator sprouts. Data scientists say that there is no one-for-all best algorithm, always depending on the type of problem in hand and adopted approach.

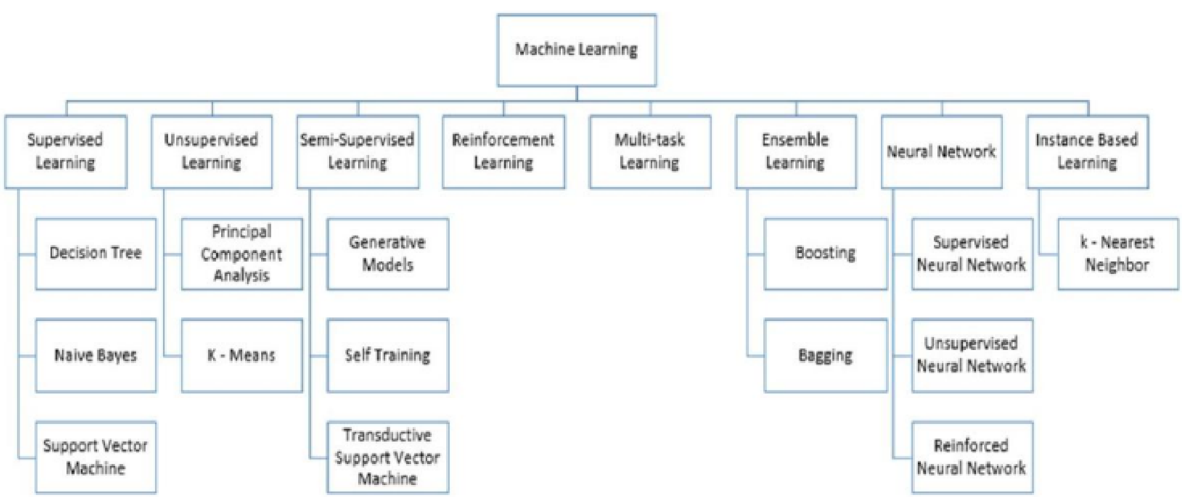


Figure 2.4: ML applications and algorithms [56].

The clear definition of Neural Networks, Artificial Intelligence and Machine Learning can be a little fuzzy, and although an extensive review of all ML's branches is out of the scope of this work, it is important to discuss a little further about Neural Networks, also known as Artificial Neural Network (ANN), in an industrial point of view. Neural Networks are based on the human's greatest tool, the brain. As an analog to biological neurons, Neural Networks, have as it's building blocks the artificial neurons. Each neuron represents a processing unit for an algorithm and is a part of the neural network [57]. Figure 2.5 illustrates the similarities between a biological neural network and an artificial one. The neuron presented is the perceptron, being the building block of a neural network, where $x$ is the input, $w$ represents the weight of each connection and $\sigma$ is an activation function of $\epsilon$, the activation functions may vary, i.e. simple threshold based functions, sigmoid, gaussian, and sinusoidals.

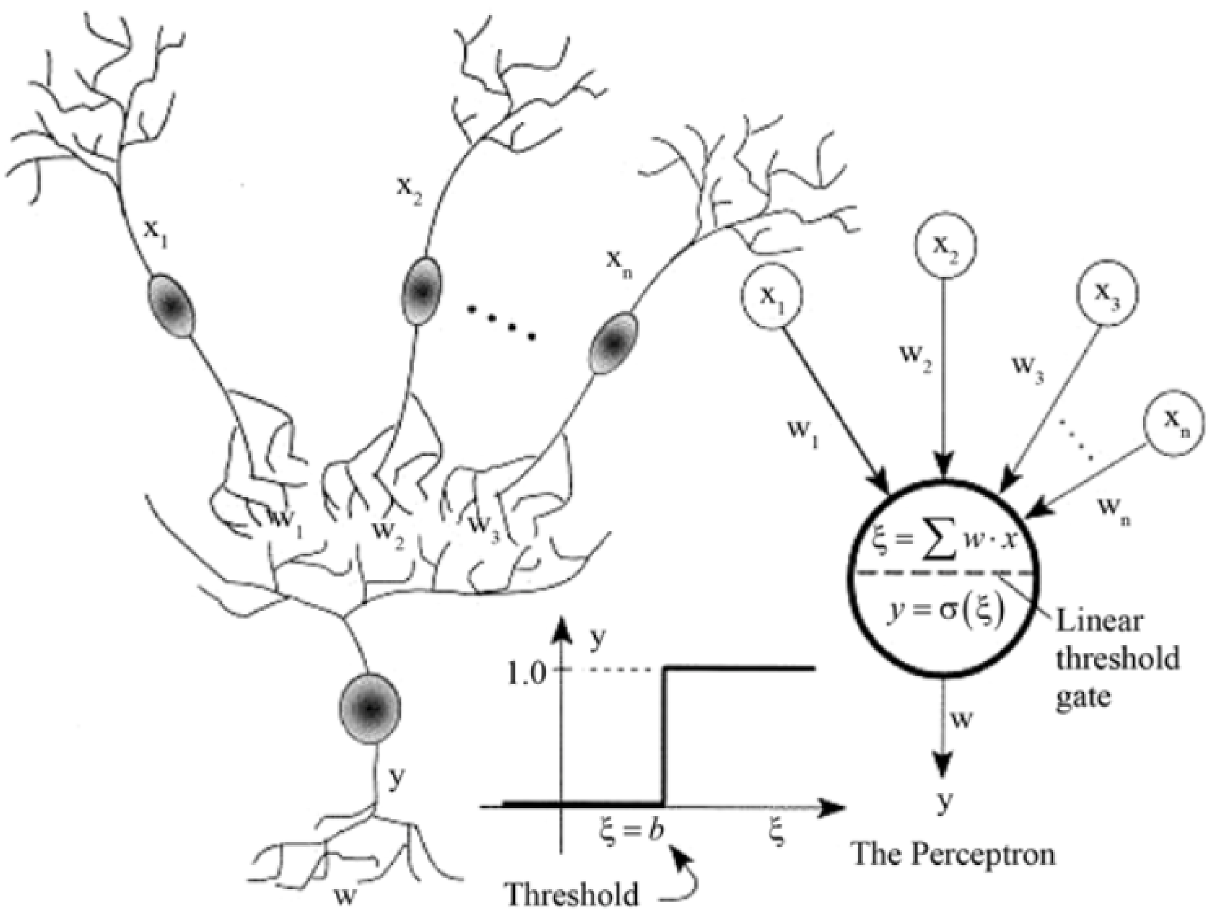


Figure 2.5: Comparison of a biological and an artificial neuron. [57].

In the Artificial Neural Network topic, a greater subdivision can be made, similar to the division made in [56], where different methods can group different algorithms that perform under a certain sets of conditions. The popular methods are supervised, unsupervised, semi-supervised, reinforcement, and transfer learning [58]:

- Supervised learning: Ideal for small yet labeled datasets that humans cannot perceive the relationship between the data, in other words, the expected output is known in the training dataset. Vastly used in classification and prediction algorithms [59].
- Unsupervised learning: Uses large sets of unlabeled data, where the expected output is not known either in the training or testing dataset, the algorithm must create relationships between the data and find patterns. Clustering techniques falls in this group.
- Semi-supervised learning: Manual labelling of data tends to be costly. The semi-supervised method takes advantage of few labeled data, applying a model created with the labeled data to label to extrapolate the relationships found into the unlabeled data. A new ML algorithm can then perform in the whole dataset.
- Reinforcement learning: A reward-based type of ML operation, where a set of rules determine either the algorithm will be rewarded by their actions or not, guiding the algorithm through the steps required to solve the problem. An analogy can be made with a pet being trained, where for each right action it may receive a treat as reward.
- Transfer learning: Equates to using an already developed model for one solution into another related problem, shortening research and deployment times.

Since I4.0, industries have a surplus of data being gathered every second, since the amount of data presents huge opportunities for all sectors and industries wants to explore its potential [60]. In environments like this is where ML thrives. Industrial manufacturing can expect great advances in the field of energy management via the implementation of ML techniques.

In the last years, the field of ML techniques applied to predictive maintenance became a hot topic in the academic area. There is a clear trend of continuous growth in the literature [61], where a whole plethora of ML techniques are applied, such as Autoregressive Integrated Moving-Average (ARIMA), Long Short-term Memory (LSTM), Support Vector Machines, and Linear Regression.

As an example, Paolanti *et al.* [44] trained a Multiclass Random Forest algorithm with a dataset consisting of machine records, fault records, real-time monitored machine conditions, and machine and operator data to predict failures in a cutting machine, achieving an overall accuracy of 95% of in the predictions. Both an ARIMA and ANN implementation is presented in [62], where a PLC controller was used to monitor the energy consumption of a company that produces automation equipment.

# Chapter 3

# Description of the case study

Catraport is a company located in the industrial zone of Mós, Bragança, being one of the companies that belongs to the P&C Automotive group. It was founded in 2015, and started production in 2017. The factory produces metal components for the automotive industries through a cold molding process, being born to attend to the growing demands of two of the group's largest customers. Given it's purpose, Catraport is always seeking for ways to increase their production, e.g. reduce the downtime during maintenance, predict failures to avoid wear of the machine, monitor the machine's health condition.

## 3.1 Cold molding machine

The monitored machine during the study is the Zanni cold molding machine, as illustratred in Figure 3.1. It stamps metal parts to make complex shapes, applying up to 400 tons of force and operating in a range of 1 to 60 strikes per minute. As the machine has various dies, the process can vary in the number of steps it takes to make a final piece, the pressure of each strike, the speed in which the metal sheet is fed to the machine.

The whole cold molding process begins with an aluminum coil being unwound. The continuous sheet passes through a feeding system, that meticulously moves forward the sheet to be pressed by the Zanni. The leftover metal is then rewound, while the usable parts are ejected from the sides of the press. However, some metal scraps may eject from

Figure 3.1: Zanni Cold molding Machine

the the sides too, depending on the type of piece being manufactured. A Top-Down view of the whole process can be seen in Figure 3.2.

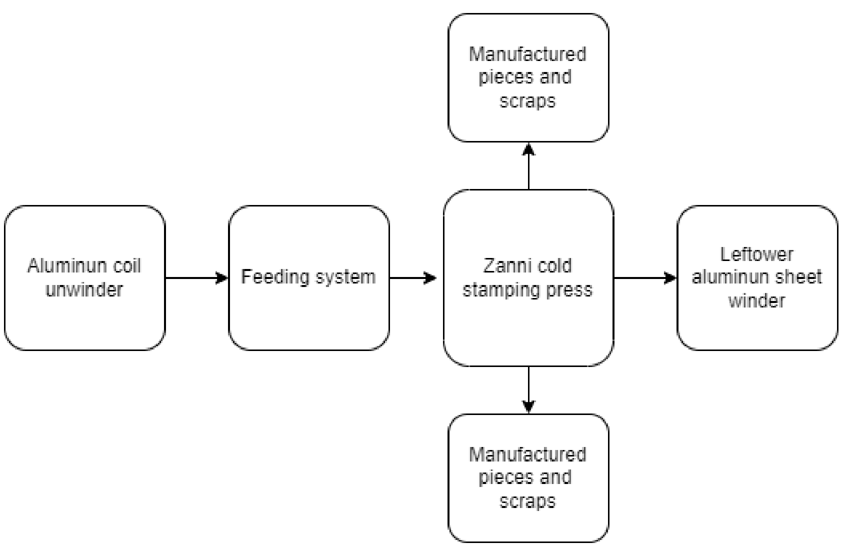


Figure 3.2: Zanni process - a top-down view.

Catraport's Zanni operational parameters are monitored and shown in a live dashboard, yet no storage is made by Catraport regarding these parameters. Figure 3.3 contains data about said operational parameters related to internal pressure, oil temperature and the speed in which the press head is moving. Upon further inspection, no data about electrical consumption or vibration was being collected by Catraport.


Figure 3.3: Operational parameters dashboard.

## 3.2 IT infrastructure

Since all the monitoring devices transmit data over the Wi-Fi, the original IT infrastructure present on Catraport had to be revised. On account of thick, noise-dampening doors and walls, no internet signal could be transmitted from the metrology room, that doubles as the server room, to Zanni's vicinity. Figure 3.4 illustrates the original layout and this problem. From the metrology room, the Wi-Fi signal is dampened while travelling through two thick metal doors that must remain closed all the time: one that gives access to the shop floor, and another one to the metrology room.

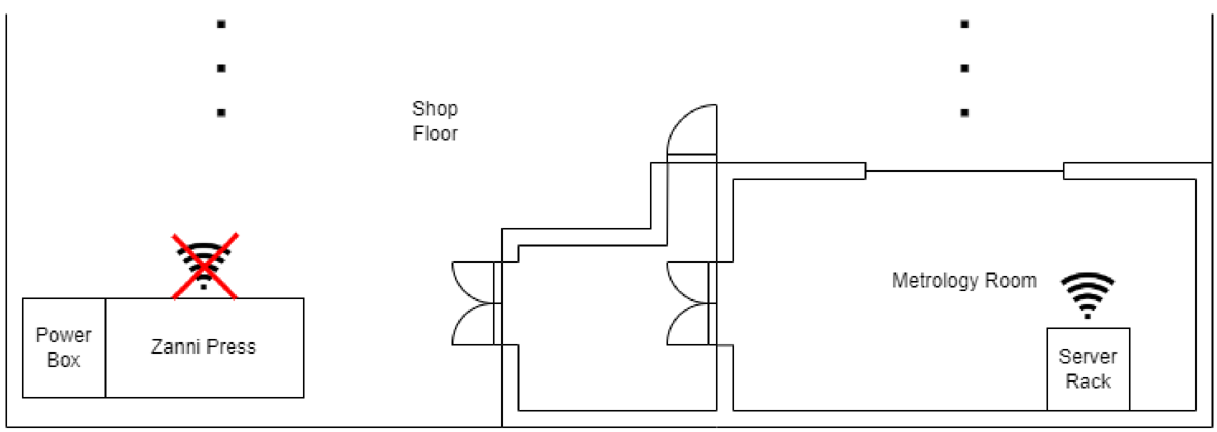


Figure 3.4: IT infrastructure layout. Model not to scale.

# Chapter 4

# System architecture & sensing equipment

In this chapter, the system architecture is presented, and the sensing equipment, used to support the system architecture and adopted approach, are detailed.

## 4.1 System architecture

The system architecture created with the objective of allowing the digitalization is based on the Digital Twin (DT) concept, illustrated in Figure 4.1. This architecture brings the capability to digitalize an physical asset as a virtual model, feeding real-time data gathered from the asset, obtained via sensing equipment to perform condition monitoring and failure prediction.

The *data collection & fusion* module is responsible for the aggregation of the two main inputs of data into the data storage module, namely the electrical and vibrational data. It serves as the middleman from the physical asset's data and the data storage. Using an IoT communication interface, the data is then pushed into the *data storage* module.

Stored data is visualized through curated dashboards that can communicate with the database and provide the maintenance team with an overview of measured parameters.

The *Monitoring & Prediction* module consist of live monitoring of collected data and

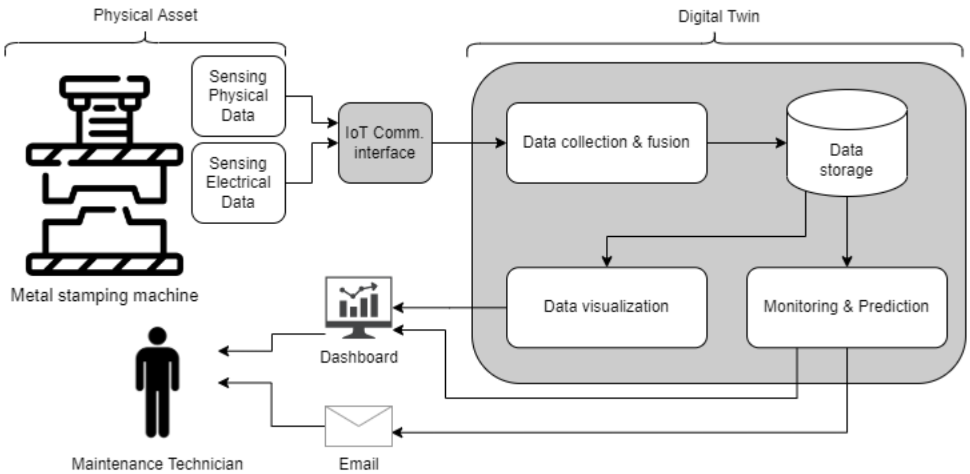


Figure 4.1: An overview on the system architecture.

prediction of future values using linear regression or ML algorithms. The goal is to detect anomalies in the data within a reasonable time frame so that when the displayed values are analyzed, the maintenance team can be contacted, and may get ready to perform the required job. The recommendations given can be as simple as alerts on a dashboard and emails, or even more advanced predictions and trends in the data.

## 4.2 Sensing equipment

Monitoring the electrical and physical parameters of the machine is of key utility for the maintenance crew. In this case study, the technologies used for monitoring, namely, IoTaWatt, and a custom IoT node to gather vibrational data, will be discussed further in their respective subsections.

### 4.2.1 IoTaWatt

The IoTaWatt is an open software and hardware Wi-Fi electricity monitor [63], that can monitor up to 14 different circuits, given its 14 inputs for current transformers (CT) in a wide range of current intensity. It can store, reshape, and monitor data to the native Web server via a friendly Graphical User Interface, where all the installed CTs and the reference power transformer must be configured in the device configurations.

There is also the possibility to transmit collected data to databases, called in the interface as data uploaders such as Emomcms, PVoutput, and InfluxDB. The IoTaWatt device installed is presented in Figure 4.2, and an example of a CT can be seen in Figure 4.3.

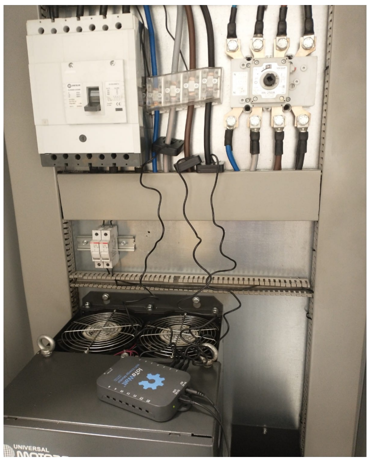

Figure 4.2: IoTaWatt device measuring each of Zanni's phases.

Figure 4.3: IoTaWatt's current transformer. [63]

For a precise measurement, IoTaWatt uses a power transformer as a reference. This input is used in the energy monitor's different calculations, along with the raw current intensity read by the CT, to transform this 2 values into the electrical quantities such as:

- Active Power [W];
- Reactive Power [VAr];
- Apparent Power [VA];
- Power Factor;
- Current Intensity [A];
- Reference Tension [V].
- Reference Frequency [Hz];

The choice of IoTaWatt as a device to collect energy data is due to several factors. One being the fact that it is open source, allowing greater access to documentation and understanding of the inner workings of the device. Since the device communicates over Wi-Fi, no additional cables are required to transmit the data, which is a big advantage as IoTaWatt's installation is mainly *in loco*, as close as possible to the data source. Native communication with InfluxDB enables a direct data streaming path as well.

The IoTaWatt device was installed inside Zanni's power box, as mentioned earlier. In this way, the TDC DE-10-09 9-volt reference input is directly connected to one of the three measured phases, given the short cable length present in the model. This reference tension is required by the device to do all the needed calculations, alongside the CT input.

A SCT019-000 CT is connected to each measured phase, shown as gray, brown, and black cables in Figure 4.2. The current transformers are in turn connected to IoTaWatt for further configuration. For the data to be read correctly, both the 9 volt reference input and CT must be specified in the input field on the IoTaWatt web server. Figures 4.4 and 4.5 show the configuration for the voltage transformer and the current transformer, sequentially.

Since the machine uses three-phase power, each configured CT must be assigned a phase A, B, or C. IoTaWatt assumes that the phase connected to the VT is phase A, and the subsequent phases are assigned via trial and error. All calculations inside IoTaWatt regarding the B and C CTs are done by phase-shifting from phase A.

After the inputs have been configured correctly, the output data must also be configured. The user must provide IoTaWatt with all desired electrical quantity, for example, for data monitoring, all electrical quantities read from CT are stored in InfluxDB so that the data can be pushed with the correct measurements and field key-value pair. Figure 4.6 shows a typical configuration for the power output of the first phase or phase A.

Following the configuration of the inputs and outputs, the data uploader, in this case the InfluxDB 1.8, is configured. The IP and credentials to use Influx's API are required too. Measurement values get the same name as the outputs. Custom measurements can be created through the same interface presented in Figure 4.6.

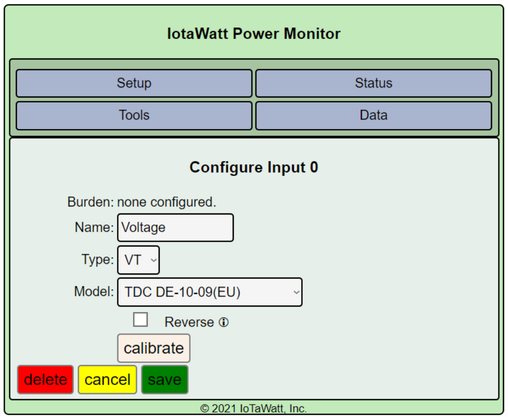


Figure 4.4: Voltage transformer configuration.

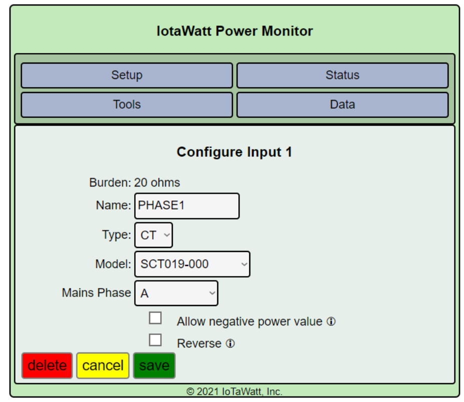


Figure 4.5: Current transformer configuration.

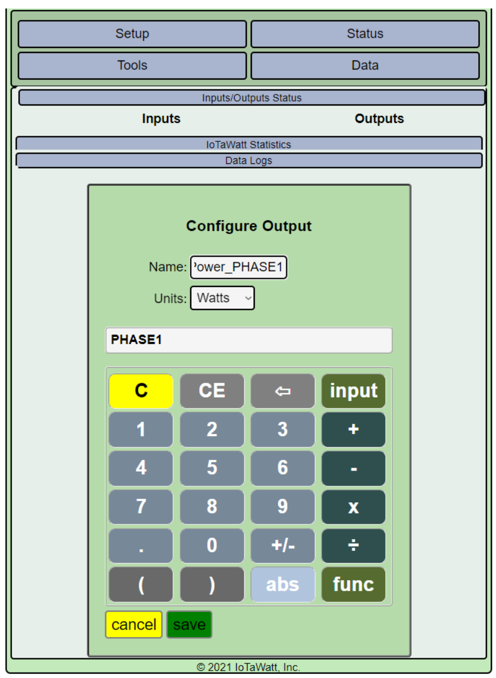


Figure 4.6: Configuring the first phase power output.

### 4.2.2 Accelerometer node

The accelerometer node consists of a ESP8266 connected with an LSM303DLHC (LSM303) compass and accelerometer.

The ESP8266 is a microcontroller with Wi-Fi integrated, built for IoT development [64]. With great flexibility for being both a prototype and a permanent solution to a problem, easiness of code with the Arduino Integrated Development Environment, a small profile, and the possibility for the Inter-Integrated Circuit ($I^2C$) communication protocol, the ESP8266 was the chosen microcontroller for the accelerometer node, to gather vibration data from the Zanni.

As for LSM303, it's a 3-axis accelerometer and magnetometer board made by Pololu [65]. Supports a range from ±2g to ±16g and ±1.3 to ±8.1 gauss having a 16-bit data output for the acceleration, giving a high float point precision, with an error of up to ±60mg or $\pm 0.59 m/s^2$. With both Inter-Integrated Circuit ($I^2C$) or Serial Peripheral Interface (SPI) protocols of communication, changes to the LSM303 configurations can be made on the fly. For this study, only the acceleration readings on the LSM303 are used, and the readings are made in $m/s^2$. The board does not require any further calibration, since the manufacturer ensure the calibration via values that are stored in the non-volatile memory [66].

Figure 4.7 show the accelerometer node in a close up. The red and yellow circles are for the LSM303 board and ESP8266 microcontroller, respectively. For the deployment version, only the black power cable is needed. The jumpers present in the figure are for code upload and debug only.

The module was installed directly into one of Zanni's supporting beams, gathering residual vibration data when the machine is working. Figure 4.8 shows the physicial installation of the module. The node contains multiple fail-safe precautions in case of the Wi-Fi signal goes down during a measuring period. Upon an established connection, the node transmits data regarding the acceleration on all three axis.

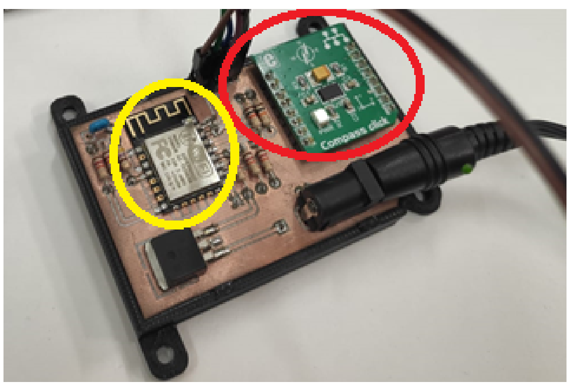

Figure 4.7: Accelerometer node.

Figure 4.8: Acceleration node installed on site.

# Chapter 5

# Implementation of the monitoring system

In this chapter the implementation of the architecture modules is detailed in their respective sections. The changes to the original IT infrastructure and the production parameters are also shown.

## 5.1 Proposed IT infrastructure

Since many of the problems faced early in this study were regarding the Wi-Fi quality and range in the shop floor and the data storage, the original approach consisted of a RaspberryPi, serving as the server and a Wi-Fi repeater inside the same metrology room, as close to the monitored machine as possible. Although this approach yielded the first dataset for testing different algorithms for prediciton, the layout proved inefficient as time went by.

Since Catraport is located in the industrial area of Mós, a twenty-minute drive from the IPB, a Secure Shell (SSH) or Virtual Network Computing (VNC) connection must be implemented to satisfy one of the principles of the IoT and ensure a remote connection. Otherwise, a daily visit to Catraport is required to access the data. In this layout, the RaspberryPi was installed directly on the server rack.

Due to fluctuations in the Wi-Fi network, sometimes the IoTaWatt device, or even the RaspberryPi would lose connection for more than a day. Problems with the RaspberryPi also emerged during the period, where the data was corrupted and all the installations were lost. This proved that a new approach should be thought of, taking into account all the drawbacks encountered during this one.

In the implemented infrastructure approach, the Wi-Fi repeater was moved from the metrology room directly to the shop floor and the RaspberryPi was replaced by a VM hosted on IPB. Upon moving the Wi-Fi repeater outside, a better signal strength and stability was seen on the IoTaWatt device. In this layout, the accelerometer node was also installed. In Figure 5.1 is exemplified this layout, where the accelerometer node is installed on the Zanni's frame.

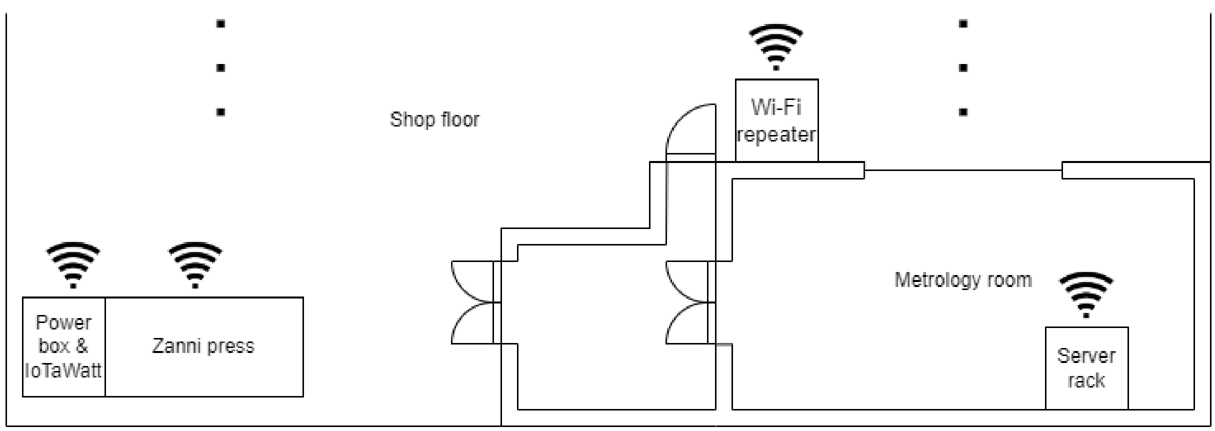


Figure 5.1: Adopted approach to the problem, model not to scale.

## 5.2 Data collection and storage

In this section, the data collection protocol used in the work, with a brief description of the upload frequency and collected parameters is described. The chosen database engine and collected production parameters regarding the stops are also presented.

### 5.2.1 Data collection

For the data collection, the sensing equipments described in Section 4.2 used the InfluxDB Application Programming Interface (API), via a HTTP POST method to inject data in the database. Inside the Post method, a Structured Query Language (SQL) statement is created to insert the data into the data storage.

The IoTaWatt device have a native function that pushes the data into the database every 5 seconds. Each measured electrical quantity is uploaded with a common timestamp, unifying the database. As for the accelerometer node, it uses an ESP8266 library that connects to the database and publishes data every 0.2 seconds via a POST method, while a Wi-Fi connection is present. Table 5.1 summarizes all the collected data and their upload frequency.

| Equipment | Measurement | Upload frequency |
|---|---|---|
| IoTaWatt | Current intensity<br>Active power<br>Reactive power<br>Apparent power<br>Power factor<br>Reference tension<br>Reference frequency | 5 seconds |
| Accelerometer node | X-axis acceleration<br>Y-axis acceleration<br>Z-axis acceleration | 0.2 second |

Table 5.1: Collected data from the sensing equipment.

Other collected data is regarding the production parameters. During the period of 30th of november to 10th of december of 2021, three service orders where collected. These orders are produced at the end of a work day, or on-demand via an application that generates an excel table for each order, containing details on the production time, working time, offline time and stops. All of these values are timestamped from beginning to end and labeled with a custom identification number, followed by an identification phrase. Figure 5.2 shows a pie chart with the stops occurred during one service order, in the order of the most frequent to the least.

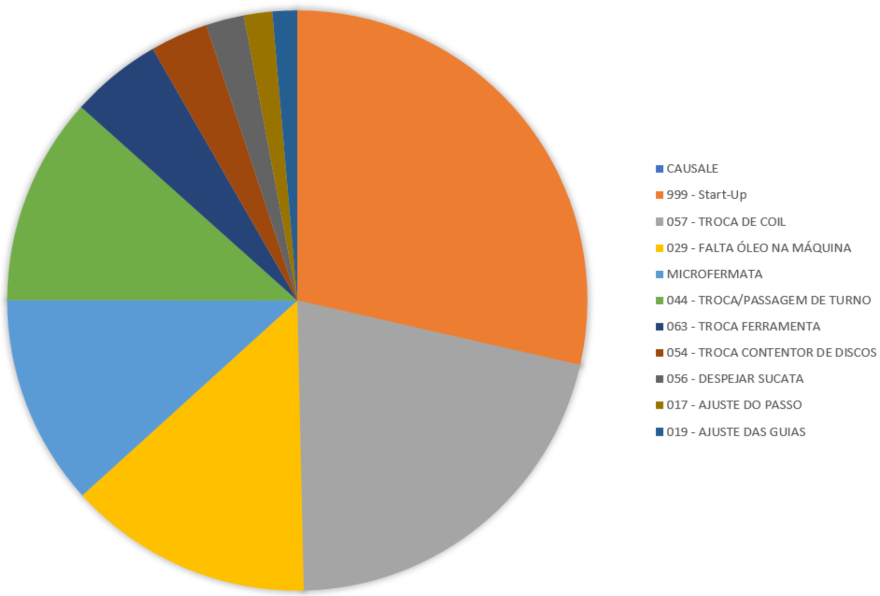


Figure 5.2: Stops presented in one service order.

Table 5.2 briefly describes the unique reasons that led to a machine stop, gathered for all three service orders. From the table, most of the stops are common stops on a daily work-cicle, however, the lack of oil in the machine is the only stop not caused by a human, being qualified as a stop due to machine conditions.

### 5.2.2 Data storage

Observing the evolution of a monitored parameter throughout the time is essential for condition-based monitoring. This narrows down the number of database engines available, where opting for a time series driven engine is beneficial.

| Stop | Description | Count |
|---|---|---|
| Microfermata | Micro stops with no more than 3 minutes | 188 |
| Start-Up | Stops that occur during the setting up of the machine | 75 |
| Coil change | Change of the metal coil to feed the machine | 21 |
| Shift change | Stop to change the machine's operator | 17 |
| Lack of oil in the machine | No oil pressure in the chamber that transmits the motor rotation to the downward force of the pressing die | 12 |
| Lunch stop | A longer stop, typically one hour, for lunch | 8 |
| Coffee break | Quick stop for a coffee break | 6 |
| Dump scrap | Dumping excess scrap | 5 |
| Piece container exchange | While producing pieces, the container must be changed when full | 5 |
| Waiting for tool maintenance | Idle time until the maintenance in the tool is made | 5 |
| Scrap exit cleanup | Removal of scrap pieces that may have become stuck in the outlet | 3 |
| Tool change | Change the pressing die | 3 |
| Machine cleanup | Overall cleanup of the machine, removing excess oil drops and scraps | 2 |
| Temp. stop by the quality dept. | Quick stop for quality inspection | 2 |
| Disk container exchange | When producing disks, the container must be changed when full | 2 |
| Medical visit | Operator had to leave early for a medical visit | 1 |
| Guides adjustment | Adjustment to the guides that keep the metal sheet straight | 1 |
| Step adjust | Stop to make adjustments to the amount of sheet is fed to the machine at each step | 1 |
| Control of the part in the loader | Quality control related stop | 1 |

Table 5.2: Unique machine stops during the monitored period.

A time series data base is build specifically for handling metrics, events or measurements that are time-stamped. The main difference between a time series data and regular data is that questions asked about it are made over time [67]. InfluxDB, or Influx, is a high-performance time series engine, with a powerful API [68] that enables for a plethora of devices to communicate seamlessly with the database.

Given its robustness, ease of use, and integration IoTaWatt and Grafana, the InfluxDB is the chosen database for this study, allowing for a more compact IT infrastructure.

Being open source and with API for retrieving and writing new data into the data, InfluxDB can easily be accessed via a Python script for a more intricate analysis of the collected data, as well as a cleanup of undesired data.

Since Influx is a time series driven engine, the data structure differs a traditional database. Each database on influx consists of a series of elements, such as:

- Measurement;
- Timestamp;
- Field keys and values;
- Tag keys and values;

Measurements can be thought of as tables, that contains the timestamp, field keys and values, and optionally the tag keys and values. In a single database, many measurements can coexist.

The timestamp, or time, is the association of the data and the time. If a value is written into the Influx database without a timestamp, the engine automatically assign the timestamp as being the exact time the value was uploaded. As for field keys and values, they are the metadata and raw data, respectively. The field keys are the name of each field, or the column headers in a table, while the values are the column values. The same comparison can be regarding tag keys and values, but these are both metadata, being even optional in the construction of a data structure. However, tags are more a performance efficient way to explore the data.

## 5.3 Data visualization

For a more robust way of visualizing data, the Grafana data visualization tool was selected. Since it allows for a better understanding of the gathered data by providing a variety of graphs and dashboards to be created via the web server that comes with Grafana's installation.

The use of Grafana as the visualization mechanism for this study is due to the integration presented with InfluxDB, as well as having some perks such as alerts, that can be set given some thresholds designed by the user. This came in hand when developing the Nelson Rules alerts, explained further in section 5.4.

The InfluxDB data is then queried by Grafana, accessing the InfluxDB API, via a HTTP GET method. As stated previously, Grafana is a powerfull tool for creating dashboards for data visualization, with many math formulas and functions to restructure the data to the desired need.

For a more concise data visualization, an overview dashboard is created, with time series graphs for important measurements, followed by Key Performance Indicators (KPIs) and alerts following the Nelson Rules.

Figure 5.3 shows the implemented dashboard. The overview dashboard is divided in three vertical sections, the first section contains the all the time series graphs for the current intensity, power and power factor of the three phases. On the middle section, the important KPIs regarding each measurement is displayed, such as the maximum, mean, and minimum values gathered for the last 5 minutes. The last section contains a panel that overviews all the alerts triggered by the Nelson Rules, along with the acceleration data on the X, Y, and Z axis. Each time series graph contains a legend on the bottom left to make the user better acquainted with the meaning of each color.

On the top left, a time series graph illustrates the evolution of the electrical power for the three phases over the time. To the right of this graph, a moving window of the last 5 minutes sums the power from all three phases and retrieves the minimum, average, and maximum values.

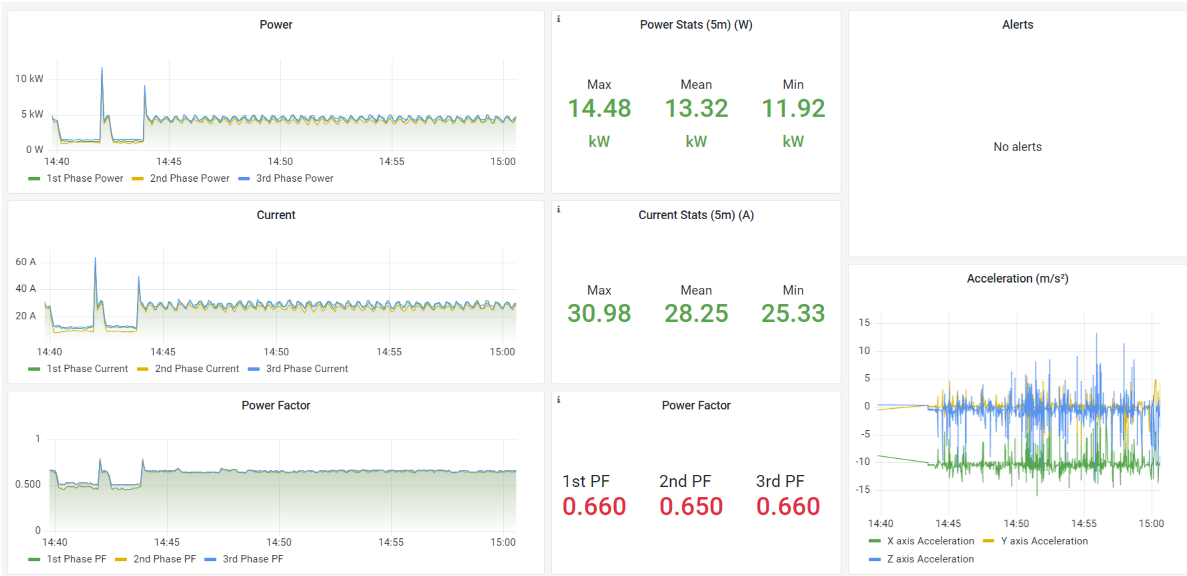


Figure 5.3: Overview dashboard.

On the middle row, the same idea is used to showcase the current intensity, with the development of the electric current, analyzed for the three phases, over time. On the right, there are statistics about the mean of the electric current on the three phases. These statistics are analyzed under the last 5 minutes of data present in the database.

The bottom left row, the time series graph of the power factor is displayed. On the right of this graph, a statistics graph shows the last non-null value held by the database. As of 2010, Portugal actively penalizes industries with a power factor lower than 0.96 [69] for the whole factory. To allow the user to see at a glance whether the power factor is within the range of values that are not subject to additional taxation, the numbers on the display take on different colors depending on the value measured. The rule for the threshold values follows the following distribution:

- Value $< 0.75$ is colored red;
- $0.75 \leq$ Value $< 0.92$ carries an orange color;
- $0.92 \leq$Value $< 0.96$ is marked as yellow;
- Value $\geq 0.96$ is colored green.

During the period of the study, Catraport kindly provided three electrical bills, and the three had a power factor that ranged from 0.90 to 0.93, this shows that the machine might be the culprit of degrading the overall energy efficiency of the factory.

For each variable visualized, a more in depth dashboard is created. Figure 5.4 spotlights the power dashboard. An alert panel presents all the triggered alerts according to Nelson Rules. The time series graph is a scaled version of the graph presented in Figure 5.3. Each measured phase have a special set of statistics graphs, with the last non-null value being displayed, followed closely by the maximum, mean, and minimal values measured in the last 5 minutes.

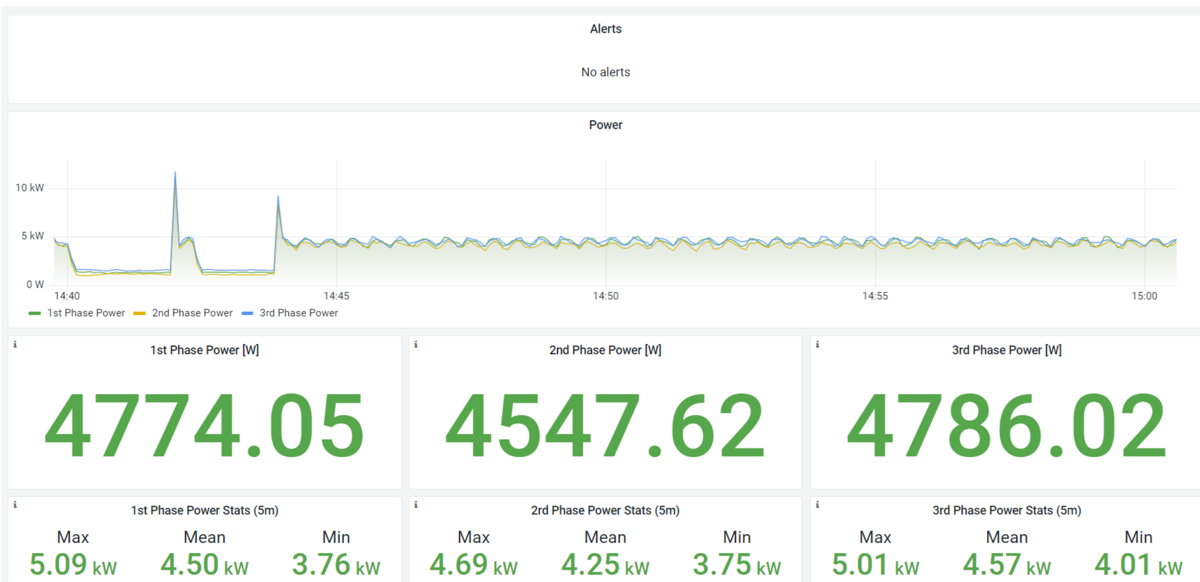


Figure 5.4: Power dashboard.

The current intensity dashboard is presented in Figure 5.5. Following the same principles as the power dashboard, with a wider time series graph, for more detailed analysis.

As for the power factor dashboard, since this is a key indicator on how effective the machine is regarding its energy consumption, a slightly different dashboard was created, presented in Figure 5.6. After the time series graph, gauge indicators shows the current value for each measured phase, with a range of 0 to 1. The gauge is color coded following the same distribution used for the power factor values in Figure 5.3. Below each gauge, a moving window analysing the past 5 minutes of data presents the maximum, mean, and


Figure 5.5: Current dashboard.

the minimum values from the moving window. This is an important KPI for the maintenance technician, to see how the machine is working and how effective is the machine, regarding energy consumption and usage.


Figure 5.6: Power factor dashboard.

A new dashboard has been implemented for acceleration data. The dashboard consists of three time series plots, one for each measured axis. These have threshold colors for the UCL and LCL, this makes it easier to read the data, as the operator can see which values are out of control. Given the position where the node was installed, the X-axis additionally measures the acceleration due to gravity.

Figure 5.7 illustrates the implemented acceleration dashboard. The dashboard is divided into two vertical sections, the first section contains time series graphs regarding the acceleration on each measured axis. The second section highlights the maximum, minimum, and mean values of the acceleration data for each axis, gathered from the last 5 minutes.


Figure 5.7: Acceleration dashboard.

## 5.4 Monitoring using Nelson Rules

Out of control samples of data can be seen in almost any databases. The so called outliers can actually tell much needed information about the analysed dataset. In this work, the out of control condition test used is a sub set of the Nelson Rules.

The Nelson Rules [70] are a series of 8 rules, being an update to the popular Western Electric Rules (WECO Rules), proposed in 1984. The update aims to equalize the probability of detecting an out-of-control sample among all rules.

Detection of the out of control samples takes into account the mean ($\bar{x}$) and the standard deviation ($\sigma$). The mean equation is presented in Equation (5.1), where $x_i$ is the i-th number from a set of $n$ numbers. Equation (5.2) shows the standard deviation, that uses the previously calculate value of the mean. The rules uses actual and previous values to detect outliers, shifts in the values, trends in the data, noise above a common noise, among other detections are possible with just these analysis.

$$\bar{x} = \frac{1}{n}\left(\sum_{i=1}^{n} x_i\right) \tag{5.1}$$

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}} \tag{5.2}$$

As more and more rules are implemented into a dataset, more frequently a rule will be triggered, sometimes with not a greater it is up to the implementer to choose a subset out of the eight rules. For this work, four rules were chosen, these being rule number one, three, four and six.

Rule number one aims to detect an outlier value, indicating that something may have gone wrong on that time period. For this, rule number one will trigger if a value surpasses the amount of three $\sigma$ above or below the mean. Figure 5.8 illustrates the activation of rule 1 on a fictitious set of values, where UCL and LCL means, respectively, Upper Control Limit and Lower Control Limit.

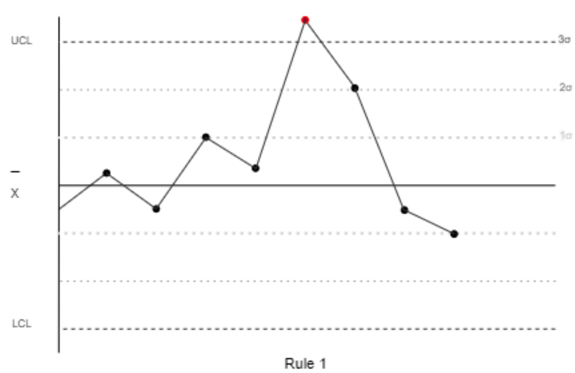


Figure 5.8: Nelson Rule number 1, adapted from [70].

Rule number three is presented in Figure 5.9. If six or more values are continually increasing, or decreasing, then a trend exists. This rule disregards the mean or the standard deviation. In a time series manner, an user can estimate that, if said trend continues, the measured value can be out of control in the near future.

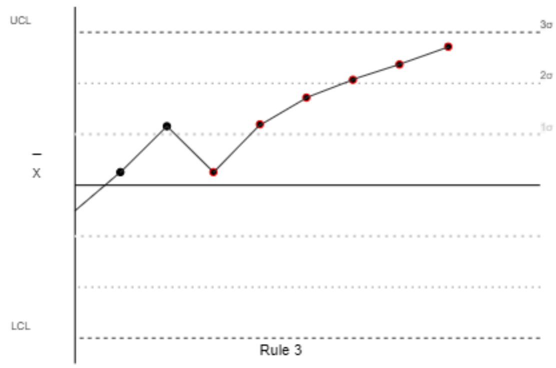


Figure 5.9: Nelson Rule number 3, adapted from [70].

Rule four, Figure 5.10, attends to the analysis of oscillation in a dataset, grouping the data in groups of 14 values. If all the values in the group alternate in direction, then the rule is triggered, as there is an oscillation beyond common noise on the analysed group.

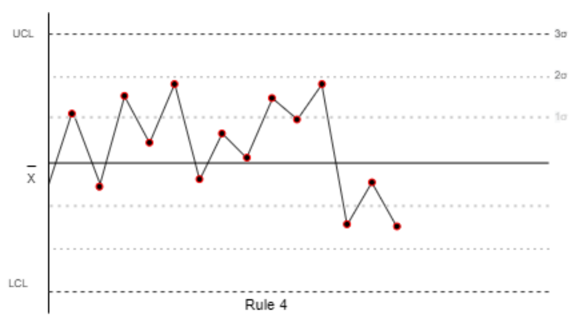


Figure 5.10: Nelson Rule number 4, adapted from [70].

The last of the selected rules, rule number six, states that if four or more out of five points deviate more than one standard deviation from the mean, the tested values have a slight tendency to get out of control. Figure 5.11 illustrates the activation of this rule.

Rule 1 is the standard out of control condition test rule. For this study, the addition of rule 3 is due to the fact that, on the prediction module of the proposed architecture,

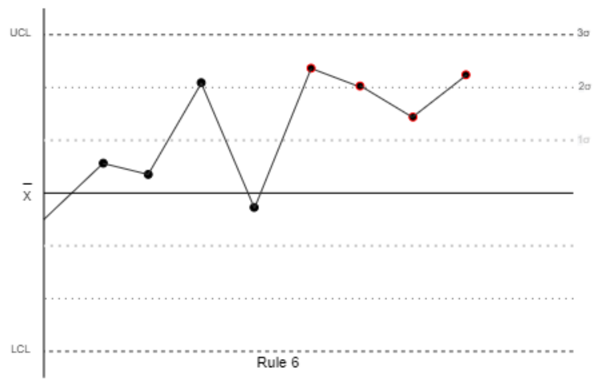


Figure 5.11: Nelson Rule number 6, adapted from [70].

detecting trends in data could prove beneficial. For example, if the machine is needing more power to do the same function it was doing minutes ago, this could mean that there is a lack of lubrication in the moving parts, generating an excessive force that may lead to a failure. The oscillation that rule 4 detects could have a negative impact on the manufactured pieces. Rule 6 is a complement to rule 1, supporting the idea of values being out of control, or heading towards an out of control zone.

Using the presented Nelson rules, the monitoring of the data was carried on. The rules are implemented via two scripts coded in Python, one for electrical data, and one for vibration data.

For the first one, every 70 seconds checks the electrical database and analysis the new values. This time window is chosen given the rule 4 needs of 14 values and the IoTaWatt device uploading data every 5 seconds. For the electrical values only the current intensity and power are being monitored, since these can highly represent the other gathered values.

After reading the data, the code then requests the mean and standard deviation from the database for each measured value. Table 5.3 summarizes values for mean, standard deviation, and the UCL for the Nelson rules. The LCL value is not relevant because the value of $\bar{x} - 3\sigma$ is always negative and there are no negative values for either current or power.

For the acceleration data, a Python script is deployed in the virtual machine as a service. Every 2.8 seconds the code gathers the last 14 values of acceleration in all three

| Collected data | | Mean | Standard deviation | UCL |
|---|---|---|---|---|
| | Phase | | | |
| Current [A] | 1 | 22.87 | 16.03 | 70.96 |
| | 2 | 19.82 | 15.73 | 67.01 |
| | 3 | 22.09 | 15.43 | 68.38 |
| Power [kW] | 1 | 3.60 | 2.91 | 12.33 |
| | 2 | 3.09 | 2.78 | 11.43 |
| | 3 | 3.48 | 2.76 | 11.76 |

Table 5.3: Mean, standard deviation, and UCL for the electrical data.

axis, To prevent false alarms in the acceleration data, since the device also measures acceleration values while the machine is not actively working, negatively influencing the mean and standard deviation, data gathered passes through a native percentile function on InfluxDB query. The filter aims to remove the stagnated values that revolves around -10, 0.1 and -0.5 meters/second$^2$ for the X, Y and Z-axis, respectively. Table 5.4 shows the filtered mean, standard deviation, UCL, and LCL for the acceleration data.

| Collected Data | | Mean | Standard deviation | UCL | LCL |
|---|---|---|---|---|---|
| | Axis | | | | |
| Acceleration [m/s$^2$] | X | -10.06 | 1.59 | -5.29 | -14.83 |
| | Y | 0.17 | 1.05 | 3.32 | -2.98 |
| | Z | -0.43 | 1.86 | 5.15 | -6.01 |

Table 5.4: Mean, standard deviation, UCL, and LCL for the acceleration data.

Once a rule is activated, the time where that disturbance occurred is stored in a specific measurement in the alarm database created in InfluxDB, along with a value indicating which rule was triggered, and an email is sent to the maintenance team. An example email is presented in Figure 5.12, translated to English. Upon clicking on the "Dashboard" hyperlink, the user is directed to the web page with the general dashboard view, present in Figure 5.3, within the time frame where the rule was activated. The original email is written in Portuguese.

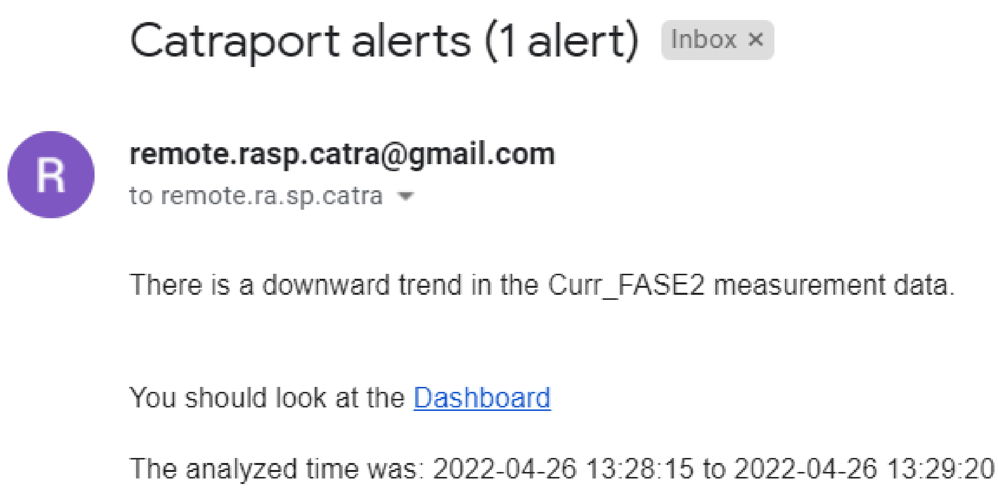


Figure 5.12: Example email.

Grafana uses an alert system that constantly checks the values in a time series graph. In this way, whenever a monitored values goes above a certain threshold, an alarm is raised on Grafana's web server. Knowing this feature, the data written to the database is tailored to be prone to an alert. Figure 5.13 displays the alerts regarding the first rule for the current intensity. The red line indicates the threshold for a value to be considered an alarm in Grafana.

It was seen that during a normal workday, spikes in current intensity appear when the machine starts, either at the beginning of a shift or after a short stop. This may be related to an increased need for torque for the motor to bring it out of inertia.

In Figure 5.14, constant trend detection usually reflects moments when the machine is idle or about to enter that mode. Trends while the machine is running are rare because current, power, reactive power, and other measured values remain relatively constant within the mean. No trends mean that the machine is running smoothly and does not

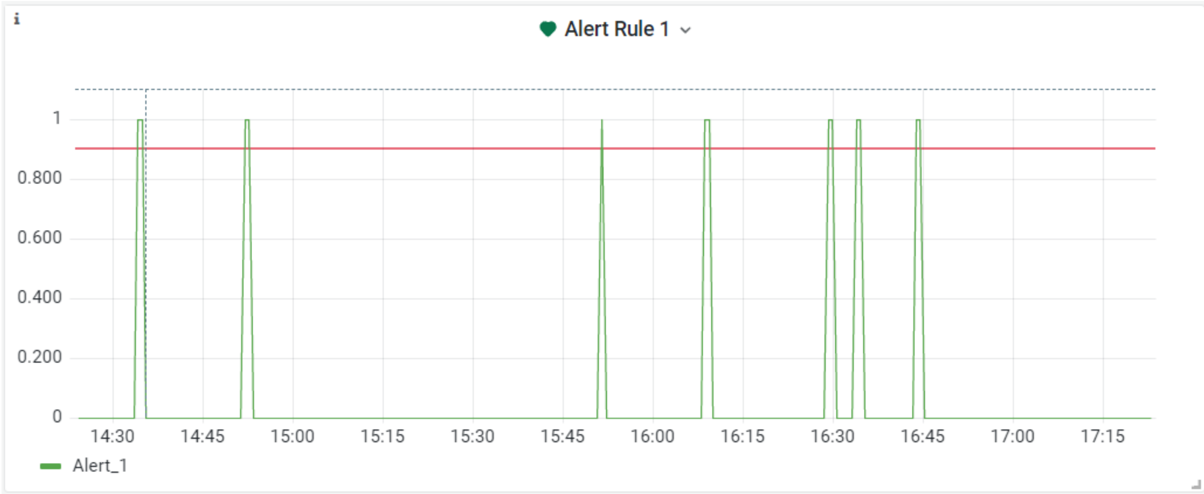


Figure 5.13: Rule 1 alerts.

require more power to reproduce the same pressing motion as before. This indicates that the operational parameters are very well adjusted.


Figure 5.14: Rule 3 alerts.

As for rule 4, Figure 5.15, no fluctuation that can be categorized as an oscillation above the general noise is seen during the observed period. This indicates that all data fluctuations are limited to the normal working noise that can be expected on a factory floor.

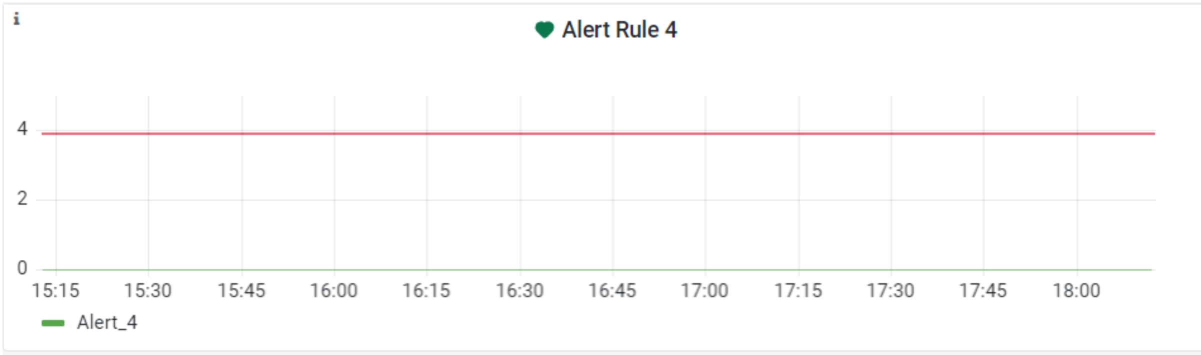


Figure 5.15: Rule 4 alerts.

Rule 6 relates to a measurement being slightly out of control and did not trigger during the observed period. As a complement to the analysis of the rule 3, given the stationarity of the data while the press is working as normal, no 4 points are above, or below $1\sigma$ for a prolonged period of time. An alert of rule 6 would be troublesome for the maintenance team, given that the machine would be demanding up to 80% more energy to perform the process, for at least 20 seconds. This makes of the rule 6 a valid indicator of the machine health.

## 5.5 Problems faced during implementation

The biggest problem faced during the implementation was ensuring a high quality Wi-Fi connection. Many times, during either the original approach or the adopted approach the Wi-Fi signal would go down. Although the IoTaWatt devices stores monitored data until a new Wi-Fi connection is made, there is so much the device can store before it's capacity is full.

Catraport shuts down the Zanni's power box at the end of the workday, since IoTaWatt's 5v input is installed in the first phase, the device also goes offline. If the timestamp of the values present in IoTaWatt is different from those present in the database, the device uploads the data until they are equal, but the upload is done every 5 seconds, being the minimum time between IoTaWatt data uploads. Given the data upload start time, October 08, 2021, when changing from the original approach to the current one, the system was without monitoring data in real time for 7 working days. Along with this problem, if the cold molding machine is down for maintenance, the IoTaWatt device also becomes unavailable for configurations or data upload.

In the adopted approach, the main problem persisted with the Wi-Fi signal. Not being dependent of a device installed in the shop floor to store the data, since the data is uploaded to the virtual machine, provided a huge bonus for working with the uploaded data while the machine is offline. However the acceleration node does not have storage capacity in the same way that the IoTaWatt does so when there is no Wi-Fi connection, there is no upload of acceleration data.

A more robust Wi-Fi network is one possible solution to avoid dead spots. By installing access points at key locations on the factory floor, Catraport has more ways to monitor the energy parameters of its other machines, giving it a better view of their monthly energy use.

# Chapter 6

# Prediction of failures and data forecasting

This chapter discusses an exploratory analysis of production parameters, using clustering techniques with electrical data, and prediction techniques using ML.

## 6.1 Exploratory analysis of production parameters

An initial idea came from the production parameters collected in section 5.2.1 to see if there was a correlation between the time before a malfunction, namely the lack of oil pressure alarm and the behavior of the electrical parameters.

For this purpose, the operational data collected during the period of one month was processed. The stops for lack of oil occurred 12 times, so the data were split and analyzed between each stop. Figure 6.1 showcases the current intensity of phase 1 for 2 of the 12 time frames. When the red line representing failure due to lack of oil goes to 1, an alarm occurs and the machine is stopped shortly after. From the figure, is possible to see that there is no pattern regarding an increase, decrease or peak in current. This same lack of pattern is seen on the active power and reactive power.

Clustering techniques using summary statistics such as mean, standard deviation,

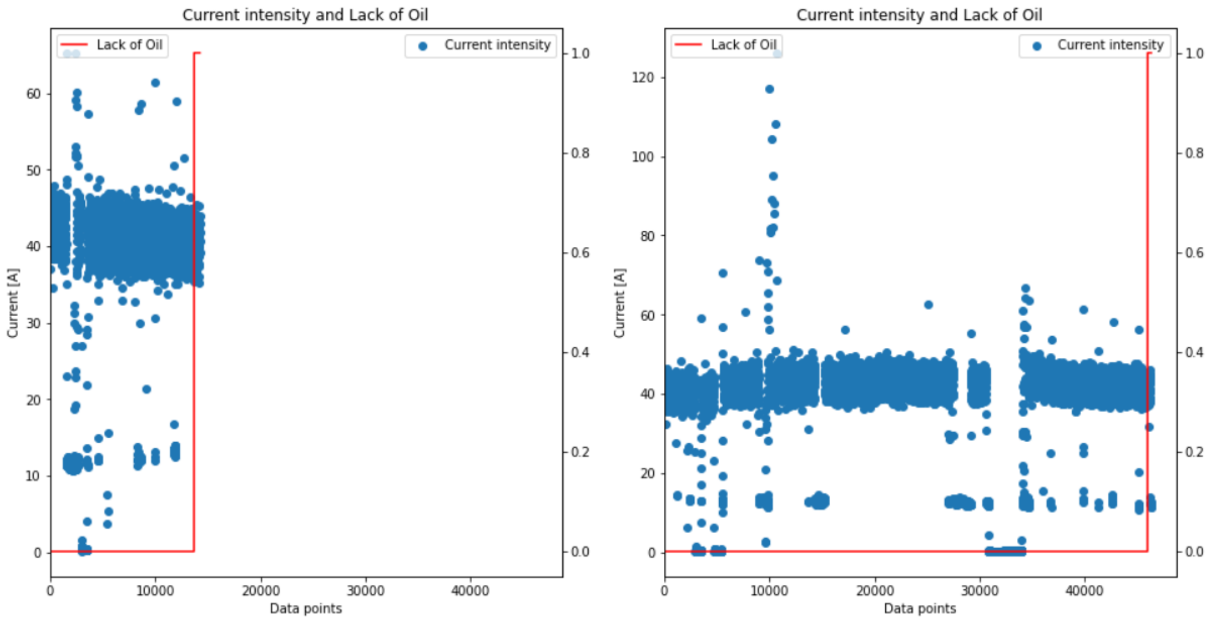


Figure 6.1: Current intensity before a stop.

skewness and kurtosis are used, these provides aggregated information of the time series [33], [71], aiming to check for any distinct cluster for the data, this way being able to actively determine when a failure may happen. As the mean and standard deviation equations were presented in Equations (5.1) and (5.2), Equations (6.1) and (6.2) represents the equations for skewness and kurtosis, respectively. They have been calculated using a 30 sample (2.5 minutes) sliding window over the current intensity of phase 1. Three series were discarded for being too short for the sliding window.

$$Skewness = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^3}{(n-1)\sigma^3} \tag{6.1}$$

$$Kurtosis = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^4}{(n-1)\sigma^4} \tag{6.2}$$

Summary statistics or features, are compared by plotting them against each other to see if a cluster can be formed from the approach. Figure 6.2 illustrates the clustering technique for a case of failure, being plotted against each other element-wise. The data was divided into 3 categories: Points over 60 minutes are colored in gray, points between

60 minutes and 10 minutes are yellow, and points below 10 minutes are colored red. Upon plotting the statistical features, it is desirable to find a potential cluster of values near-failure, however, no discernible cluster is found. In the other hand, this analysis unraveled a new view point of the data, where the analysis of the physical data should be used along to analyze possible correlations, using vibration data and additional features in time and frequency domains.

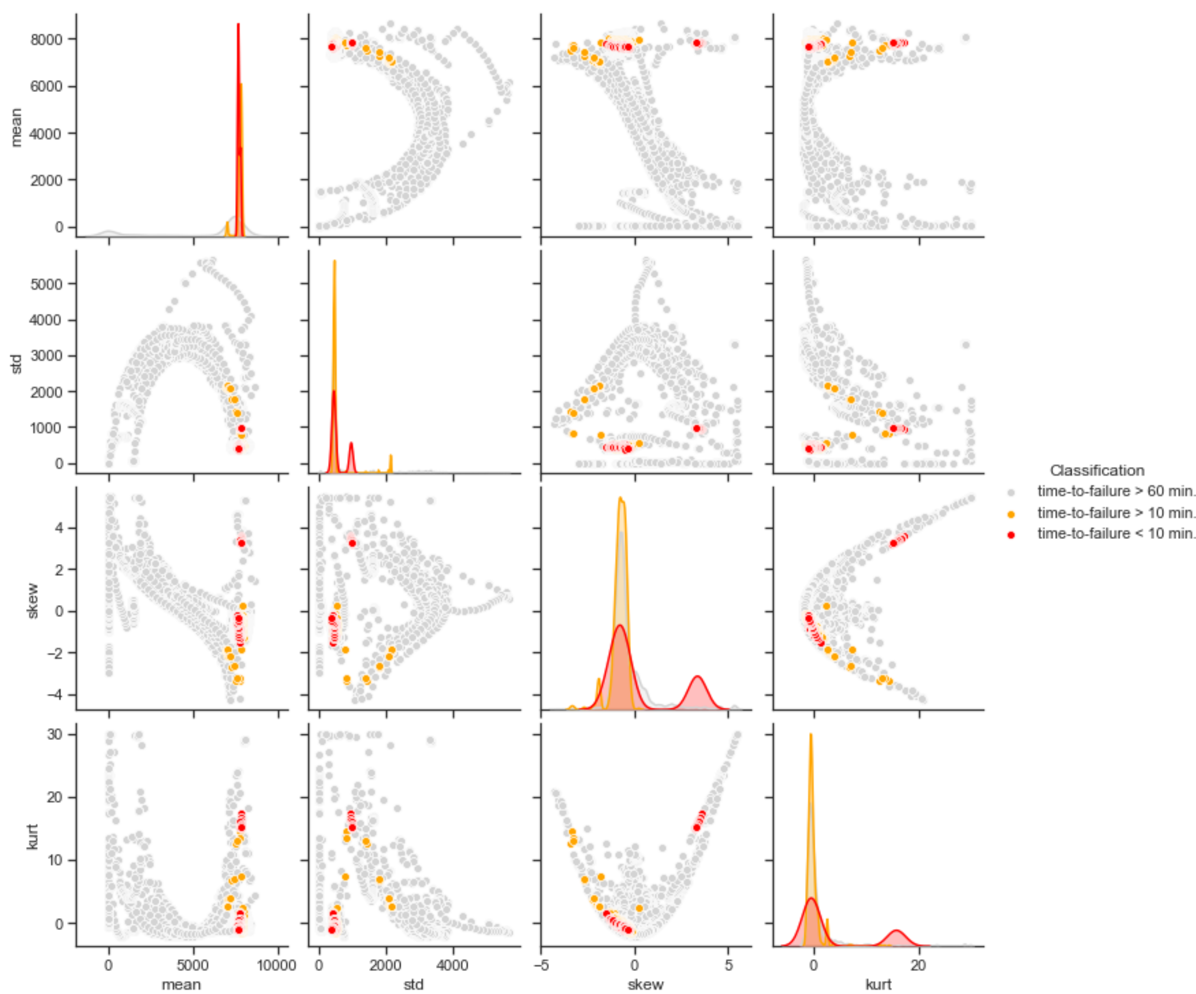


Figure 6.2: Relationship between features for one case of failure.

The high correlation of the electrical data is shown in Figure 6.3. Since the nature of the correlation between power and current is linear for a fairly stable input voltage. A correlation matrix have limits from -1 to 1. The closer to 1 a value is, represents that the two variables are more positively correlated, passing through 0 no correlation exists between the variables. Lastly, a value close to -1 in a correlation matrix suggests that the variables are negatively correlated, as one increases the other decreases.

| | Power_Phase1 | Power_Phase2 | Power_Phase3 | Curr_Phase1 | Curr_Phase2 | Curr_Phase3 | PF_Phase1 | PF_Phase2 | PF_Phase3 |
|---|---|---|---|---|---|---|---|---|---|
| Power_Phase1 | | | | | | | | | |
| Power_Phase2 | 0.98 | | | | | | | | |
| Power_Phase3 | 0.97 | 0.97 | | | | | | | |
| Curr_Phase1 | 1.00 | 0.97 | 0.96 | | | | | | |
| Curr_Phase2 | 0.98 | 1.00 | 0.97 | 0.98 | | | | | |
| Curr_Phase3 | 0.96 | 0.97 | 1.00 | 0.97 | 0.97 | | | | |
| PF_Phase1 | 0.88 | 0.87 | 0.88 | 0.90 | 0.88 | 0.90 | | | |
| PF_Phase2 | 0.76 | 0.80 | 0.76 | 0.72 | 0.78 | 0.72 | 0.61 | | |
| PF_Phase3 | 0.88 | 0.87 | 0.89 | 0.90 | 0.88 | 0.91 | 0.97 | 0.61 | |

Figure 6.3: Correlation matrix of the analysed data.

## 6.2 Current intensity forecasting using ML

From the correlation matrix and clustering technique, another approach arose: forecast the current intensity and monitor the forecast value using the Nelson Rules. For this approach, two different forecasting models where applied, the time series model consisting of an ARIMA algorithm, and two different LSTM architectures, a plain LSTM and a EECP-CBL architecture. These architectures were chosen for the forecast to see which one better adapts and learn from the dataset. For the tests, the database consists of 6100 points sampled every 20 seconds of machine work time over the span of a week. The data was split in 80% for training, and 20% for tests. The LSTM models were trained equally for 200 epochs.

### 6.2.1 ARIMA forecasting

The ARIMA algorithm consists of an autoregressive (AR), Integrated (I) and a moving average (MA) part, a method proposed by Box and Jenkins [72] in 1976. In the autoregressive part, new values of the model depends exclusively on the weighted linear combination of the previous values. The moving average model, as the names implies, uses the moving average and previous errors to determine new values. AR and MA equations are defined in Equations (6.3) and (6.4), respectively, where $\alpha$ is a constant, $\mu$ is the mean of the series, $p$ and $q$ are the orders of the AR and MA components, $\epsilon_t$ represents the white noise at time $t$, $X_t$ is the time series value at time $t$, $\beta_i$ and $\sigma_i$ are the weights.

$$X_t = \alpha + \sum_{i=1}^{p} \beta_i X_{t-i} + \epsilon_t \tag{6.3}$$

$$X_t = \mu + \sum_{i=1}^{p} \sigma_i \epsilon_{t-i} + \epsilon_t \tag{6.4}$$

The integrated part refers to a parameter $d$, that indicates how many times the time series are differentiated:

$$y_t' = y_t - \sum_{i=1}^{d} y_{t-i} \tag{6.5}$$

where $y_t$ is the value of the series at time $t$. The ARIMA model requires data to be stationary for the algorithm.

Through a trial-and-error analysis, the best ARIMA order for the dataset used in the forecasting is (1,1,1). The following Phyton implementation was used to train the algorithm.

```python
import pandas as pd
from statsmodels.tsa.arima.model import ARIMA
#Reading the dataset
dfn = pd.read_csv('influxTest.csv',index_col='time',skipinitialspace=True)
#Retrieving only the Phase 1 current values
data = dfn['Curr_FASE1'].values
#Getting the length of the dataset (43854 values)
data_len = len(data)
#Splitting into train and test data
data_train = data[:int(data_len*0.8)]
data_test = data[int(data_len*0.8):]
#Setting each value for the training dataset
history = [x for x in data_train]
predictions = list() #Create an empty list of predictions
# walk-forward validation
for t in range(len(data_test)):
```

```
model = ARIMA(history, order=(8,0,2))
#train the model
model_fit = model.fit()
#forecast the next value
output = model_fit.forecast()
yhat = output[0]
predictions.append(yhat)
obs = data_test[t]
history.append(obs)
```

The algorithm used a walk-forward validation [73], where for each value in the testing dataset, a new model was trained, the predicted value is stored and the test value is appended to the training dataset, until the last value in the testing dataset. Figure 6.4 presents the forecast value versus the real value, for a graphical comparison. Metrics were used to compare the model versus the other 2 forecasting methods, presented in Section 6.3.

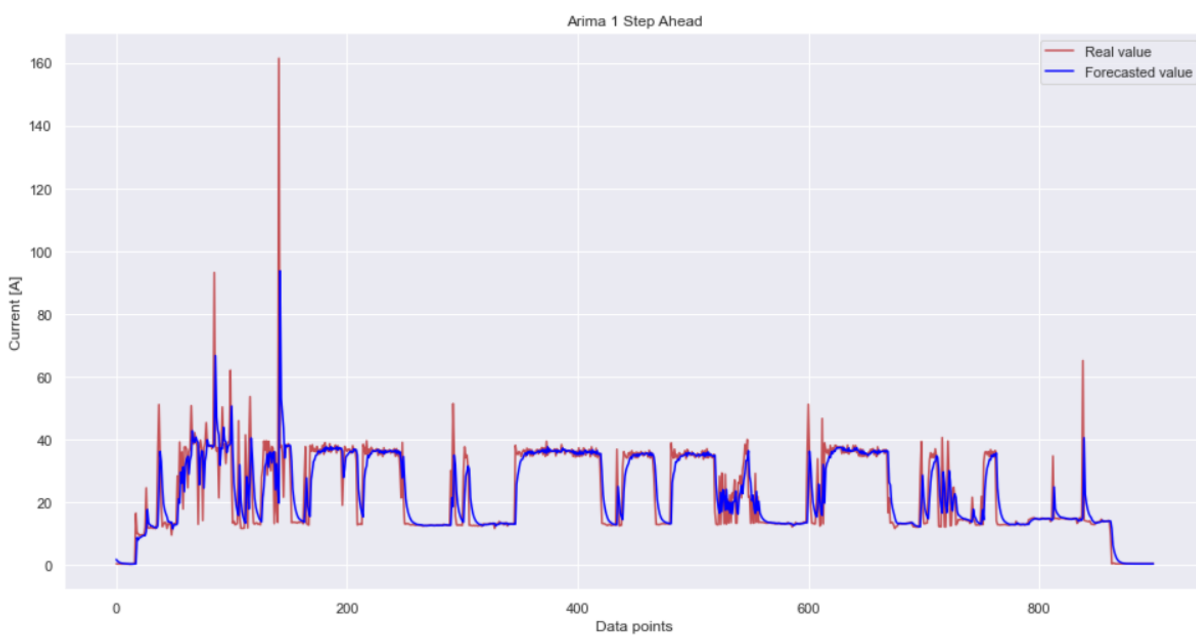


Figure 6.4: Real and forecast values for the ARIMA forecast, 1 step ahead.

### 6.2.2 LSTM forecasting

The Long Short-Term Memory (LSTM) was first introduced by Hochreiter and Shmidhuber [74] in 1997. Being a special form of Recurrent Neural Network (RNN) that, by introducing the gate controller, can prevent a problem of vanishing or exploding gradient that occurs in back-propagation processes [75]. The LSTM uses units that carries over information between one LSTM module to another, ensuring the ability of the network to learn long-term dependencies. The majority of RNNs uses epochs as a training hyperparameter. It defines the number of times that the algorithm will learn through the whole training dataset.

A standard LSTM model was developed consisting of two LSTM layers each with 70 and 20 units, respectively. The dropout layer prevents overfitting of the data and the output layer is a densely connected layer with one output, to predict one step ahead. The compiled model used as optimizer the Adaptative Moment Estimation (Adam) that accelerates the gradient descent by considering the exponentially weighted average of the gradients. As a metric to a RNN calculate the performance of the model, the loss function used is the mean square error. Figure 6.5 shows the forecast data for the LSTM method.

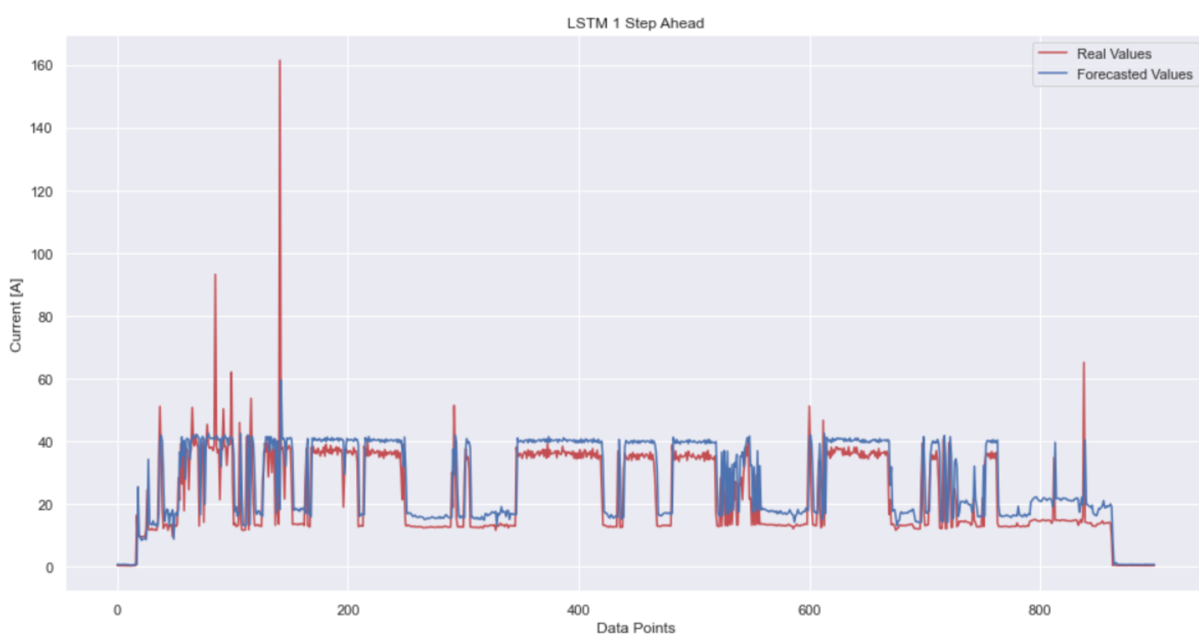


Figure 6.5: Real and forecast values for the LSTM forecast, 1 step ahead.

### 6.2.3 EECP-CBL forecasting

The Convolutional Neural Network (CNN) and Bi-directional Long Short-Term Memory network, called EECP-CBL was first proposed by Le, *et al.* [76] in 2019. The network input consists of two CNN layers to extract key features of the data. The values are then passed to two Bi-LSTM layers combines the forward and backward directions of data fed from the CNN layer. Finally, two densely connected layers produce the desired output. In this work, some modifications were made to the original model allowing for the forecast of more than 1 step ahead, without the need to feedback the predicted value to reevaluate the new prediction.

One advantage that EECP-CBL have is the ability to handle multiple inputs, such as the current intensity of other phases, power factor and power, improving the accuracy of the model. This multiple input EECP-CBL, called EECP-CBL (M.I.) is also presented in Section 6.3. Figure 6.6 shows the comparison between forecast and real values for 1 step ahead predictions using the EECP-CBL (M.I.) model.

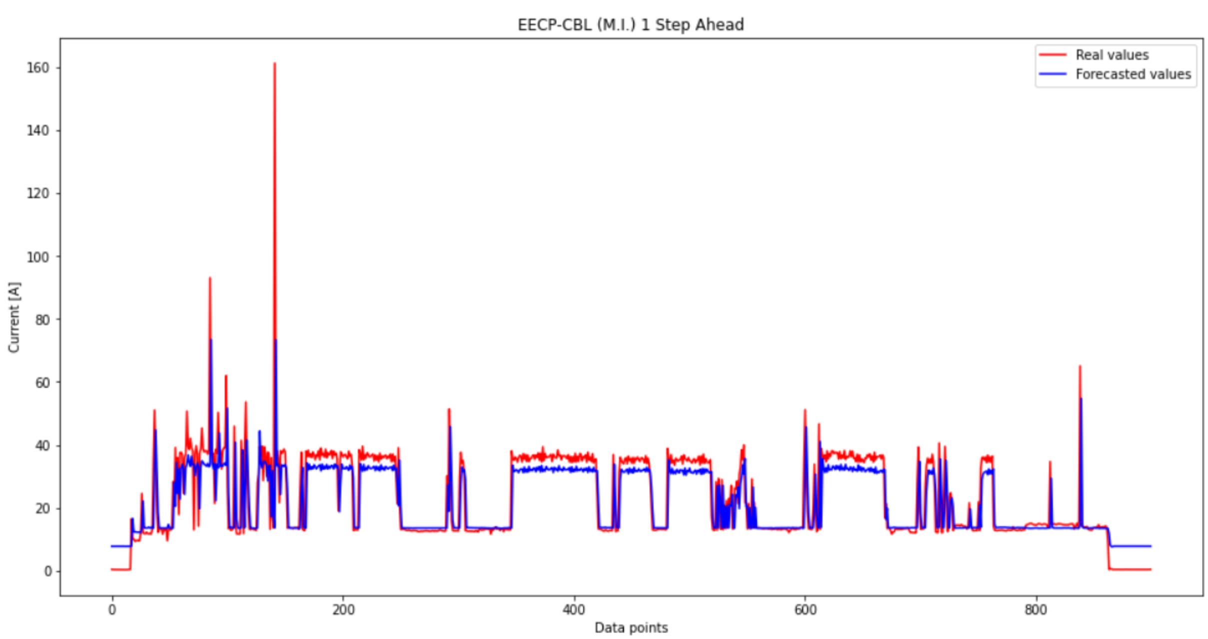


Figure 6.6: Real and forecast values for the EECP-CBL (M.I.) forecast, 1 step ahead.

## 6.3 Comparison of models

To chose the best model for data prediction, some metrics are introduced in this section, to measure how well the model performed against the training dataset. The metrics chosen were the Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE), with equations presented in Equations (6.6) to (6.8), respectively, where $\hat{y}_i$ is the prediction number $i$, $y$ is the real value.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{6.6}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{6.7}$$

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{6.8}$$

Table 6.1 presents the comparison of the metrics where EECP-CBL (M.I.) is the multiple input model. Each model was forecast 20 seconds (1 step), 3 minutes (9 steps) and 5 minutes (15 steps) ahead. The best model was the EECP-CBL (M.I.), with a lower overall metric than ARIMA for 5 minutes ahead. Of course that EECP-CBL is not a fair comparison with the other models, however it's important to show that an architecture can be more malleable as to accept other inputs to increase the overall accuracy, being capable of a greater generalization with little to none detriment in performance.

The idea behind the forecast steps are a way to show how the best model can outperform a standard time series driven algorithm when a forecast of a bigger time period is desired. It was seen that ARIMA converges to the mean value of the trained data after a short amount of steps, this proves that the model does not really learns the patterns as much as copy them. 3 minutes ahead shows the tipping point between ARIMA's accuracy and EECP-CBL's. Five minutes ahead is a good time ahead, based on the work presented in [77], since it gives the maintenance crew some key time to lookout for any deviations in the machine, or even request a full stop. A comparison between the real and forecast

| Model | forecast steps | RMSE [A] | MAE [A] | MAPE [%] |
|---|---|---|---|---|
| ARIMA | 1 | 8.81 | 3.98 | 26.92 |
| | 9 | 10.79 | 6.02 | 55.69 |
| | 15 | 11.38 | 6.75 | 69.67 |
| EECP-CBL | 1 | 10.37 | 7.06 | 44.59 |
| | 9 | 11.72 | 6.78 | 65.96 |
| | 15 | 9.64 | 5.5 | 50.89 |
| LSTM | 1 | 9.69 | 6.12 | 38.75 |
| | 9 | 9.53 | 5.63 | 38.08 |
| | 15 | 9.34 | 5.32 | 42.83 |
| EECP-CBL (M.I.) | 1 | 5.40 | 2.83 | 39.62 |
| | 9 | 11.30 | 8.48 | 26.64 |
| | 15 | 8.30 | 6.83 | 40.45 |

Table 6.1: Comparison metrics for EECP-CBL, LSTM and ARIMA models.

values in the real factory environment is highlighted in Figure 6.7, using the EECP-CBL multiple input model.

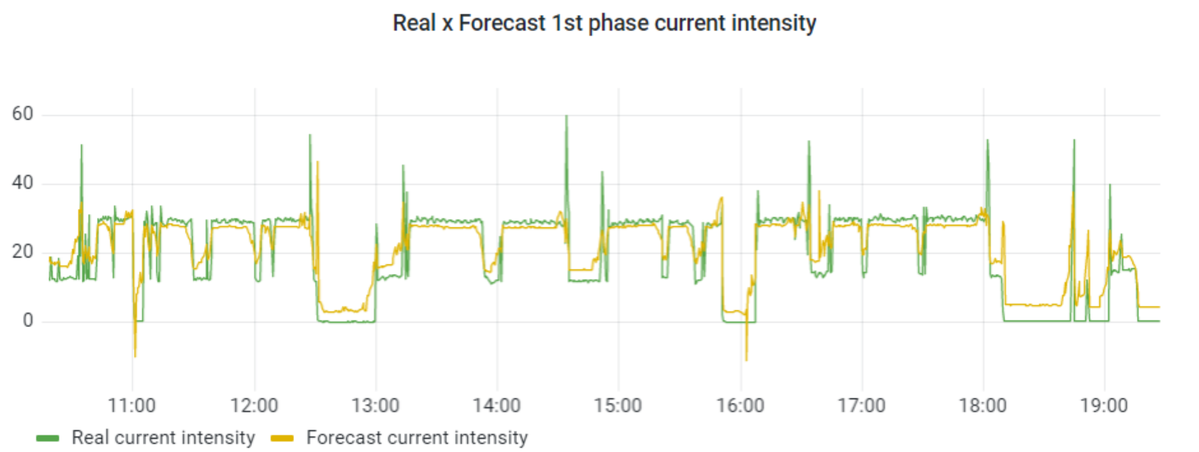


Figure 6.7: Real and forecast values using the selected model.

From the three different algorithms, some insights can be made. The ARIMA model is effective for short predictions, or successive predictions made using new values acquired from previous predictions, however, if the sliding window is not correctly configured, this algorithm may underperform. In the other hand, the LSTM based algorithms tend to learn from the trained data, being able to extrapolate and forecast different time ranges ahead, with a certain level of fidelity to the data's peculiarities.

## 6.4 Nelson Rules applied on the forecast

Using the Nelson rules presented in section 5.4, the forecast values are monitored with a service runs 24 hours a day, 7 days a week. If there is no new values in the database, i.e. the machine is offline, the service will not monitor or forecast new values, otherwise the trained model then forecast the next 5 minutes and these values are analysed.

Figure 6.8 exhibits the comparison between triggered alarms for the real values (yellow) and forecast values (green) in the time span presented in Figure 6.7. Although the forecast values are fewer and some out of sync with the real ones, these can reflect the period of high deviation of the data. These forecasts can give the maintenance team a head start over a failure that might happen.


Figure 6.8: Comparison between a forecast value alert and a real value alert.

# Chapter 7

# Conclusions and future works

This work addressed the development of a data acquisition and monitoring system for Catraport, which manufactures metal components for the automotive industry. The proposed work was to monitor electrical from one of their cold molding machine. The work has evolved to encompass the monitoring of physical parameters also.

The problem presented in this work is related to develop an IT infrastructure for data collection, using IoT technologies, monitor the gathered data via dashboards on webservers to give a more in-depth view of the monitored parameters, and implementation of machine learning and out of control sample monitoring algorithms for the detection of failures and malfunctions.

Initially, the implementation considered an on-site installation, however due to Wi-Fi problems and limitations of the RaspberryPI, a more disconnected approach was used. A Wi-Fi repeater was used as a temporary solution to address the Wi-Fi problem, yet a more robust Wi-Fi network through the shop floor is requested for a more reliable data collection. A Virtual Machine hosted on IPB to serve as the server for all the data fusion and monitoring, both via dashboards and algorithms.

The Grafana dashboard proved to be a reliable data monitoring tool, being fit for an industrial environment. Live refreshes of data and intricate created dashboards give the maintenance technician a deep understanding of the machine.

The custom physical data collection node worked as intended. Acceleration data was uploaded to the database as long as Wi-Fi signal was present. Although the IoT node was coded specifically to reconnect as soon as the Wi-Fi connection was lost, some time periods had no data collected due to lack of connectivity.

As for the data monitoring, the usage of Nelson Rules as an out of control monitoring algorithm proved reliable for the job. Rule selection must be selected with extreme caution and deep thought process, as more rules deployed tends to more false alarms.

Regarding the power factor, the monitoring showed that the machine was running far below the recommended average value of 0.96 out of 1. This implies in a higher electrical cost and a poor electrical efficiency of the machine. This was present also in the overall factory power factor, that from the three electrical bills had a maximum power factor of 0.93. After the monitoring system was deployed, an instruction manual has been created to guide the dashboard users and help them become more acquainted with the monitoring system.

As a means of comparison of the objectives presented in Section 1.1, the study regarding the existing IoT devices was carried out, as well as the DT based architecture and the installation of the sensing equipments, namely the IoTaWatt and the accelerometer node. The creation of the plataform for data visualization was made using the Grafana platform. Finally, the ML techniques were applied for data forecasting using real time data, since for failure prediction the data had no distinct pattern before a failure. The manual was the last addition made to complement the developed system.

For future works, the development of a more robust Wi-Fi network on the shop floor may prevent collected data of being lost, e.g. study of points to install access points to ensure whole shop floor coverage via the creation of a heatmap, to find weak spots. Works that aims to improve Zanni's electrical efficiency should be carried on, some research fields are: Calculation of a capacitor bank for in-line compensation or entire factory power factor compensation, via monitoring of other Machines.

When it comes to the current configuration, Catraport is dependent on the IPB for the data monitoring, since the server is hosted on an external cloud. Ideas to solve this problem may include using a computer with Linux as an S.O. to be the server to collect and monitor the data for the entire factory could be a solution, or the implementation of a more advanced Raspberry Pi-like single board computer such as the Asus Tinket Board 2s or the Nvidia Jetson Nano for the ML algorithms.

Moreover, the development of more accurate ML models, with the use of acceleration data, electrical data and operational data is one of the next steps in the implementation of this model, following the proposed system architecture. The proposed forecast method and Nelson Rules combo serves more as a warning to the maintenance crew that something might go out of control, but the time ahead that it predicts is short for the technician to have time to devise a failure contingency plan. With operational data such as Zanni's internal pressures and temperature, failure data labeling can be done. These measures may lead to a classification algorithm with a high accuracy and a wider time until failure.

# Bibliography


[1] M. T. Okano, "Iot and industry 4.0: The industrial new revolution," in *International Conference on Management and Information Systems*, vol. 25, 2017, p. 26.

[2] L. Allen, J. Atkinson, D. Jayasundara, J. Cordiner, and P. Z. Moghadam, "Data visualization for industry 4.0: A stepping-stone toward a digital future, bridging the gap between academia and industry," *Patterns*, vol. 2, no. 5, p. 100 266, 2021.

[3] Grafana. "Grafana: The open observability plataform." Acessed on april, 2022. (2022), [Online]. Available: `https://www.grafana.com`.

[4] Tableau. "Tableau - Business Intelligence and Analytics Software." Acessed on April, 2022. (2022), [Online]. Available: `https://www.tableau.com`.

[5] Infogram. "Infogram - Create Infographics, Reports and Maps." Acessed on April, 2022. (2022), [Online]. Available: `https://infogram.com/`.

[6] M. Manns, R. Wallis, and J. Deuse, "Automatic proposal of assembly work plans with a controlled natural language," *Procedia CIRP*, vol. 33, pp. 345–350, 2015, 9th CIRP Conference on Intelligent Computation in Manufacturing Engineering - CIRP ICME '14.

[7] J. P. Usuga Cadavid, S. Lamouri, B. Grabot, R. Pellerin, and A. Fortin, "Machine learning applied in production planning and control: A state-of-the-art in the era of industry 4.0," *Journal of Intelligent Manufacturing*, vol. 31, no. 6, pp. 1531–1558, 2020.

[8] S.-V. Buer, J. O. Strandhagen, and F. T. Chan, "The link between industry 4.0 and lean manufacturing: Mapping current research and establishing a research agenda," *International journal of production research*, vol. 56, no. 8, pp. 2924–2940, 2018.

[9] L. Bassi, "Industry 4.0: Hope, hype or revolution?" In *2017 IEEE 3rd International Forum on Research and Technologies for Society and Industry (RTSI)*, IEEE, 2017, pp. 1–6.

[10] H. Kagermann, W. Wahlster, and J. Helbig, "Recommendations for implementing the strategic initiative INDUSTRIE 4.0," *Final report of the Industrie 4.0 WG*, no. April, p. 82, 2013. DOI: 10.13140/RG.2.1.1205.8966.

[11] E. Y. Nakagawa, P. O. Antonino, F. Schnicke, R. Capilla, T. Kuhn, and P. Liggesmeyer, "Industry 4.0 reference architectures: State of the art and future trends," *Computers & Industrial Engineering*, vol. 156, p. 107 241, 2021.

[12] M. Khan, X. Wu, X. Xu, and W. Dou, "Big data challenges and opportunities in the hype of industry 4.0," in *2017 IEEE International Conference on Communications (ICC)*, IEEE, 2017, pp. 1–6.

[13] T. Zheng, M. Ardolino, A. Bacchetti, and M. Perona, "The applications of industry 4.0 technologies in manufacturing context: A systematic literature review," *International Journal of Production Research*, vol. 59, no. 6, pp. 1922–1954, 2021.

[14] Q. Wang, X. Zhu, Y. Ni, L. Gu, and H. Zhu, "Blockchain for the iot and industrial iot: A review," *Internet of Things*, vol. 10, p. 100 081, 2020, Special Issue of the Elsevier IoT Journal on Blockchain Applications in IoT Environments.

[15] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[16] A. Dorri, S. S. Kanhere, and R. Jurdak, "Blockchain in internet of things: Challenges and solutions," *arXiv preprint arXiv:1608.05187*, 2016.

[17] S. Chen, H. Xu, D. Liu, B. Hu, and H. Wang, “A vision of iot: Applications, challenges, and opportunities with china perspective,” *IEEE Internet of Things Journal*, vol. 1, no. 4, pp. 349–359, 2014.

[18] Statista. “Number of internet of things (iot) connected devices worldwide from 2019 to 2030.” Acessed on may, 2022. (2022), [Online]. Available: `https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/`.

[19] U. N. I. Council, “Disruptive Civil Technologies - Six Technologies With Potential Impacts on US Interests Out to 2025,” Acessed on may, 2022, 2008. [Online]. Available: `https://www.dni.gov/index.php/who-we-are/organizations/mission-integration/nic/nic-related-menus/nic-related-content/nic-publications`.

[20] P. Wegner. “Global iot market size grew 22% in 2021 — these 16 factors affect the growth trajectory to 2027.” Acessed on may, 2022. (2022), [Online]. Available: `https://iot-analytics.com/iot-market-size/`.

[21] J. Manyika, M. Chui, J. Bughin, R. Dobbs, P. Bisson, and A. Marrs, “Disruptive technologies: Advances that will transform life, business, and the global economy,” MCKinsey Global Institute, 2013.

[22] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of things: A survey on enabling technologies, protocols, and applications,” *IEEE Communications Surveys Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[23] P. Gokhale, O. Bhat, and S. Bhat, “Introduction to iot,” *International Advanced Research Journal in Science, Engineering and Technology*, vol. 5, no. 1, pp. 41–44, 2018.

[24] M. Grieves, “Origins of the digital twin concept,” Acessed on may, 2022, Aug. 2016. [Online]. Available: `https://www.researchgate.net/publication/307509727_Origins_of_the_Digital_Twin_Concept`.

[25] M. Grieves and J. Vickers, "Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems," in *Transdisciplinary perspectives on complex systems*, Springer, 2017, pp. 85–113.

[26] I. Errandonea, S. Beltrán, and S. Arrizabalaga, "Digital twin for maintenance: A literature review," *Computers in Industry*, vol. 123, p. 103 316, 2020.

[27] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, "Digital twin in manufacturing: A categorical literature review and classification," *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 1016–1022, 2018.

[28] F. Pires, A. Cachada, J. Barbosa, A. P. Moreira, and P. Leitão, "Digital twin in industry 4.0: Technologies, applications and challenges," in *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, vol. 1, 2019, pp. 721–726.

[29] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, "Digital twin in manufacturing: A categorical literature review and classification," *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 1016–1022, 2018, 16th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2018.

[30] P. Aivaliotis, K. Georgoulias, Z. Arkouli, and S. Makris, "Methodology for enabling digital twin using advanced physics-based modelling in predictive maintenance," *Procedia Cirp*, vol. 81, pp. 417–422, 2019.

[31] J. Lee, E. Lapira, B. Bagheri, and H.-a. Kao, "Recent advances and trends in predictive manufacturing systems in big data environment," *Manufacturing letters*, vol. 1, no. 1, pp. 38–41, 2013.

[32] D. M. D'Addona, A. Ullah, and D. Matarazzo, "Tool-wear prediction and pattern-recognition using artificial neural network and dna-based computing," *Journal of Intelligent Manufacturing*, vol. 28, no. 6, pp. 1285–1301, 2017.

[33] G. A. Susto, A. Schirru, S. Pampuri, S. McLoone, and A. Beghi, "Machine learning for predictive maintenance: A multiple classifier approach," *IEEE transactions on industrial informatics*, vol. 11, no. 3, pp. 812–820, 2014.

[34] W. Luo, T. Hu, Y. Ye, C. Zhang, and Y. Wei, "A hybrid predictive maintenance approach for cnc machine tool driven by digital twin," *Robotics and Computer-Integrated Manufacturing*, vol. 65, p. 101 974, 2020.

[35] N. K. Verma, R. K. Sevakula, S. Dixit, and A. Salour, "Intelligent condition based monitoring using acoustic signals for air compressors," *IEEE Transactions on Reliability*, vol. 65, no. 1, pp. 291–309, 2016.

[36] R. M. W. Horner, M. El-Haram, and A. Munns, "Building maintenance strategy: A new management approach," *Journal of quality in maintenance engineering*, 1997.

[37] B. A. Ellis and A. Byron, "Condition based maintenance," *The Jethro Project*, vol. 10, pp. 1–5, 2008.

[38] D. Raheja, J. Llinas, R. Nagi, and C. Romanowski, "Data fusion/data mining-based architecture for condition-based maintenance," *International Journal of Production Research*, vol. 44, no. 14, pp. 2869–2887, 2006.

[39] MIMOSA. "MIMOSA OSA-CBM." Acessed on may, 2022. (2010), [Online]. Available: `https://www.mimosa.org/mimosa-osa-cbm/`.

[40] M. Bengtsson, "Condition based maintenance system technology–where is development heading," *Condition Based Maintenance Systems–An Investigation of Technical Constituents and Organizational Aspects*, vol. 55, 2004.

[41] R. Ahmad and S. Kamaruddin, "An overview of time-based and condition-based maintenance in industrial application," *Computers & Industrial Engineering*, vol. 63, no. 1, pp. 135–149, 2012.

[42] D. Galar, A. Parida, A. Gustafson, and L. Berges, "Use of vibration monitoring and analysis as kpis in paper industry," in *Condition Monitoring and Diagnostics Engineering Management International Congress*, 2011, pp. 135–147.

[43] E. Abdelaziz, R. Saidur, and S. Mekhilef, "A review on energy saving strategies in industrial sector," *Renewable and Sustainable Energy Reviews*, vol. 15, no. 1, pp. 150–168, 2011.

[44] M. Paolanti, L. Romeo, A. Felicetti, A. Mancini, E. Frontoni, and J. Loncarski, "Machine learning approach for predictive maintenance in industry 4.0," in *2018 14th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, 2018, pp. 1–6.

[45] U. E. I. Administration. "International energy outlook 2021: Delivered energy tables." Acessed on may, 2022. (Oct. 2021), [Online]. Available: `https://www.eia.gov/outlooks/ieo/tables_side_pdf.php`.

[46] B. Sheldrick and S. Macgill, "Local energy conservation initiatives in the uk: Their nature and achievements," *Energy Policy*, vol. 16, no. 6, pp. 562–578, 1988.

[47] S. Darby *et al.*, "The effectiveness of feedback on energy consumption," *A Review for DEFRA of the Literature on Metering, Billing and direct Displays*, vol. 486, no. 2006, p. 26, 2006.

[48] P. Bandarra, M. T. Valdez, and A. Pereira, "Solutions for monitoring and analysing for energy consumption — energy management systems," in *2016 51st International Universities Power Engineering Conference (UPEC)*, 2016, pp. 1–5.

[49] M. D. Mudaliar and N. Sivakumar, "Iot based real time energy monitoring system using raspberry pi," *Internet of Things*, vol. 12, p. 100 292, 2020.

[50] K. Collins, M. Mallick, G. Volpe, and W. Morsi, "Smart energy monitoring and management system for industrial applications," in *2012 IEEE Electrical Power and Energy Conference*, 2012, pp. 92–97.

[51] K. Chooruang and K. Meekul, "Design of an iot energy monitoring system," in *2018 16th International Conference on ICT and Knowledge Engineering (ICT KE)*, 2018, pp. 1–4.

[52] L. Zhao, J.-l. Zhang, and R.-b. Liang, "Development of an energy monitoring system for large public buildings," *Energy and Buildings*, vol. 66, pp. 41–48, 2013.

[53] Y. Kabir, Y. M. Mohsin, and M. M. Khan, "Automated power factor correction and energy monitoring system," in *2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, 2017, pp. 1–5.

[54] M. Esposito, K. Bheemaiah, and T. Tse. "What is machine learning?" Acessed on may, 2022. (2017), [Online]. Available: `https://theconversation.com/what-is-machine-learning-76759`.

[55] P. P. Shinde and S. Shah, "A review of machine learning and deep learning applications," in *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, 2018, pp. 1–6.

[56] B. Mahesh, "Machine learning algorithms-a review," *International Journal of Science and Research (IJSR).[Internet]*, vol. 9, pp. 381–386, 2020.

[57] E. Funes, Y. Allouche, G. Beltrán, and A. Jiménez, "A review: Artificial neural networks as tool for control food industry process," *Journal of Sensor Technology*, vol. 5, 2015.

[58] S. K. Chinnamgari, *R Machine Learning Projects: Implement supervised, unsupervised, and reinforcement learning techniques using R 3.5*. Packt Publishing Ltd, 2019.

[59] A. Angelopoulos, E. T. Michailidis, N. Nomikos, *et al.*, "Tackling faults in the industry 4.0 era—a survey of machine-learning solutions and key aspects," *Sensors*, vol. 20, no. 1, p. 109, 2019.

[60] D. A. Narciso and F. Martins, "Application of machine learning tools for energy efficiency in industry: A review," *Energy Reports*, vol. 6, pp. 1181–1199, 2020.

[61] T. P. Carvalho, F. A. A. M. N. Soares, R. Vita, R. da P. Francisco, J. P. Basto, and S. G. S. Alcalá, "A systematic literature review of machine learning methods applied to predictive maintenance," *Computers & Industrial Engineering*, vol. 137, p. 106 024, 2019.

[62] C. Nichiforov, I. Stamatescu, I. Făgărăşan, and G. Stamatescu, "Energy consumption forecasting using arima and neural network models," in *2017 5th International Symposium on Electrical and Electronics Engineering (ISEEE)*, 2017, pp. 1–4.

[63] IoTaWatt. "IoTaWatt - Open WiFi Electricity Monitor." Accessed on April, 2022. (2022), [Online]. Available: `https://iotawatt.com/`.

[64] Espressif. "Esp8266 wi-fi mcu." Acessed on may, 2022. (2022), [Online]. Available: `https://www.espressif.com/en/products/socs/esp8266`.

[65] Pololu Robotics & Electronics. "LSM303DLHC 3D Compass and Accelerometer Carrier with Voltage Regulator." Acessed on may, 2022. (2022), [Online]. Available: `https://www.pololu.com/product/2124`.

[66] Pololu, *Ultra-compact high-performance ecompass module: 3d accelerometer and 3d magnetometer*, Nov. 2013. [Online]. Available: `https://www.pololu.com/file/0J564/lsm303dlhc.pdf`.

[67] S. N. Z. Naqvi, S. Yfantidou, and E. Zimányi, "Time series databases and influxdb," *Studienarbeit, Université Libre de Bruxelles*, vol. 12, 2017.

[68] InfluxDB. "Influxdb: Open source time series database." Acessed on april, 2022. (2022), [Online]. Available: `https://www.influxdata.com`.

[69] Entidade Reguladora dos Serviços Energéticos - ERSE, "Despacho n.º 7253/2010," 2010. [Online]. Available: `https://files.dre.pt/2s/2010/04/080000000/2194521949.pdf`.

[70] L. S. Nelson, "The Shewhart Control Chart-Tests for Special Causes," *Journal of Quality Technology*, vol. 16, no. 4, pp. 237–239, 1984.

[71] Y. Lu, R. Xie, and S. Y. Liang, "CEEMD-assisted bearing degradation assessment using tight clustering," *The International Journal of Advanced Manufacturing Technology*, vol. 104, no. 1, pp. 1259–1267, Sep. 2019.

[72] G. Box, G. Jenkins, and H. Day, *Time Series Analysis: Forecasting and Control*, ser. Holden-Day series in time series analysis and digital processing. Holden-Day, 1976.

[73] J. Brownlee. "How to create an arima model for time series forecasting in python." Acessed on may, 2022. (2017), [Online]. Available: `https://machinelearningmastery.com/arima-for-time-series-forecasting-with-python/`.

[74] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[75] N. Wei, C. Li, X. Peng, F. Zeng, and X. Lu, "Conventional models and artificial intelligence-based models for energy consumption forecasting: A review," *Journal of Petroleum Science and Engineering*, vol. 181, p. 106 187, 2019.

[76] T. Le, M. T. Vo, B. Vo, E. Hwang, S. Rho, and S. W. Baik, "Improving electric energy consumption prediction using cnn and bi-lstm," *Applied Sciences*, vol. 9, no. 20, 2019.

[77] A. Cachada, J. Barbosa, P. Leitão, *et al.*, "Maintenance 4.0: Intelligent and predictive maintenance system architecture," in *2018 IEEE 23rd international conference on emerging technologies and factory automation (ETFA)*, IEEE, vol. 1, 2018, pp. 139–146.