UNIVERSITÀ DI PARMA

UNIVERSITA' DEGLI STUDI DI PARMA

DOTTORATO DI RICERCA IN INGEGNERIA INDUSTRIALE

CICLO XXXVI

Digital Twin-Based Control System Enhanced by Machine Learning in food processing

Coordinatore:

Chiar.mo Prof. Gianni Royer Carfagni

Tutore:

Chiar.mo Prof. Ing. Giuseppe Vignali

Dottorando:

Ing. Giovanni Paolo Carlo Tancredi

Anni Accademici 2020/2021 – 2022/2023

Digital Twin-Based Control System Enhanced by Machine Learning in food processing

1 Introduction

The rapid advancements in Industry 4.0 are revolutionizing the food manufacturing sector (Mourtzis et al., 2022). One crucial technology in this transformation is the implementation of Digital Twin (DT) coupled with ML (ML). This thesis proposes an integrated approach using LabVIEW and Python to create a robust DT for process control in the food industry, enabling real-time monitoring, optimization, and predictive analysis. This work emphasizes the importance of data acquisition, modelling, control algorithms, and continuous learning to enhance efficiency, quality, and safety in food production.

1.1 Background and Motivation

The food industry, a fundamental pillar of global economies, is undergoing a transformative phase with the advent of Industry 4.0. This era of interconnected systems, digitalization, and automation has the potential to revolutionize food production, from farm to fork. Key enabling technologies (KETs), such as the Internet of Things (IoT), Artificial Intelligence, and Data Analytics, have given rise to innovative solutions for enhancing productivity, quality, and Safety in food manufacturing. However, this technological transformation comes with its own set of challenges. The food industry is characterized by complex processes, rigorous quality and safety regulations, and the need for precise control to ensure consistent product quality. Traditional manufacturing approaches struggle to cope with the increasing demand for customization, real-time monitoring, and predictive insights that Industry 4.0 promises. In this context the paradigm of DTs (DT), represent a powerful tool which bridges the gap between the physical and digital realms. A DT is a virtual representation of a physical system or process, allowing real-time monitoring, analysis, and optimization. By creating a DT of food manufacturing processes, industries can gain valuable insights into their operations, predict potential issues, and optimize resource utilization, leading to improved efficiency, reduced waste, and enhanced product quality. In this context, the integration of ML into DT works holds immense promise. ML techniques, which enable systems to learn from data and make informed decisions, can augment the capabilities of DTs. By harnessing the vast amount of data generated in food production processes, ML algorithms can provide predictive analytics, anomaly detection, and adaptive control, leading to more resilient and responsive manufacturing.

1.2 Scope of the work

The primary objective of this proposed work is to design, develop, and implement a comprehensive DT system for the food industry, leveraging LabVIEW, integrating ML in Python, and adopt data communication protocol. The work aims to address key challenges faced by the food industry in the era of Industry 4.0, with a focus on process control, optimization, especially for industrial plants not ready for the 4.0. The specific target of this work can be summarized in Table 1.

Table 1 Specific target of this work

1. Real-time Monitoring	•Enable real-time monitoring of food manufacturing processes through the creation of a virtual DT, to visualize and analyze process data.
2. Predictive Analysis	•Utilize ML algorithms to analyze historical and real-time process data, enabling predictive insights to anticipate potential issues, optimize resource allocation, and ensure consistent process reliability.
3. Control strategies	•Implement control strategies within the DT to respond to dynamic process conditions, optimizing parameters in real-time to improve efficiency.
4. Data Communication	•Integrate data from various sources within the food production environment, including sensors and actuators, to provide a holistic view of the process.
5. Remote Monitoring	•Provide real time information on the machine status and send alert to operators in case anomalies occurs
6.Interoperability	•Build up the paradigm of CPS and M2M ommunication and data exchange between M2M and M2H, though the adoption of adequate softwares and data communication protocols
7. Decision Support	•Provide decision support data to operators and decision-makers, allowing them to make informed choices in real-time.

By achieving these objectives, this works aims to allow the food industry to harness the benefits of DTs and ML, driving efficiency, and innovation in the ever-evolving landscape of Industry 4.0.

1.3 Structure

The scope of this work includes the development of a DT system for process control in the food industry, utilizing LabVIEW as the foundation for data acquisition and control, and integrating ML techniques in Python to enhance the system's capabilities. The focus is primarily on real-time monitoring, predictive analysis, and adaptive control of food manufacturing processes, with an emphasis on user safety and process efficiency.

The work will address the following key components:

i. Data Acquisition: The work will cover the integration of LabVIEW for real-time data acquisition from sensors, actuators, and other relevant sources within the food production environment. This component ensures the availability of accurate and timely data for analysis and control.

ii. DT Creation: The development of a DT will be a core aspect of the work. This virtual representation of the physical food manufacturing process will enable real-time monitoring and control, providing insights into the process dynamics.

iii. ML Integration: Python will be seamlessly integrated with LabVIEW to implement ML algorithms. This integration will enable predictive analysis, anomaly detection, and adaptive control based on historical and real-time process data.

iv. Decision Support: The work will provide actionable insights to operators and decision-makers, supporting informed choices during production.

v. Model Validation: Comprehensive validation of the work will be conducted through simulations and real-world experiments in food manufacturing settings, demonstrating its effectiveness in achieving the stated objectives.

vi. DT Framework for process control: Outline a DT framework suitable to adopt adequate control strategies into food manufacturing processes, for fluid a granulated food.

The work will be organized into the following main sections. Section 1 provide an overview of the background, motivation, objectives, and the significance of the work in the context of Industry 4.0 and the food industry. Section 2. DT shows the foundational concepts of DTs, their types, benefits, and challenges with a particular focus on food manufacturing sector. It will exploit the state of the art of the “bricks” to build up a Digital Twin as we intended it, which are:

I. Simulation, i.e., the system modelling though the physical equations and developer assumptions.

II. Process control strategies,

III. Machine learning algorithms, providing an overview of ML basics, its relevance in the food industry, and the potential benefits it brings.

IV. Data communication protocols.

Section 3 shows the material and methods adopted providing description of the pilot plant, software adopted LabVIEW as a Control Platform: This section will explore the key role of LabVIEW into both models developed in term of, in data acquisition, signals triggering, system modelling, process control, and data communication. Section 4 shows the description of the pilot plant investigated in whit sork, the digital model developed, work Validation and Performance: Comprehensive validation methods, experimental setups, performance metrics, and results obtained from simulations and test carried out during the laboratory experiments. It will also describe the ML algorithm coded with Python and their integration into the digital environment. Last section regards future Directions and Challenges: Discussion of results achieved, and challenges that arose in implementing the work. The structure outlined above ensures a comprehensive exploration of the DT work, from theoretical foundations to practical application, while validating its effectiveness in addressing the unique challenges faced by the food industry in the industry 4.0 scene.

2 Literature Analysis

2.1 Digital Twin in food industry

The concept of a DT has emerged as a powerful paradigm in the context of Industry 4.0, representing a virtual counterpart to a physical system or process (Semeraro et al., 2021). This digital representation, or "twin," is more than just a simulation; it is a living entity that mirrors the behaviour, attributes, and status of its physical counterpart in real-time (Singh et al., 2021). The DT concept holds great significance in various industries, with its potential to enhance efficiency, enable predictive insights, and optimize operations (Melesse et al., 2020). In the context of the food industry, DTs offer a transformative approach to process control, quality assurance, and resource management (Henrichs et al., 2021a). A DT is a digital replica of an actual product., process, or system whose purpose is to simulate, predict and optimize the behaviours of the physical counterpart (Davila Delgado & Oyedele, 2021; M. Liu et al., 2021). The first definition of this concept dates to the early 2000s, but already in the 60s the aerospace industry developed this technology using it during the Apollo 13 mission in 1970. Following the explosion of the oxygen tanks, the mission became a rescue operation, and the keystone was the ability to test multiple solutions at ground level through a DT of the spacecraft (Hazrathosseini & Moradi Afrapoli, 2023). Only the advent of Industry 4.0, however, has made it possible to develop this technology from the aerospace sector to the industrial context and the management of buildings so that, thanks to the development of the Internet of Things, Gartner, a strategic consulting company, has included the DT among the ten technological trends in 2017 (Perno et al., 2022). Dr. Michael Grieves, who is currently Chief Scientist for Advanced Manufacturing at the Florida Institute of Technology, first introduced the idea of the DT in 2002 while teaching a Product Lifecycle Management (PLM) course at the University of Michigan. He described the DT as the virtual, digital equivalent of a physical product. At the base of the model was the idea that each system was composed of a physical part present in real space and always existed, and a virtual counterpart containing the information of the previous one in virtual space. The connection between the real and virtual part (mirroring or twinning) took place through a continuous exchange of data and information (Grieves & Vickers, 2016). This concept was summarized by Grieves during the presentation of the course through the image below and containing all the characteristic elements of the DT: real space, virtual space, data flow from real to virtual space and flow of information from virtual to real and virtual subspaces. The conceptual idea for Product Lifecycle Management, emphasized how the systems remained connected throughout the entire life cycle, creating a dynamic model that could change over time through the four phases of creation, production (manufacturing), operation (support/support) and disposal. In literature there are several definitions of DT; one of these, taken from the whitepaper "DT: Mitigating Unpredictable, Undesirable Emergent Behaviour in Complex Systems" by Michael Grieves and John Vickers, Principal Technologist at NASA, defines the DT as "a set of virtual information constructs that completely describes a potential or actual manufactured physical product from the microtomic level to the level macro-geometric. At its optimal level, any information that could be obtained from the inspection of a manufactured physical product can be obtained from its DT." The concept is then divided into different types; DT Prototype (DTP), DT Instance (DTI), and DT Aggregate (DTA). The first contains all the sets of information necessary to make a physical product while the second describes a specific physical product to which a digital model remains connected throughout its life cycle. The aggregation of all DTIs constitutes the DTA which could be a computer construct capable of querying DTIs proactively. They operate in the DT Environment (DTE) i.e., the cloud. While over the years the introduction of the concept of digital model the quality and quantity of information related to virtual and real space have progressed rapidly. Grieves, in the Whitepaper "DT: Manufacturing Excellence Through Virtual Factory Replication", explains how focusing on the connection between the real and the virtual allows conceptualizing, comparing, and collaborating. Humans, unlike computers, do not process information sequentially, but conceptualize and contextualize the problem. This aspect, during the process of acquiring visual information, reducing to symbols and letters and visual reconceptualization, leads to a great loss of information and inefficiency over time (Melville et al., 2023). The use of the DT makes it possible to eliminate inefficient and counterproductive mental steps aimed at diminishing information and translating it from visual to symbolic information and back to visually conceptual information. A powerful intellectual tool is confrontation. However, it is inefficient because it involves analysing the physical and virtual product and identifying differences (Li et al., 2022). With the digital model, you can identify the ideal feature, the tolerance corridor, i.e., the positive or negative deviation allowed before a result is deemed unacceptable, and the actual trend line to determine whether the physical part is in line with the virtual one (VanDerHorn & Mahadevan, 2021).

Finally, another fundamental and characteristic aspect of human beings is collaboration. Thanks to the digital model and shared conceptualization, information about a product can be seen by an unlimited number of individuals without them having to share the same location (Lyytinen et al., 2016). The DT, therefore, allows the transition from the physical world, in which human beings operate inefficiently, to the virtual world, to put in place a common visualization and to identify the difference between what is and what should be by collaborating (Boje et al., 2020).

2.1.1 Structure and conceptual architecture of a DT

In the previous paragraphs, the DT has been defined as the virtual counterpart of an object or a physical process capable of optimizing business performance. The building blocks of the digital model can be identified as:

- a series of sensors distributed along the process able to capture operational and environmental data and actuators necessary to intervene directly on the physical process and optimize it
- data, i.e., aggregations of information detected by sensors from the physical world. These are part of the virtual world and can also contain design drawings, connections to external data feeds, and logs made by devices in the field.
- analysis techniques able to analyse data through routine simulations and visualizations and to predict changes and improvements to optimize the process

The conceptual architecture of the digital model, can be understood as a sequence of six steps able to create a closed-circuit connection between the physical and virtual parts. The six basic steps are summarized below.

- Creation: Introduction of sensors into the physical process that can perform measurements that are operational, performance-criteria, or external measurements that affect operations
- Communication: helps the two-way real-time connection between the physical and digital process
- Aggregation: the data are aggregated and inserted into an archive to prepare the next analysis
- Analysis: in this phase the data is analysed and visualized
- Deepening: once the analysis has been carried out, the in-depth phase allows to highlight the unacceptable differences between the physical part and the virtual counterpart and evaluate changes and improvements
- Action: the information collected can proceed in the opposite direction and be returned to the physical part with actuators which complete the interaction between the real and the virtual closing the cycle

The continuous contrast with developing countries, characterized by low labour costs, has pushed the most advanced countries to innovate the concept of factory toward Smart Manufacturing aiming to optimize, through digital, products and processes (B. Wang et al., 2021). The DT Shop-Floor is defined as the application of the digital model to production lines in the industrial field (H. Zhang et al., 2019). The traditional production process begins with the generation of a production plan based on orders and historical data and is followed by the actual production. At the end of this the products are inspected to verify that they are compliant for transport in the warehouse. All information generated during the process is kept in files for the next cycle. For this reason, the function of virtual space is limited and tends to overlap with the physical world by focusing on the collection, storage, and control of data, but ignoring simulation, optimization, and prediction information. What is missing is effective synchronization between virtual space and real space (Tao et al., 2022). The DT Shopfloor consists of four main components: the Physical Shop-Floor (PS), the Virtual Shop-Floor (VS), the Shop-Floor Service System (SSS) and the Shop-Floor DT Data (SDTD). The model sees at the center a database (SDTA) which receives and sends information to the other components present. The physical part (PS) and the virtual counterpart (VS) interact continuously, and the exchanged data is sent to the central database, which in turn exchanges the information with the Shop-Floor Service System. The latter houses all the company information systems for the control, management, and planning of production. After that, the information returns to the physical part in the form of commands Product Design (DTPD) is used to create a DT of a product and use the information obtained to support the product design process. Nowadays the success of a product depends more and more on the ability to manage the data received relating not only to the product, but also to the context in which it is used (Lee & Lee, 2015). The term Big Data refers to a very extensive collection of data in terms of volumes, speed, and variety to require specific technologies and analytical methods to be analyzed (Gandomi & Haider, 2015). Big Data analysis aims to extract useful information and process a multitude of disconnected data. This allows you to create a product based on the Big Data collected and their analysis (Mikalef et al., 2018). In this way, data-driven product design differs from traditional product design for several reasons; Design is no longer based on designers' experience in identifying relevant data, traditional methods are structured to process organized and clear data, and finally, they are unable to respond to changes in data of interest. The DTPD consists of three main parts: physical entities in real space, virtual entities in virtual space and connection through a continuous exchange of data in a bidirectional way. During design and production, virtual model parameters are transferred to the production line, while virtual models are processed into real physical products generating a closed loop.

The extension of the physical world, the world of atoms, to the virtual world, the world of bits, has led to the creation of DTs capable of simulating the behaviour of the physical counterpart. The usefulness of the DT is transversal to all manufacturing industries operating in various industrial sectors up to the creation of smart buildings and cities (Novák & Vyskočil, 2022). With the DT, businesses can work to eliminate unplanned downtime and reduce maintenance costs to improve productivity and efficiency (Errandonea et al., 2020).

Virtual space analytics offers a strategic opportunity to:

- anticipate and prevent problems
- carry out preventive maintenance activities based on real-time data provided by sensors
- resolve issues promptly to ensure that the physical part works as intended
- Improve physical twin performance by continuing testing in real-world situations and updating software in the product
- carry out durability tests by accelerating the passage of time to evaluate several years of operation in a few hours

A DT is a dynamic, virtual representation of a physical object, process, or system that is created and maintained through the continuous exchange of data between the physical entity and its digital counterpart. The DT captures real-time information, behavior, and interactions of the physical entity, providing a platform for monitoring, analysis, simulation, and control. It allows stakeholders to gain a deep understanding of the physical system's behavior, enabling real-time decision-making, optimization, and predictive insights (Jones et al., 2020).

Real-time Synchronization: The DT concept relies on real-time synchronization between the physical system and its virtual counterpart. Data from sensors, actuators, and other sources in the physical world are continuously fed into the DT, ensuring an up-to-date representation.

Two-Way Communication: The DT is not a static model but an interactive entity. Changes in the DT, such as simulations, optimization algorithms, or control actions, can influence the physical system. Likewise, data from the physical system can impact the DT, leading to a continuous feedback loop.

Simulation and Analysis: The DT enables simulations and analysis of the physical system's behavior under different conditions. This capability allows operators to test scenarios, predict outcomes, and identify potential issues before they occur in the real world.

Predictive Insights: By leveraging historical and real-time data, the DT can provide predictive insights. ML algorithms can analyze this data to identify patterns, detect anomalies, and forecast future behavior, helping operators make informed decisions.

Optimization and Control: The DT serves as a platform for optimization and control strategies. Algorithms within the DT can adjust parameters in real-time to improve efficiency, quality, and resource utilization.

Multi-Disciplinary Application: DTs are applicable across various industries, including manufacturing, energy, healthcare, and, in our case, the food industry. They offer a holistic approach to understanding complex systems and processes. In the context of the food industry, the DT concept holds immense potential for improving production processes, ensuring product quality, minimizing waste, and adapting to dynamic market demands. By creating a DT that integrates seamlessly with LabVIEW for data acquisition and control and harnesses the power of ML in Python, we aim to create a robust work that empowers the food industry in the era of I4.0. DTs come in various types, each tailored to specific applications and domains. Understanding these types is essential for selecting the appropriate approach when creating a DT work for the food industry. In our context, we will explore three primary types of DTs: product twins, process twins, with a focus on their relevance to the food manufacturing sector. A product twin focuses on creating a virtual representation of a specific physical product. It allows for detailed modelling, analysis, and simulation of the product's behavior, design, and performance throughout its lifecycle. In the food industry, a product twin could be used to optimize the design and manufacturing process of a specific food product, ensuring it meets quality standards, nutritional requirements, and consumer preferences. Recipe and Formulation Optimization: A product twin can simulate different ingredient combinations and processing techniques to optimize the taste, texture, and nutritional content of a food product. Packaging Design: The twin can assess packaging materials' effectiveness in preserving freshness and preventing contamination. Quality: Product twins can be used to predict the shelf life of perishable food items and identify factors affecting product quality. Process twins focus on replicating the behavior of a manufacturing or production process. They enable real-time monitoring, analysis, and control of the process, allowing operators to optimize parameters, detect anomalies, and ensure efficiency. In the food industry, a process twin could be applied to optimize food processing, packaging, and distribution processes. Real-time Monitoring: Process twins can monitor critical parameters (e.g., temperature, humidity, pressure) during food processing to ensure that the process adheres to safety and quality standards. Energy Efficiency: By analysing process data, process twins can identify energy-intensive stages and suggest optimizations to reduce energy consumption.

Predictive Maintenance: Process twins can detect signs of equipment failure, enabling proactive maintenance to prevent production disruptions.

As the food industry embraces the transformative wave of Industry 4.0, it encounters a multitude of benefits that can revolutionize processes, enhance efficiency, and improve product quality. However, this journey is not without its challenges, especially in an industry with stringent regulatory requirements, complex supply chains, and the need for maintaining consumer trust (Leng et al., 2022). Let us explore both the benefits and challenges that the food industry faces in the era of Industry 4.0. Those can be summarized as follow:

i. Improved process Control: Industry 4.0 technologies, such as DTs, enable real-time monitoring and analysis of production processes. This results in better quality control by detecting anomalies and ensuring that products meet stringent quality standards.
ii. Innovation: The integration of emerging technologies like IoT, AI, and ML encourages innovation in products.
iii. Workforce Training: Industry 4.0 requires a skilled workforce capable of operating, maintaining, and innovating with new technologies. Upskilling existing employees and attracting new talent are ongoing challenges.

The benefits of Industry 4.0, particularly when coupled with the DT work and ML, can significantly outweigh the challenges, leading to a more efficient, responsive, and robust food industry (Guruswamy et al., 2022). The introduction of Digital Twin (DT) models into the food processing industry represents a significant step forward in optimizing production, enhancing product quality, and ensuring efficient resource utilization (Verboven et al., 2020). As we delve deeper into the practical applications of DT within this sector, it becomes evident that this technology has the potential to revolutionize food manufacturing. While previous studies have explored the theoretical benefits and challenges of DT models in the food industry, few have ventured into the realm of real-world implementation. This thesis seeks to bridge that gap by presenting two compelling case studies that showcase the successful integration of DT models with control works in actual food processing plants. These case studies provide concrete evidence of the positive impact that DT technology can have on various aspects of food production. In addition to presenting these case studies, we offer a structured approach to assist food industry professionals in implementing DT-enabled control systems effectively. This approach draws from our experiences and observations from the case studies and aims to explain the complexities of incorporating DT models into food processing operations, aiming to promote a practical adoption of DT technology within the food industry. By providing real-world examples, guidance, and a work for implementation, we aim to allow food manufacturers to embrace DT models and leverage their potential for transformative change. As the industry continues to evolve, those who harness the power of digital twin technology will undoubtedly lead the way towards a more efficient, agile, and quality-driven future in food production. A mathematical model configured as a Single Input Single Output (SISO) to control the fluid level of a coupled tank process was developed by(Naha & Das, 2024). They have then applied both conventional and advanced control systems, including PI, fuzzy, neuro and neuro-fuzzy models, which were designed and simulated for the plant under examination using MATLAB/Simulink. As a result, compared to other controllers applied to this nonlinear system, the developed fuzzy controller tracked the setpoint faster (Messai et al., 2011) reported an autonomous, multi-farm, produce drying system, doubling as an oven using a Raspberry PI and an inexpensive PLC. In this case, the user can control both oven temperature and humidity from an easy-to-use web interface available on a mobile device, for example, or on the oven's human-machine interface (HMI). The results show that the system designed can successfully control both the temperature and humidity of the dehydrator. (M. Wang et al., 2023) analyses cooking, which is not a highly automated operation, especially at the household level. They have designed a prototypical control system that uses PLCs, computers, and electromechanical devices to assist in cooking two typical Indian foods, including pancakes and rice cakes. Instead, (Rostam et al., 2023) developed a closed-loop PID control system with a self-tuning function for temperature monitoring using SCADA-integrated PLCs. This is difficult to achieve with traditional control methods. Alternatively, the PLC is modelled using ladder logic (Alphonsus & Abdullah, 2016). As Industry 4.0 technology becomes more prevalent, data driven PID control systems are being superseded by more accurate types of control (H. Yu et al., 2020). Predictive models, in which process variables are analysed and optimized using predictive algorithms, have also been used in a model predictive controller (MPC) approach (Afram et al., 2017). For example, (Bagyaveereswaran et al., 2016) use MATLAB to design a MPC algorithm that compares its performance with standard PIDs and cascading PIDs. Distributed MPCs, which aim to consider the inherent modularity of the process, have also been proposed for plant-wide control (Fortela & Mikolajczyk, 2023) These systems were designed to control each module with a dedicated controller which uses locally available information and relevant knowledge gained through interaction with the other controllers(van Niekerk et al., 2023).

Recently, process control approaches with the general goal of optimizing manufacturing processes in real time (Stavropoulos et al., 2023) or optimizing quality control (Gao, 2023) have moved on to introducing DT models in several areas. DT models are effective not only for implementing model-based control, but also for creating a closed-loop control system because they allow real-time integration of sensor data, model predictions, and control algorithms. According to these issues, (C. Zhang et al., 2020) propose a comprehensive study for a knowledge-based DT manufacturing cell to support autonomous manufacturing through an intelligent control policy that incorporates simulation and prediction model features. (Karagiannopoulos et al., 2023) for instance, designed a DT control system with the goal of optimizing the manufacturing and remanufacturing processes typical of WEEE logistics. Several researchers have proposed an outline of the potential of DT for food process modeling in the food industry, the focus of this paper. For example, (Sarantinoudis et al., n.d.) examined the application of DT in the food processing industry, with a particular focus on the potential for using these models to optimize production planning. It also highlights key challenges, opportunities, and unique needs of food processing versus other process sectors.(Kannapinn et al., 2022) further suggested that DT systems had the potential to improve conventional control methods in heat-processing by simulating a process in real time, thus replacing information collected by sensors when this was not available or effective. In fact, their proposal was for a DT model for autonomous food processing. (Henrichs et al., 2021b) investigated the possible use of DT in the food industry. They have provided a classification of the available implementations as well as the challenges for the application of DT in this sector. However, there are few studies describing the use of DT models in the food industry. For fruit digital models, the only significant case is provided by (Defraeye et al., 2021), who model the thermal properties of mango fruit during cold chain transport. Regarding the modelling of food processes, most of the existing investigations have proposed mathematical or simulation models having the capability to be turned into a DT tool for process control, but the implementation part is often missing. For example, (Bianco et al., 2022) designed semiempirical mass and heat transfer models for dehydrating and cooling green vegetables. As a result of their study, explaining how the proposed model can be used to develop a DT framework for this process in future development. Similar considerations were made by (Gai et al., 2023); they proposed a mathematical model for the simulation of circulating fluidized bed gas-solid flow systems. Similarly, (Zewdie et al., 2022) developed a heat and mass transfer model to predict the distribution of moisture and temperature during the ripening process of onions, which, if not managed appropriately, can lead to weight loss during storage. Finally, they suggested the future use of their results in a DT system model of the process. An exception is (Maheshwari et al., 2022),who developed a DT model of a food processing plant producing ice cream. They found that DT increased the performance of the existing system in many ways, such as plant availability, engineering effectiveness, and worker effectiveness. However, the study focused on evaluating the ice cream company's KPIs using a DT model of the process, while the control was not embedded in the model. In a related example from the pharmaceutical sector, (W. Yu et al., 2022) reported on DT modelling of drying processes. They provided a framework that incorporates machine learning strategies and collects data from multiple in-line sensors in the equipment. This resulted in significant savings for the pharmaceutical company, as the control system automatically identified the optimal endpoint for the drying process. Therefore, the scientific literature focused on the implementation of DT models in food processing is limited. Furthermore, there are few examples of DT methods linked to process control systems. In this review, there is a consensus among the authors that the application of DT in the food sector is still at an early stage (Nychas et al., 2021). Notably, a limited number of applications have described how to retrieve data from physical to virtual objects (Jiang et al., 2021). Against this background, the work (Maheshwari et al., 2022) is the only study that has formalized an architectural work for the implementation of a DT model in a food context. Based on a case study of an ice cream factory, the authors proposed a set of basic steps for designing and implementing a DT model, focusing on management concerns and process production metrics. Although there are other papers in the literature (Kober et al., 2023), these deal with more general aspects related to the potential and acceptance of DT without addressing the concrete implementation.

However, due to the specific characteristics of the food industry (Tancredi et al., 2022a) and the presence of specialized facilities, a rigorous methodology for DT model implementation is required. In practice, DT models can also be difficult to implement independently, as they rely on accurate modelling of the real process. For effective control of the system, (Bottani et al., 2020)real-time communication capabilities should also be provided. The purpose of this paper is to present two case studies that illustrate the application of DT systems in the food industry. In addition, the DT models for the two food plants were integrated into a control system. Finally, a general approach for the implementation of DT-based control systems in the food industry was outlined based on the results of the case studies.

2.2 Food industry process control

Several factors including the final product, technological procedures used in the food companies can differ (Earle, 1997). However, there is a common necessity for designing food process control systems, which includes defining the sensors, actuators, controllers, and software (Morgan & Haley, 2019). Different forms of control systems, including feedback control, feedforward control, and model-based control, have been examined in the ground-breaking work by (McFarlane, 1995). The first type is known as "closed loop" control and denotes a kind of control that utilizes comprehension of the system's behaviours or output for changing and adapting the input signals to produce the desired results. The specific criteria and properties of the process being regulated determine the control system to be used. In fact, there are numerous applications of PID models and PLC utilized in the food sector for process control in the literature. A PID controller was designed for a PLC controlled pasteurisation system (Torres & Galvis, 2017). A Model Predictive Controller (MPC) for the process was created using MATLAB. A mathematical model for controlling the fluid level of a couple tank process has been developed using a Single Input Single Output configuration (Lakshmi et al., 2022). PI, fuzzy, neuro, and neuro-fuzzy models developed and tested for use in control systems were then used alongside more sophisticated control systems (Weldcherkos et al., 2021). The simulation results showed that the developed fuzzy controller can track setpoints more quickly than previous controllers used on this nonlinear system. (Oluwaleye et al., 2021) have concentrated on creating a self-contained, multi-farm produce dehydrator that doubles as an oven and uses a Raspberry PI. The user should be able to easily adjust the temperature and humidity of the drying chamber using a user-friendly online interface that may be accessed via a mobile device or the dehydrator's human-machine interface. They discovered that the developed system could successfully control the dehydrator's temperature and humidity. (Rao et al., 2021) have examined the cooking process as well, considering that it benefits from little automation, especially at the domestic level (household cooking). To facilitate the cooking of two common Indian food items, such as pancakes and rice cakes, they have created a prototype control system using PLCs, computers, and electro-mechanical components. (Priyanka et al., 2021; Soyguder & Alli, 2010) , on the other hand, used PLC combined with SCADA to construct a closed-loop PID control system for temperature monitoring, which is difficult using conventional control approaches. Data-driven PID control systems can be replaced by more precise forms of controllers at the cutting edge, concurrently with the adoption of Industry 4.0 technologies (Olaizola et al., 2022; Wakitani et al., 2019). The Model Predictive Controller (MPC) method, which uses a predictive algorithm to assess and optimize process variables, has also been used to implement predictive models (Maxim et al., 2018). For instance, (Khan et al., 2017) developed a fuzzy logic model that is integrated into a PLC real-time interface for process variables in nonlinear domains. A MPC algorithm was created using MATLAB, and its performance is contrasted with that of conventional PID and PID cascade controllers.

2.3 Machine Learning in food industry

The food industry is one of the most important and growing industries in the world, with a huge variety of products and production processes that require strict control to ensure quality, food safety and efficiency (Trienekens & Zuurbier, 2008). In recent years, Machine Learning (ML) has emerged as a key technology for improving production processes and optimizing operations in the food industry (Kumar et al., 2021). This article will explore the different types of machine learning used in this industry and their potential uses in production process control, predictive maintenance, and process monitoring. Figure 1shows the taxonomy of ML typologies:

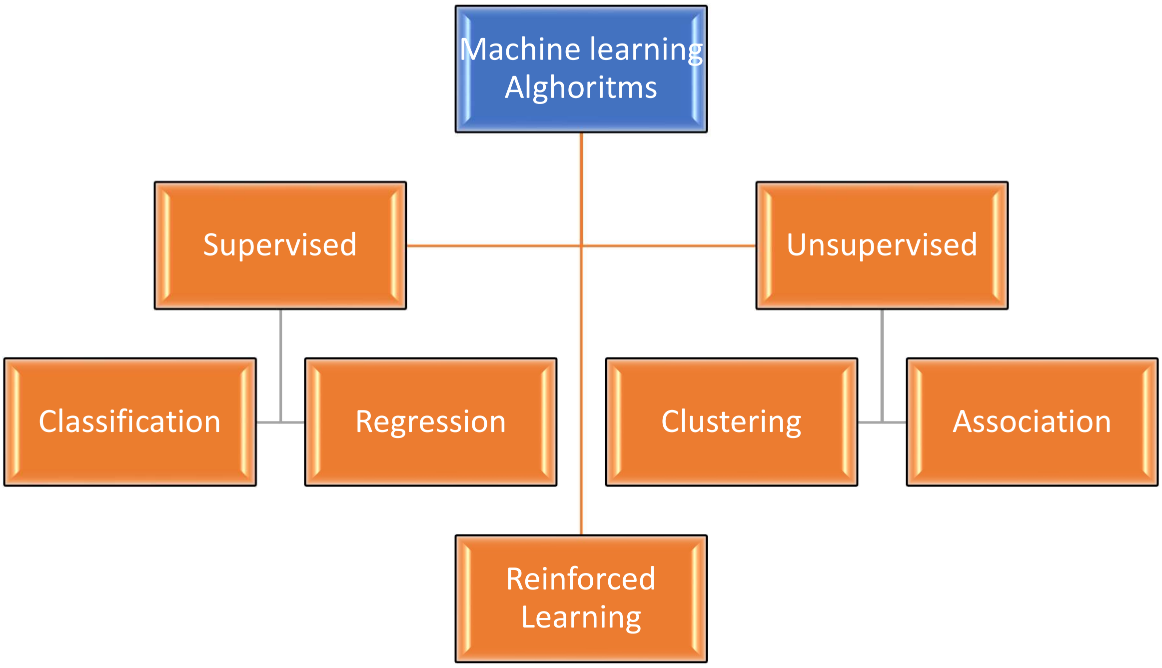


Figure 1 ML Taxonomy

Supervised learning involves training machine learning models using labelled data. It requires the analysis of the data collected on field, by an analyst which assign a category or a status, i.e., the label to each row of the data frame collected, it can be used for prediction. While unsupervised learning read the data collected and identify patterns, this kind of algorithm are useful to find relationships between the collected parameters. Reinforced learning can be employed for optimizing decision-making processes. In the food industry, it can be used to optimally manage supply chains and product distributions. ML can be used in production process control in several ways, such as process optimization, ML-based control systems can automatically adjust production process parameters to maximize efficiency and reduce downtime (Ayvaz & Alpay, 2021). Predictive analysis is crucial to avoid costly downtime and ensure production continuity (X. Han et al., 2021). ML can contribute in various way, such as:

- Condition Monitoring: by monitoring data collected on the filed thought sensors data allowing to have a real time machine status, rather than identify early signs of impending failures, enabling preventive interventions.
- Predictive Maintenance: ML can be used to schedule the maintenance time in an optimal production slot, such as machine format changes, to minimize downtime.

- Anomaly Detection: algorithms can predict and display trigger alert to users, in case of deviations from the established parameters occurs, helping to prevent process failure and or critical issues.

2.4 Data communication protocols

Data communication protocols allows to create the linkage between the physical layer and the digital one (C. Han et al., 2013). A wide variety of data communication protocols can be adopted in the industrial field (Caro, 2016). Those are:

- Modbus: A widely used serial protocol for communication between industrial automation devices, such as PLC (Programmable Logic Controller) and sensors.
- PROFINET: An industrial Ethernet protocol used for real-time control and monitoring of industrial devices, often used in industrial automation applications.
- TCP/IP: An industrial Ethernet protocol based on open standards that enables communication between industrial devices from different vendors.
- OPC UA (Unified Architecture): A communication protocol and interoperability framework used for sharing data and information between industrial devices and control systems.
- CAN (Controller Area Network): A serial protocol used in automotive and industrial applications for connecting distributed devices.
- Profibus: A serial communication protocol used for control and monitoring of industrial devices, often used in process automation applications.

For the cases under examination, each pilot requires a specific data communication protocol, i.e., TCP/IP and Modbus. The choice of the proper protocol has been established to fit with the compatibility of the already-installed machine components. The TCP/IP protocol is a suite of communication protocols that regulate the transfer of data between devices within a computer network or between different networks. They provide commands and fundamental principles for the operation of the Internet. The protocol is composed of a layered structure, each of which has a specific task (King & Hunt, 2000). These layers, listed in

Table 2 include:

Table 2 TCP/IP protocol layers

Link Layer:	This stratum oversees the physical transmission of data across a communication medium, such as cables or radio frequencies. It encompasses protocols like Ethernet and Wi-Fi, governing the tangible aspects of data transfer.
Network Layer:	This layer writes the navigation of data packets within a network, endowing devices with unique IP addresses and determining the optimal route for their transit. The predominant protocol operating at this stratum is the Internet Protocol (IP), serving as the network's cartographer.
Transport Layer:	This level delivers secure, connection-oriented correspondence between two devices. At this stratum, the Transmission Control Protocol (TCP) plays a pivotal role, managing error correction, data sequence, and the retransmission of information.
Application Layer:	This layer hosts application-specific protocols, such as HTTP for the web, SMTP for email, and FTP for file transfer. These protocols define how applications communicate and exchange data.

This protocol enables communication between devices on the network through the transfer of data packets. These packets contain data, destination information, and other metadata. The IP protocol manages packet routing, while the TCP protocol manages reliability and flow control during data transmission (Abdelsalam et al., 2017).

3 Programming environment

3.1 LabVIEW Environment

National Instruments developed the robust and popular graphical programming environment known as LabVIEW (Laboratory Virtual Instrument Engineering Workbench) (NI). It is a useful tool in many different industries, including research, engineering, manufacturing, and test automation, because it is specifically made for developing custom measurement and automation systems. In a visually intuitive environment, LabVIEW enables users to create applications that involve data acquisition, instrument control, signal processing, data analysis, and system integration. The use of graphical programming (G-Code) in LabVIEW is one of its distinguishing characteristics. Users can build software by dragging and connecting graphical nodes, also known as Virtual Instruments (VIs). The software employs a dataflow model, which means that rather than following a set of sequential instructions, a VI's execution is determined by the availability of data. As a result, complex data processing pipelines can be represented more simply and efficiently in parallel. NI hardware, such as data acquisition (DAQ) devices and instrument control modules, is frequently used in conjunction with LabVIEW. Users can easily interface with a variety of sensors, actuators, and measurement devices thanks to this tight integration. Also possible is the creation of reusable VIs. This modular approach encourages best practices in software engineering and speeds up development. Nevertheless, LabVIEW offers a variety of tools for data analysis, visualization, and signal processing. Applications involving measurements, control systems, and scientific research all require these capabilities. Despite being known for its graphical programming, LabVIEW also supports the integration of other programming languages, such as C, C++, and Python. Users can use pre-existing code or benefit from specialized libraries thanks to this feature. The following is a summary of the key elements involved in the development of the DT discussed in this work:

- Test and Measurement: LabVIEW is commonly used to create automated test systems for quality control, product testing, and validation.
- Data Acquisition: LabVIEW's data acquisition capabilities are ideal for gathering and analyzing data from sensors, instruments, and industrial processes.
- Control Systems: It is used to design and implement real-time control systems for various industries, including manufacturing and robotics.
- Algorithm's integration: LabVIEW is a valuable tool in research settings, enabling scientists to prototype and experiment with different algorithms and models.

- Academic Use: LabVIEW is used in universities and educational institutions to teach principles of data acquisition, control systems, and signal processing.

LabVIEW's combination of visual programming, hardware integration, and extensive libraries makes it a versatile platform for engineers, scientists, and developers working on a wide range of measurement and automation projects.

Virtual Instruments are programs written in LabVIEW (VIs). The reason these programs are called "instruments" is that they operate by presenting the user with an interface comparable to a measuring instrument; on the other hand, the word "virtual" describes the interaction as taking place with a running program rather than a specific physical equipment.

A VI consists of three fundamental parts:

- Front Panel
- Block Diagram
- Icon/Connector

To create a VI program, it is necessary to work on both the Front Panel and the Block Diagram to enable the software to process inputs, execute the program, and provide output data.

The window that serves as the interface between the user and the program is called the Front Panel, Figure 2, and it allows for interaction much like with a conventional instrument. The Front Panel includes indicators, which are output variables whose values are set by the program and cannot be changed by the user, and controls, which are modifiable input variables through the Front Panel. Using a keyboard or mouse, users can interact with the Front Panel to enter numerical numbers, strings, or modify how buttons and indications appear visually.

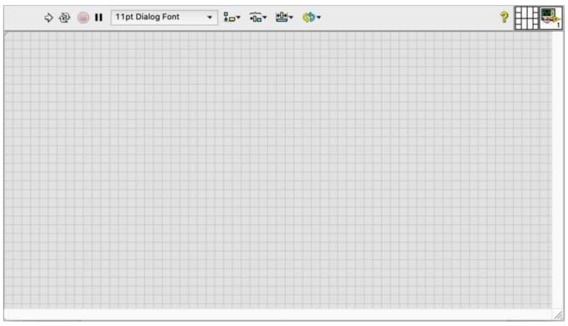


Figure 2 Front Panel LabVIEW

The Block Diagram,Figure 3, contains the code expressed in graphical language (G-Code) and consists of:

- Nodes: elements with inputs and outputs capable of performing operations. Which are the functions, instructions, operators, and subroutines in text-based programming languages, written in G-Code
- Wires: lines connecting the nodes, allowing the exchange of information. These wires have different colours and thicknesses depending on the data being exchanged. If the information you want to connect is incompatible, the wires appear dashed.

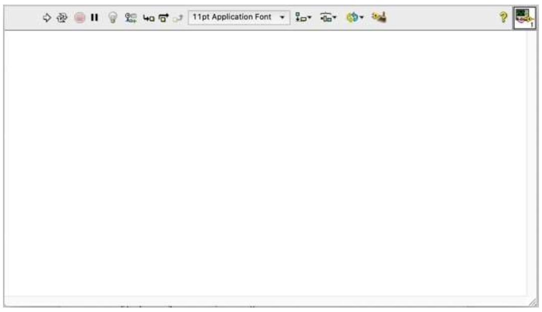


Figure 3 Block Diagram in LabVIEW

In the VI, the Icon/Connector,Figure 4, is the final essential component. To be more precise, the icon is a graphic symbol that turns the program into an item that complete the VI. The connector creates the connection between the icon's regions and the Front Panel's controls and indicators. Each VI can be used within another program to create a subroutine, i.e., a sub-VI. The sub-VI can be connected to other nodes, and in this case, the connector establishes a direct correspondence between an area of the icon and one of the input or output elements of the Front Panel associated with the sub-VI.

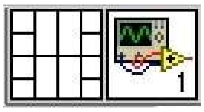


Figure 4 Icon/Connectorin LabVIEW

The Block Diagram, as mentioned earlier, contains the code of the software and is created using a graphical language. Right-clicking on the block diagram opens a window called the Functions Palette, which contains all the structures and functions provided to create the program. These can be thought of as "blocks," which are graphical elements, each representing a specific function. To create the code, it's necessary to connect various connectors and indicators to nodes or actual functions through a wiring operation. Measurement I/O functions let the system to communicate with external devices like pressure or temperature sensors, as well as data acquisition, data storage and data processing equipment like analog-to-digital converters (A/D). Measurement devices and software can share information through input and output interface. In particular, the data acquisition/generation driver, denoted by the icon "NI DAQmx" in the

Measurement I/O menu, can read input and sending output signals, among other functions. The measurement, production, and processing of data can be accomplished by programming these routines or by using a block called "DAQ Assistant," shown in Figure 5, which is included in the NI DAQmx section's Measurement I/O menu. A window for initializing the Data Acquisition block opens once users click and drag the DAQ Assistant symbol onto to the Block Diagram. This enables the configuration of the measurement type, measurement channels, number of samples to be acquired, and signal of interest.

Figure 5 DAQ Assistant block

Once the information is inputted, the DAQ proceeds with building the VI, which means creating the programming code for the acquisition/generation of that specific signal. Through I/O measurement systems, it is possible to create code for reading and processing both input and output signals using a data acquisition module, and controlled device, which are the hardware components of the LabVIEW environment.

The main components are:

- CompactDAQ.
- Analog Input module
- Analog Output module.

National Instruments created the PC-based modular data collecting platform known as CompactDAQ (cDAQ). On this PC, LabVIEW software is loaded on a Windows embedded system operator. This device allows to analyse and acquire data from different types of sensors, such as digital or analogue one, and trigger output signals useful to control drive.

The cDAQ can be connected to a laptop or desktop which, in turn, will run the software that configures, acquires/generates and records the data from the cDAQ itself. As already mentioned, this platform is modular, and this allows configuration with various modules for the acquisition/generation of different types of data relating to different sensors such as thermocouples or RTD, pressure transmitter, accelerometers with dynamic acquisition and an IEPE (Integrated Electronics Piezo-Electric) excitation, current transmitter, whit 4-20mA signal output , or voltage transmitters with an output signal of 0-5V up to 0 –10V.

cDAQ consists of two main components:

- Chassis: In charge of coordinating data generation and acquisition from the modules, linking them, and interacting with a computer system.
- I/O Modules (Moduli I/O): These modules facilitate the connection between sensors and the cDAQ system. They are available with both digital and analogue inputs and outputs, as well as signal conditioning and integrated analog-to-digital converters.

The CompactDAQ-9133 data collecting system, displayed in Figure 6, was employed in the creation of the digital twin that is the subject of this paper. This system controls the timing, synchronization, and data transfer between a computer integrated into the system itself and a series of I/O modules. It comes with 16 GB of non-volatile memory and a dual-core Intel Atom processor. A maximum of eight I/O modules featuring digital and analogue inputs and outputs, counter/timer capabilities, and a Controller Area Network (CAN).

Figure 6 CompactDAQ-9133

For the data acquisition of the signals from the fields, i.e., the sensors, the module used and connected to the cDAQ is the NI-9208, 4-20mA current input module, Figure 7.

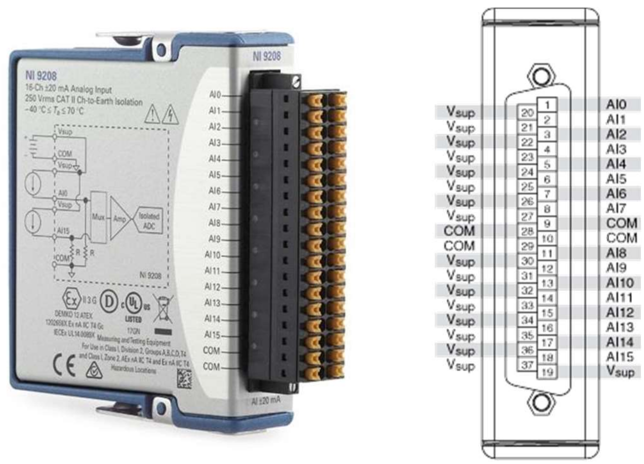


Figure 7 NI-9208 module and corresponding pinouts

While the module for the signal triggering is an AO voltage generator called NI 9263, it comes with four channels output with a digital to analogue converter capable of providing 0-10V signal output.

Figure 8 NI-9263 module and corresponding pinouts

4 Case studies

4.1 Pasteurization pilot plant

4.1.1 Plant Description

The implementations, simulations and experiments analyzed in the following chapters refer to a pilot plant located at the Technopole of the University of Parma, the headquarters of the Interdepartmental Centers for Industrial Research (Tancredi et al., 2022b). The term pilot plant refers to a small-medium size process plant whose task is to replicate the behavior of large industrial plants. The size and capacity of these plants are intermediate laboratory scale and industrial plant. The transition from laboratory scale to the industrial one, called scale-up, is performed with a view in reducing costs. A pilot plant is designed to simulate real machinery behavior and analyze processes from a predictive perspective to prevent and solve issues that arise from the tests carried out. It also allows for modifications, such as installing IoT sensors, data communication systems, internet connections, and hardware components for a 4.0 plant.

The plant examined in the project is a preheating system for a food fluid, called process fluid, through the transfer of heat by a service fluid, in our case, water. The preheating treatment is carried out before the microbial inactivation, performed thought a Pulsed Electric Field (PEF) treatment which guarantees the organoleptic characteristics of the product avoiding a thermal shock and the degradation of the food itself. The Figure 9 shows the Piping and Instrumentation Diagram (P&ID), which is a drawing of the interconnections between pipelines, equipment, and instrumentations.

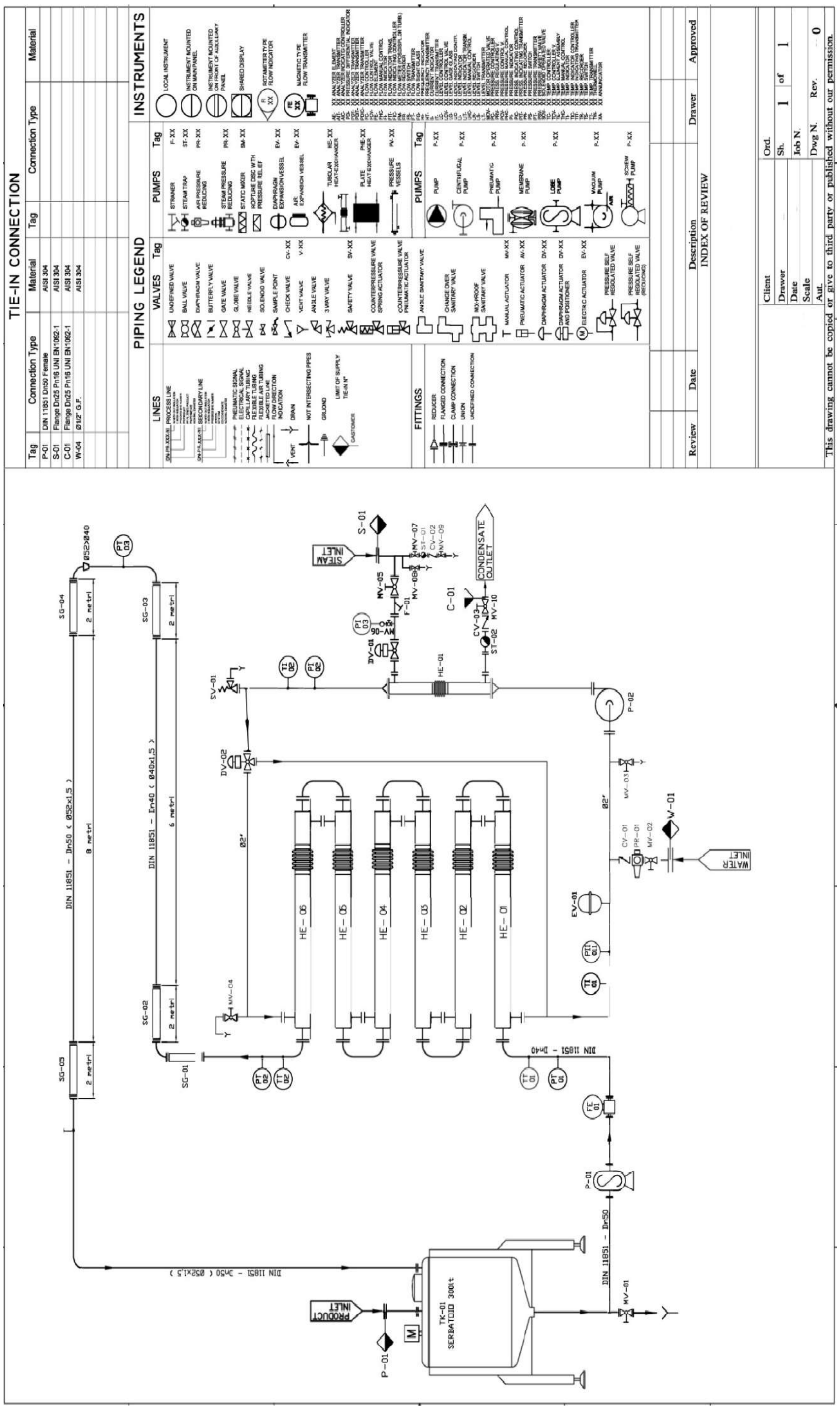


Figure 9 P&Id Pasteuritazation pilot plant

Piping and instrumentation diagrams play a key role during plant design, during the detailed design of control systems, and for operational and safety investigations, such as the Hazard and Operability study (HAZOP).

P&ID shows the following plant components:

Table 3 P&ID components

Mechanical equipment
Control instrumentation
Valves
Piping
Drainage, fittings, and purges
Direction of mass flows
Interconnections between systems

Each component has standardized symbols that are connected to each other through lines that represent piping and arrows that identify the fluid flow inside the piping itself.

The symbols used can comply with diverse types of standards, among the most important are the ISA S5 (Instrumentation, Systems, and Automation Society) and the BS 1646 (British Standards Institution). It should be noted that this flow diagram differs from the actual pilot plant in that the treated product does not return to the storage tank as shown in the P&Id.

Despite this, the P&ID shown shows the direction of the mass flow and how the heat exchange between the food fluid and the service fluid takes place.

The product, stored in a tank at room temperature (TK-01), is pumped through a tube in tube countercurrent heat exchanger via a double screw pump (P-01). The flow meter (FE-01), placed immediately after the pump, has the task of measuring the flow rate of food fluid flowing inside the pipes. Subsequently, the product arrives inside the heat exchanger in which the preheating takes place thanks to the counter-current passage of the service fluid, i.e., water previously heated by a steam generator and introduced into the system by a centrifugal pump. The product, once heated, reaches the next phase of microbial inactivation, carried out through an electrical impulse treatment (PEF – Pulse Electric Fields). PEF technology has the advantage of killing microorganisms and, at the same time, maintaining the flavor, color, texture, and nutritional values of unprocessed foods becoming an important alternative to the heat treatments normally used. In the following paragraphs, the main components of the system will be analyzed in more detail, i.e., the product transfer pump, flow meter, centrifugal pump, steam generator, heat exchanger and temperature and pressure sensors. The following Figure 10 show a photograph of the plant present in the university and its design.

Figure 10 Pilot plant located in Technopole.

4.1.2 Description of the main components

4.1.2.1 Twin screw pump

The fluid food is manually introduced into the system and stored in a 300-liter storage tank. Then moved through the piping by a twin-screw pump. This type of pump can be classified as volumetric one, has a body that houses two rotating screws. The chambers that are formed between the screws and the adjacent components allow the movement of the fluid from the suction side to the exhaust one. Reverse flow can be achieved by switching the shaft's direction. Figure 11 below shows the diagram of a double screw pump. The pump in the system is a Bornemann Screw Pump SLH4080.

Figure 11 Twin Screw Pump

4.1.2.2 Heat exchanger

The heat exchanger (HE) is the key component of the machine. It allows heat exchange between the food product and the water. The latter, after being heated, releases heat to the product and exits the exchanger at a lower temperature than the initial one.

There are different types of heat exchangers:

- mixing heat exchangers: the fluids are the same and mix with each other.
- surface heat exchangers: the fluids are separated by an impermeable surface to avoids their mixing.

The HE installed in the plant is part of the second category. It consists of six tubes formed, in turn, by two concentric ducts, (Figure 12); The service fluid flows into the annular section, while fluid food flows into the inner section of the HE. The heat is transferred through conduction to the wall that separates the fluids.

Figure 12 Tube in Tube Heat Exchanger

Tube in tube heat exchangers can be classified as:

- Equicurrent: fluids flow in the same direction.
- Countercurrent: fluids flow in the opposite direction.

In the case under examination, heat transfer takes place between fluids flowing in opposite directions. Table 4 below, shows the main dimensions of the HE:

Table 4 Tube in Tube Heat Exchanger Dimensions

Description	Measure	udm
Pipe Length	4000	mm
External Pipe: Outer diameter	76	mm
External Pipe: Inner Diameter	73	mm
Internal Pipe: Outer diameter	41	mm
Internal Pipe: Inner diameter	38	mm
Thickness	3	mm

4.1.2.3 Flowmeter

Then the product flows through a magnetic flow meter (Figure 13). The operation of this flowmeter is based on Faraday's law; The voltage induced by any conductor moving perpendicularly through a magnetic field is proportional to the speed of the conductor itself. According to this principle it is necessary that the fluid to be measured is conductive. In the system under analysis, the flow meter is from the 1300 OPTIFLEX series of the KRONHE Group.

Figure 13 Krohne Flowmeter

4.1.2.4 Centrifugal pump

The service fluid flows into the closed-loop system and is fed through a centrifugal pump. The movement of the fluid is induced by rotating mechanical organs called impellers which allow the transformation of mechanical energy into kinetic energy and then into pressure energy. The pumped fluid enters the center of the impeller and is accelerated, thanks to the curvature of the blades, in a radial direction increasing its average speed (kinetic energy). The water is then slowed down thanks to the increasing section of the pump body allowing the transformation of kinetic energy into pressure energy. The plant involves the installation of an SKB Etabloc monobloc centrifugal pump (Figure 14) with a maximum flow rate of 1200 kg/h.

Figure 14 Centrifugal Pump

4.1.2.5 Heat Steam Generator

The steam generator in the plant has the task of producing saturated steam necessary to heat the water. This consists of three 15 kW heating element packs and a variable pressure range between 0 and 11 bar. In the front panel you can see two values, a green number corresponding to the real pressure and a red one equal to the set point value set by the operator. The water temperature is therefore not set directly but depends on the saturated steam pressure. In general, a liquid consists of particles moving with a certain velocity and therefore kinetic energy. Evaporation occurs when these particles have sufficient energy to escape the attractive action that is generated between the particles themselves. These, meeting the liquid again, can return to be part of it thus generating the condensation phenomenon. Dynamic equilibrium is created when the number of evaporated particles corresponds to the condensed ones. The equilibrium condition that is created between steam and liquid at a given temperature is called saturated steam. The saturated vapor pressure of a liquid increases with temperature as the kinetic energy increases and therefore the tendency to evaporation. The relationship between pressure (p_v) and temperature ($T_{[°c]}$) is expressed using the following Clausius Clapeyron:

$$p_v = 6.11 * 10^{\frac{7.5T}{237.7T}}[mbar] \tag{1}$$

By setting a pressure on the steam generator, therefore, it is possible to obtain, through the inverse formula, the corresponding temperature value.

$$T_{[°c]} = \frac{237.7(\log_{10} p_v - \log_{10}6.11)}{7.5 - \log_{10} p_v + \log_{10}6.11)} \tag{2}$$

The following image (Figure 15)shows the heat steam generator.

Figure 15 Heat Steam Generator

4.1.2.6 Sensors

Pressure sensor

The pressure sensors in the system are WIKA S-11 flush diaphragm pressure transmitters (Figure 16). This type of sensor is characterized by an internal membrane, an inlet channel sealed by a second membrane exposed to the process and a transmission fluid having the task of transferring the pressure to the inner membrane.

Figure 16 Flush diaphragm pressure sensor

The fluid, flowing orthogonally to the sensor, compresses the membrane of a given Δ. The potential difference that is generated is converted into a 4-20mA analog current signal which allows connection to different control systems. The use of flush diaphragm pressure transmitters has the advantage of isolating the measuring instrument from the process fluid, avoiding deposits by viscous or crystalline fluids and damage due to aggressive fluids. The facing membrane can be made of special materials or be coated to prevent fluid from entering the attachment and damaging the sensor.

Temperature sensor

The most used temperature sensors in the industrial field are platinum resistance thermometers or thermometers with platinum electric resistances. This metal is widely used as it has constant electrical characteristics with varying temperatures. The most used resistance thermometers are Pt100 (Figure 17); this indication indicates the material used and the nominal resistance relative to a temperature range from 0°C to 100°C. The operation is based on the variation of the electrical resistance of the metal as a function of the temperature detected. The two quantities are linked by a linear characteristic and as one increases, the other also increases. For the case under examination, PT100 has a current-loop 4-20 mA output. As described earlier, the PT100 is a resistive temperature sensor with resistance that varies linearly with temperature. The sensor measures the variation of resistance, which is converted to current though the signal condition system i.e., the Current Transmitter. This device converts the resistance variation of the PT100 into a continuous current (4-20 mA) proportional to the detected temperature. The signal output is then connected to the DAQ hardware via 2 wires connection.

Figure 17 PT100: Temperature Sensor

4.1.3 Digital model

The digital environment has been developed considering the basic principles of the rheological properties of the fluid food and the pressure drop of the system, aiming to control and monitor the machine behavior thought the main parameters involved in the process, i.e., the fluids temperature (both fluid food and facilities), pressure at the inlet and outlet section of the HE, and the fluid food flow. In accordance with the study's objective, as mentioned previously, this paper suggests incorporating machine learning into (DT) framework which simulates the operation of the pilot plant in examination.

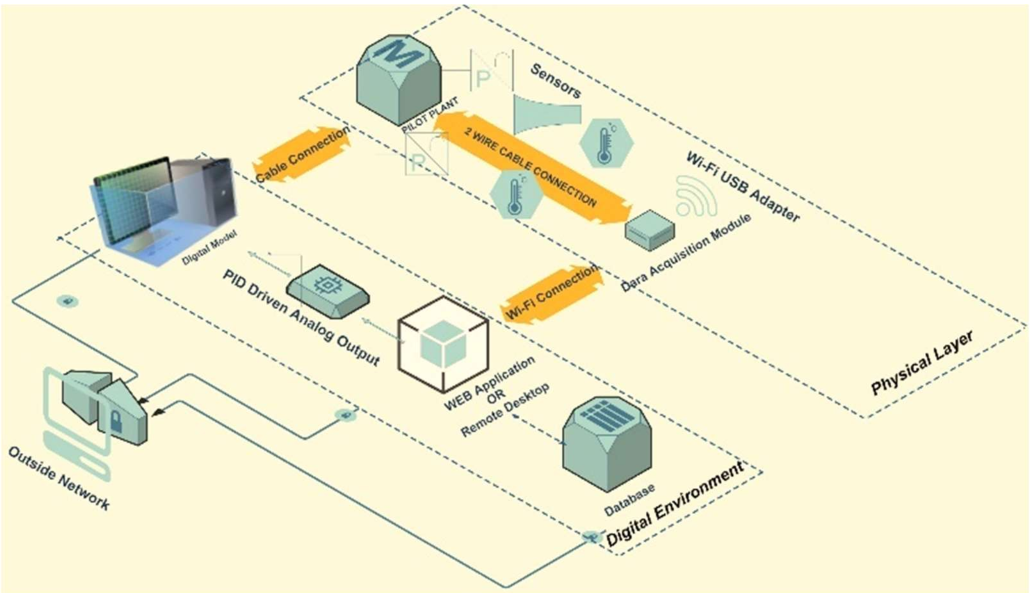


Figure 18 DT model

DT's model is composed by two layers, shown in (Figure 18), capable to interoperate and communicate each other: Those layers are:

1. The physical layer: which is represented by the plant itself, and all the components mounted, such are sensors, DAQ (Data Acquisition) and a Wi-Fi antenna to provide internet connection.
2. The digital layer: which is the counterpart of the physical one. This has been developed thought LabVIEW and Python programming language. LabVIEW is Human Machine Interface (HMI), i.e., the control panel has been coded via "LabVIEW front panel," displayed in Figure 19, while the "block diagram," forms the software's underlying code, written in G-language. As depicted in Figure 20), the system offers functionality in four distinct ways.

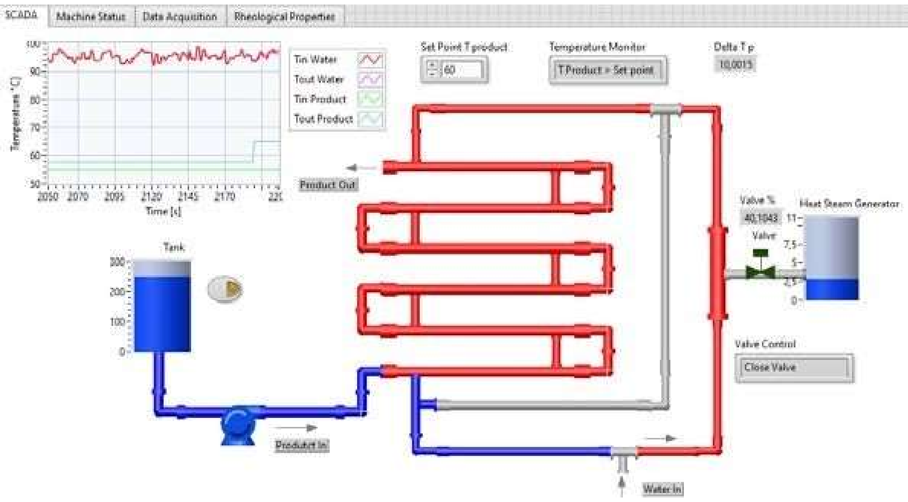


Figure 19 Digital Model Pasteurization plant

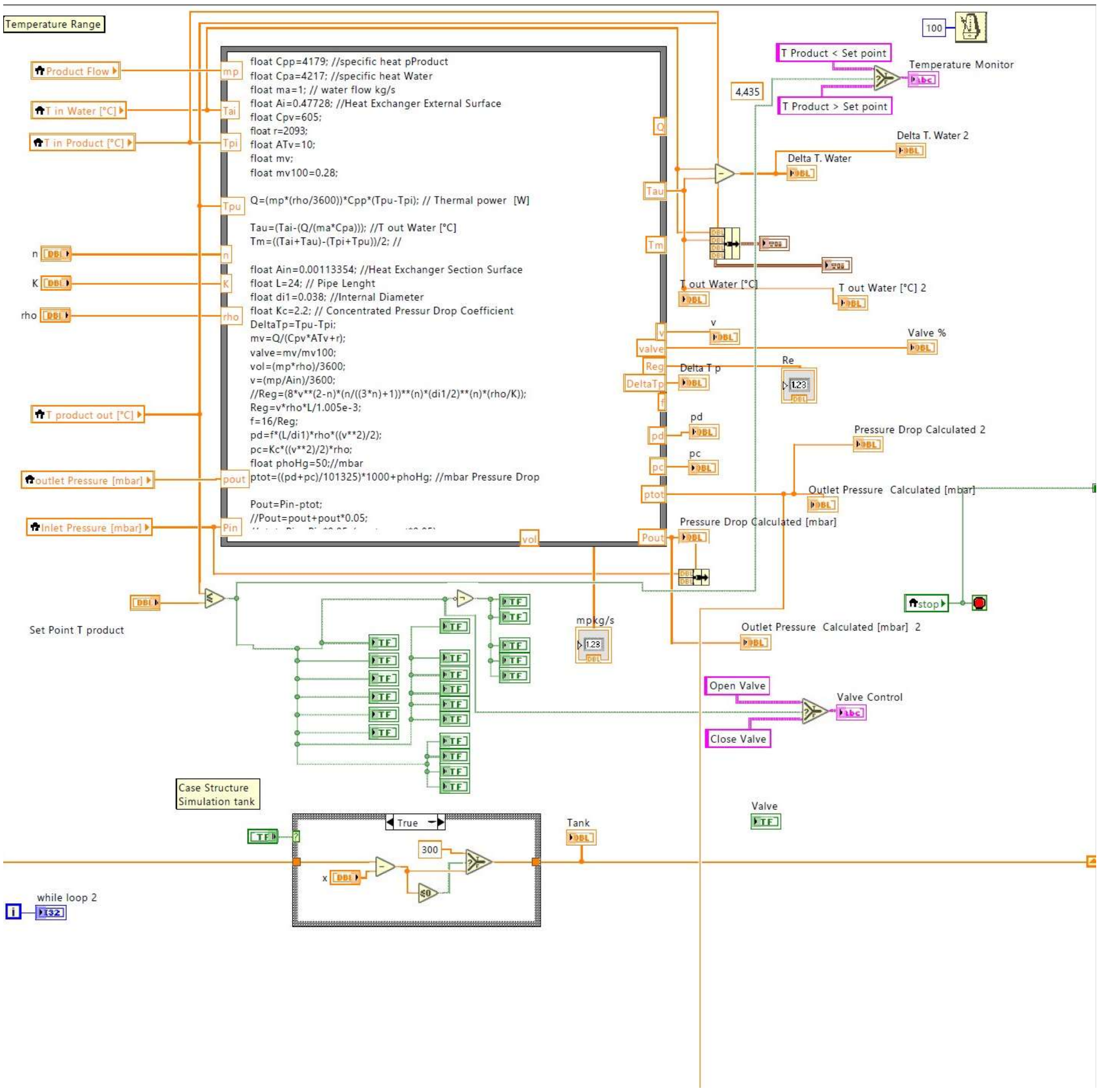


Figure 20 Pasteurization plant block diagram

By selecting the first mode the system acts as a model simulation environment. Here, the DT model's equations replicate product's physical properties based on flow and temperature, providing an output that evaluates the heat exchanger's pressure drop considering the pasteurization system's geometry and fluid rheological properties. Users can also simulate machine conditions by adjusting process parameters or generate an AO (Analog Output) to control Pump's motor, or adjust the setpoint of the controller, which is a proportional-integral-derivative (PID) one. The second function involves "Real-time monitoring" where measures are acquired directly from the field. Based on these signals and manual PID setpoint adjustments, the digital environment generates an AO to control the product flow by driving the motor pump.

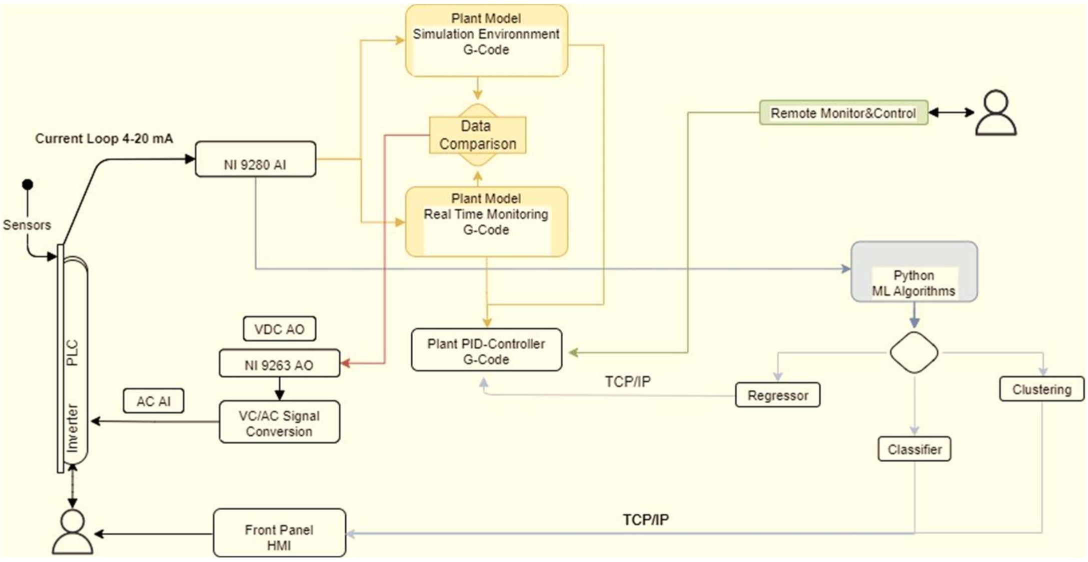


Figure 21 Digital Twin Architecture

The system can also trigger an AO signal using the "Data comparison" tool, comparing the analytically computed pressure drop via DT with the one evaluated with the-acquired values, and adjusting the pump speed accordingly. The third mode, labeled "Remote monitor and control," permits machine status control and voltage output generation through remote connection. In the fourth mode, the system is integrated with an ML-based algorithm to run the motor pump or show messages on the pilot plant's front panel (HMI) automatically. As seen in Figure 21, this "ML algorithm" makes use of Python and G-Code. Three ML models—a linear regressor, a classifier, and a clustering algorithm—have been applied whit the aim of estimating which one performs better in terms and return valuable insights for users.

4.1.3.1 Rheology

The term rheology indicates a branch of physics specialized in the study of deformations of solids and fluids because of the application of external forces. For liquids, rheology deals with the relationship between stress state and strain rate. This science plays an important role in the food sector and is useful for various reasons; It allows you to have a deeper knowledge of the molecular and microscopic structure, allows you to control the quality of products and processes, guarantees a suitable design of machinery with the characteristics of the food and, finally, makes a product acceptable or not according to consumer needs. The following paragraphs deal with the relationship between shear stress and strain rate for two different types of fluids; Newtonians, governed by Newton's law and not Newtonians, governed by the law of power. About the fluids present in the project, water is part of the first category, while most of the food fluids fall into the second.

4.1.3.1.1 Newtonian fluids

Fluids are defined as Newtonians when the shear (or tangential) stress is directly proportional to the rate of deformation (Malkin, 2013). We consider two parallel planes, one fixed and one movable, and a fluid moving between them according to parallel layers. The fluid velocity profile is shown in the image below (Figure 22).

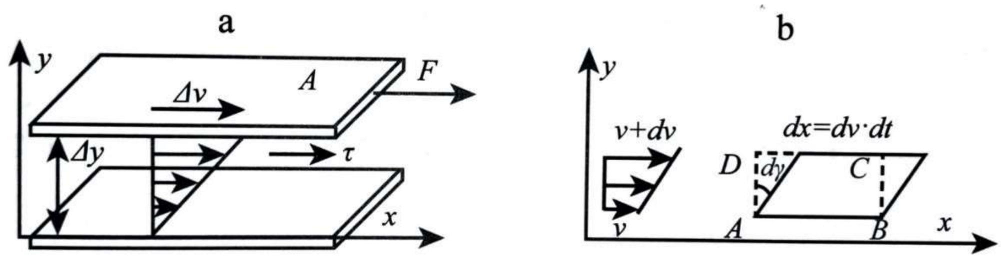


Figure 22 Velocity profile of the fluid moving between two plates, one of which is in motion; (b) extrapolation of a part of the velocity profile to the left and sliding deformation movement of an ABCD element to the right

As it can be seen, the fluid adhering to the upper plane moves with the same speed, while the molecules in contact with the lower plate have zero velocity. To keep the upper plane moving with constant velocity Δv, it is necessary to apply a force dependent on the speed, the area of the plate and the characteristics of the fluid enclosed within a quantity called viscosity μ and inversely by the distance between the two Δy planes. Dynamic viscosity [Pa ∗ s] measures the resistance of the fluid to creep and depends on the chemical-physical nature of the fluid, pressure (creep resistance increases with pressure, but since liquids are less compressible than gases, viscosity is considered independent of p), temperature (for liquids the ratio is inversely proportional), by the gradient of speed and time.

4.1.3.1.2 Non-Newtonian fluids

On the opposite, nonlinear correlation between shear stress and strains rate identifies non-Newtonian fluids (de Souza Mendes, 2007). Pseudoplastic fluids are the most recurrent among food fluids and correspond to aqueous suspensions of coarse particles. Consider the experiment carried out for Newtonian fluids and imagine that between the parallel planes is placed a pseudoplastic fluid; If the speed is low, the cells create a lattice that can reduce the flow and for this reason the fluid has a high viscosity. As the speed increases, however, the particles are arranged in the direction of motion and the viscosity decreases until it settles at a constant value. The viscosity therefore decreases with increasing speed and vice versa. The law relating shear stress and velocity gradient is called the Power Law:

$$\tau = K\left(\frac{\delta v}{\delta y}\right)^n \quad (3)$$

Where:

τ = shear stress [Pa]

K = consistency factor [Pa ∗ s^{-1}]

n = behavior index

$\frac{\delta v}{\delta y}$ =the velocity gradient

Especially:

- n = 1 we get Newton's law
- n > 1 dilating fluids
- n < 1 pseudoplastic fluids

As shown in Figure 23, another category of non-Newtonian fluids is that relating to dilating fluids; unlike the previous ones, the viscosity increases with increasing velocity gradient. In addition to these there are fluids with time-dependent characteristics; thixotropic, rheopectic and viscoelastic fluids.

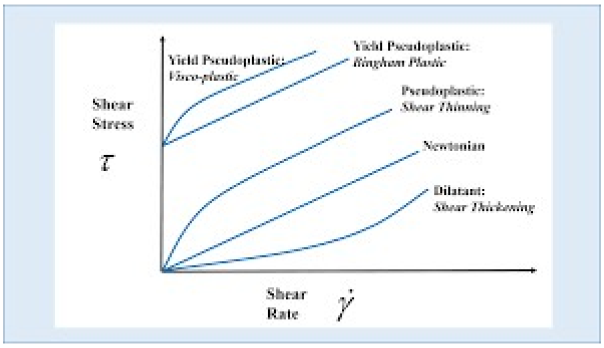


Figure 23 Non-Newtonian Fluid

Thixotropic have a decrease in viscosity over time and a structure capable of disintegrating under the effect of tangential forces. Rheopectic, on the contrary, are characterized by an increase in viscosity over time, while viscoelastic, also called fluids with memory, recover the original shape once the force acting on them is released. Finally, the last category is represented by Bingham fluids or plastic behavior. These fluids begin to flow when the force exerted exceeds a τy flow limit and then continue as a Newtonian, dilating or pseudoplastic fluid.

4.1.3.2 Pressure Drop

The primary factor to be considered for the system's evaluation of pressure drop is directly connected to the fluid food's non-Newtonian behavior. The first step involved in the pressure drop evaluation, deals with the calculation of the generalized Reynolds Number (Re_g) which is a dimensionless number identify the flow regime.

The equation (4) for the Re_g:

$$Re_g = 8^n \cdot \left(\frac{n}{3n+1}\right)^n \omega^{2-n} R^n \frac{\rho}{m} \tag{4}$$

where:

- ρ is the fluid density.
- *n* is the flow index.
- *m* is the fluid food consistency factor.
- *R* is the piping radius of the tube side in the HE.
- *ω* is the average velocity of the fluid.

The friction factor (*f*) refers to the energy losses brought on by friction inside the tube. It can be evaluated with the following relation (5):

$$f = \frac{64}{Re_g} \tag{5}$$

Once friction factor and Reynold has been calculated, it follows the pressure drop estimation using Darcy-Weisbach equation (Daneshvar 2023), which relates system geometry (tube length and diameter), fluid velocity, to f as follows (6):

$$\Delta P = \frac{f}{2}\frac{L}{D}\omega^2 + \sum_i k_i \frac{\omega^2}{2} \tag{6}$$

where:

- *L* is the length of the tube.
- *D* is the hydraulic diameter of the tube.
- k_i is the resistance coefficient of the piping.

The Block Diagram developed in G-Code incorporates all the above-mentioned equations.

4.1.3.3 Data communication system

For the case under examination a TCP/IP data communication protocol has been implemented in the Digital Environment. This protocol enables communication between devices on the network through the transfer of data packets. These packets contain data, destination information, and other metadata. The IP protocol handles packet routing, while the TCP protocol manages reliability and flow control during data transmission. This protocol enables data exchange between a host computer and the data acquisition module (NI-9208) for data analysis. At this purpose, it has been coded two block diagram Figure 24, in particular: one for data transmission on the data acquisition module and another as a TCP/IP receiver (Figure 25). TCP/IP infrastructure has been specifically designed to enable real-time data collection from the plant, store it on a client host, and enable output prediction using real-time input and ML algorithm.

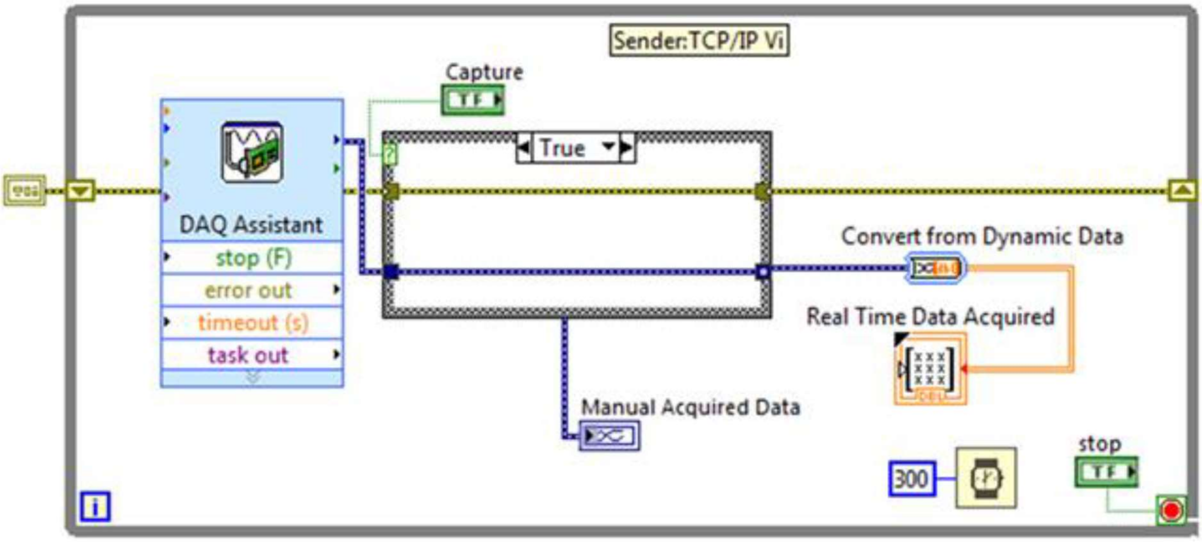


Figure 24 Block Diagram TCP/IP Sender

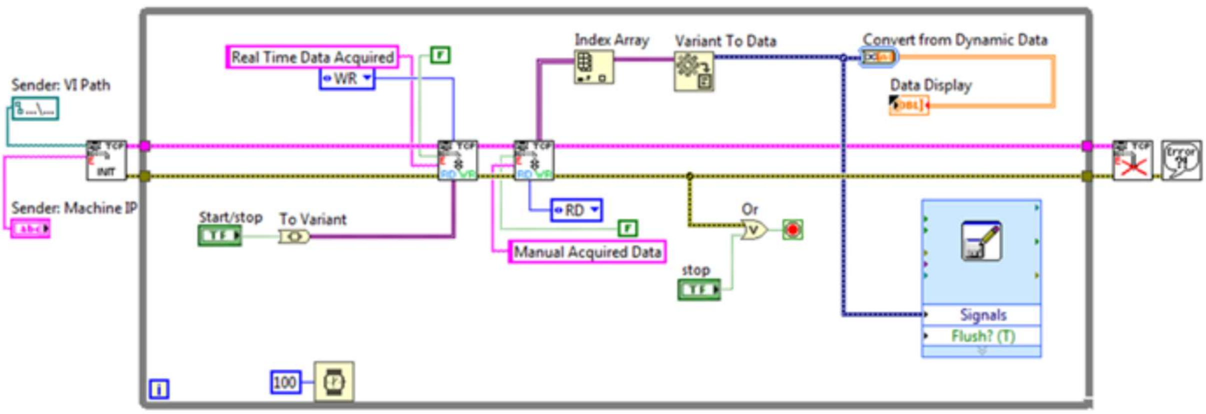


Figure 25 Block Diagram TCP/IP Receiver

4.1.3.4 Process Control strategy

Pant's control strategy is a PID (Proportional-Integrative-Derivative) based system. The coefficient of each controller parameter, which are, K_p for the proportional, K_I for the Integrative and K_d for the derivative one, are tuned using trial-and-error methodology. The HMI specifically displays the machine's behavior and pressure status: under normal working conditions, the model shows the components in blue and green colors, but in the event of an anomaly, the DT shows the components in red. The pressure difference between the machine's input and output determines whether it operates correctly or incorrectly. For example, if fluid contains foreign particles (such as small stones, an increase in pressure at the HE's inlet section could indicate piping obstruction. To accurately represent this situation, the pressure at the pump's outlet (which is at atmospheric pressure) must be monitored. Its value will be compared with the one provided by the DT model, which is based on the Navier Stokes equations and fluid rheology. (Tancredi et al. 2022b). Nevertheless, the front panel's control buttons allow the user to set the appropriate target for the desired parameters, such as the product temperature. The developed control device is a specialized system designed for processing information, with the purpose of regulating physical process variables. It can communicate with the outside environment and includes a set of functionalities, such as closed-loop control, reference value estimation (set points), management of warnings and occurrences, and machine to machine (M2M) or machine to user communication. In today's technological landscape, there exist numerous types of control systems, ranging from simple to highly sophisticated, constructed using various techniques. Although some control systems are implemented using computer-based tools, others persist in using conventional hydraulic or pneumatic technologies. This work, consider Proportional-Integral-Derivative (PID) control methodology. The PID strategy, show in Figure 26, involve three main gains, which trigger an output signal based on a proportional relation within the error and the proportional, integrative, or derivative part of the error, respectively. In the ideal input/output relationship, these errors components are mathematically represented as follows (7):

$$\theta_{\mathrm{out}} = \mathrm{k_p E\,(t)} + \mathrm{k_i E}\,(t)\mathrm{dt} + \mathrm{k_d\,E}\frac{\mathrm{Et}}{dt} \tag{7}$$

Where:

θ_out = output signal

E (t) = input error

kp = proportional constant

ki = integrative constant

kd = derivative constant.

A response that is proportional to the input error (denoted as P), resulting in a proportional control system, is an efficient strategy for implementing the control system. However, increasing the proportional gain's value, kp, may lead to difficulties in reaching a stable equilibrium at the reference value.

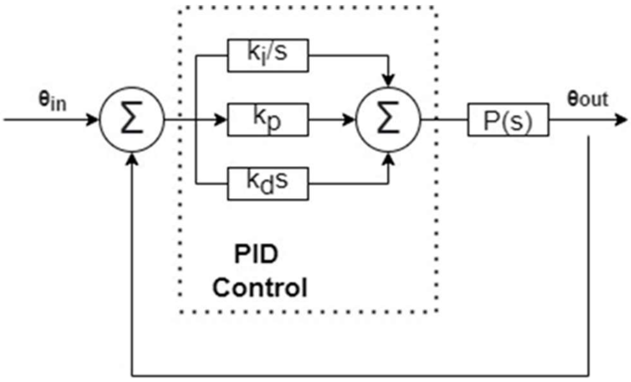


Figure 26 PID scheme

To address this issue, a straightforward solution is to adjust the output signal by incorporating an additional term, denoted as r (referred to as reset). With this adjustment, the system can be stable at the set point value. Following the abovementioned considerations, the target output can be written as (8):

$$\theta_p = k_p E\,(t) + r \tag{8}$$

The Virtual Instruments PID (VI PID) and its corresponding G-Code, shown in Figure 27 are the main parts of the implemented control system. The position of the output signal generator at the PID output allows it to achieve the adequate output voltage for the automatic and manual functions of the digital layer.

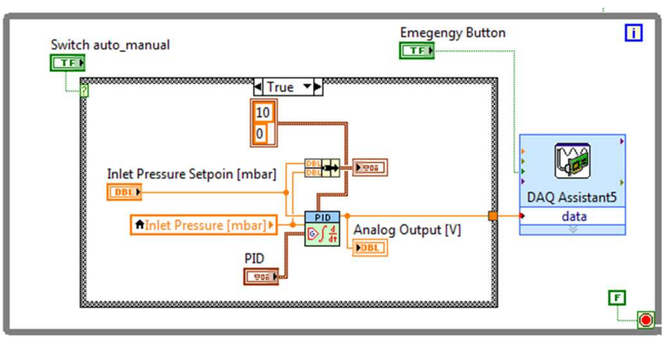


Figure 27 G-CODE: PID and Signal Trigger

A PID controller can be tuned using a variety of techniques, each with their own strategy. The Ziegler-Nichols method and the Tyreus Luyben's method are two widespread approaches:

1. Ziegler-Nichols Method (Ziegler & Nichols, 1942): During the first stage of this method, the integral (ki) and derivative (kd) gains are both set to zero, focusing on the proportional component of the controller. The proportional constant's value (K_p) is then gradually increased until the system oscillates steadily around the desired setpoint.

2. Tyreus Luyben's Method (Tyreus & Luyben, 1992): It uses process like the Ziegler-Nichols method described earlier. This approach prioritizes the integral component over the other ones.

Both methods rely on approximate identification of the system's dynamics through experiments, either in open or closed-loop configurations. When operating conditions change significantly or the system is time-varying, re-calibration of the PID controller may be necessary. This can be accomplished using an auto-tuning method, where calibration occurs automatically or is initiated by user commands.

Another simple approach to PID tuning is the trial-and-error method. In this method, the PID coefficients are adjusted iteratively.

Initially, the focus is on the proportional gain (kp) until the control loop exhibits constant-rate oscillations. Once stable oscillations have been achieved, focus is directed toward fine-tuning the integral (ki) and derivative (kd) components to reduce overshoot and produce a stable response. The developed control can be used in two ways:

1. Remote Desktop Connection: This method involves using a remote desktop connection enabled by the Microsoft Windows suite. It allows a user to access and control the digital twin as if they were physically present at the machine. This is a common way to remotely control and monitor systems and applications on a Windows-based computer.

2. Remote Data Acquisition via LabVIEW Web Services: The second approach makes use of LabVIEW's built-in support for Web Services-based remote data acquisition. The software platform LabVIEW is utilized for automation, instrument control, and data acquisition. With the help of LabVIEW Web Services, any device connected to the server via Wi-Fi, Bluetooth, or Ethernet can communicate via HTTP. The PID controller's output value is transformed into an analog signal in the 0V–10V DC range in both techniques. The physical plant (where the control signals are applied) and the digital layer (where the PID controller operates) are connected thought a signal converter (PXU-20.924), and the output generator module (NI9263 by National Instruments). This hardware configuration ensures that the voltage output is converted into a

current signal that can be carried to the analog input port of the Danfoss inverter and ranges in value from 0 to 20 mA. By altering the motor's pump's rotational speed, the Danfoss inverter regulates the frequency of the power supplied to the motor.

4.1.3.5 Machine learning algorithms

For the case under examination, three ML models were developed using Python along with external libraries. Accessible at https://scikit-learn.org/stable, the scikit-learn package (accessed on January 18, 2022) provided the setting for the development of ML. Each machine learning algorithm, especially the supervised ones like linear regression and artificial neural networks, has a specific structure that is divided into three main parts. A code that can import data from the dataset containing values gathered during the experiments makes up the first section. To make the data suitable for the algorithms' processing, some preliminary data preprocessing steps are carried out. Data formats, for instance, are transformed into the appropriate type (data of the object type), and process parameter variables are changed into floating-point data. The dataset is split into two subsets in the second section of the script: the training set, which contains 70% of the original data, and the testing set, which contains the remaining 30%. The data from the training subset is then used to train the ML model. The final section of the script is used to make predictions using the model after the algorithm has been trained.

On the other hand, the third algorithm is an unsupervised clustering model, namely a k-means clustering.

This type of model does not require a predefined output for training and does not involve a testing phase. This algorithm's predicted result for our application is the categorization of data triplets into one of three categories: "ok," "warning," or "alert." Since there is no testing phase, the only way to determine whether the algorithm's conclusions are accurate is by contrasting the clusters it finds for data triplets with the precise machine status that was initially entered into the database. The algorithm should ideally divide the triplets of data into three clusters that represent the "ok," "warning," or "alert" statuses. The potential benefit of this approach is that, if the clustering algorithm correctly classifies the data, there would be no need to further categorize the data into "ok," "warning," or "alert" categories in subsequent applications.

Three different fluids were used in a series of experiments to create the dataset needed to apply ML algorithms. In particular, the machine was tested with three different materials: water (referred to as "Fluid 1"), two mixtures of water and a food additive (Gellan Gum) in varying concentrations to mimic non-Newtonian food fluids, and a combination of water and both substances alone. Gellan Gum was present in "Fluid 2" at a mass composition of 0.1 percent while it was present in "Fluid 3" at a mass composition of 0.15 percent. An average of one sample per second was used for sampling while the sensors continuously recorded signals. To account for the additive's temperature-dependent rheological properties, the machine's inverter frequency was changed during testing in increments of 10% and three different operating temperatures were set.

Table 5 Temperatures and setting for data acquisition during testing.

Description	Test 1	Test 2	Test 3
T [°C]	20	40	60
Inverter Frequency [%]	0–100%	0–100%	0–100%
Sample Rate [Hz]	1	1	1

Table 5 presents the variations in parameters for each data collection run, including operating temperature, inverter range, and sample rate. Data collection involved altering one process parameter at a time while keeping the others constant. Adjustments were made to the inlet valve to allow the inlet pressure to be changed between 0 and 185 mbar without affecting the outlet pressure or product flow. A manual valve was used to gradually close the heat exchanger's outlet section to alter the outlet pressure. To prevent machine malfunction or harm to the operator, safety precautions were taken. The product flow was modified by closing the inlet product valve.

The collected data from machine tests were first preprocessed with the following steps:

1. Examination of each set of collected data (flow, inlet pressure, and outlet pressure).

2. Addition of a "label" column to each set of values to describe the machine's status based on parameter values and the testing scenario. Normal conditions were labeled as "ok," transitional phases as "warning," and anomalies as "alert."

3. Description of collected data by calculating their average, standard deviation, minimum, maximum, and percentiles (25% and 75%). These statistics are summarized in Table 6 for the three fluids.

4. Visualization of data using 3D plots in Python to provide an initial overview of data distribution.

The final preprocessed dataset for each fluid had 6256 rows and four columns. The machine's status and process parameters were represented in the columns. This data was saved in Microsoft Excel 2016 as a.csv file.

Table 6 provides a summary of the collected data, including counts, means, standard deviations, minimum and maximum values, as well as percentiles.

Table 6 Summary of the collected data

	Count	Mean	Std	Min	25%	50%	75%	Max
P1 Water	6256	70.30	40.7	0	50.01	50.02	50.03	182.38
P1 Fluid2	6256	145.68	41.52	52.9	137.65	151.45	169.2	250
P1 Fluid3	6256	154.8	47.21	49.07	141.14	162.27	185.9	250
P2 Water	6256	14.08	21.31	3.27	3.36	3.40	3.42	68.54
P2 Fluid2	6256	47.61	19.32	0.06	53.87	55.79	57.24	80.52
P2 Fluid3	6256	51.87	27.33	0.49	53.81	56.78	58.82	250
F Water	6256	0.29	0.58	0.007	0.008	0.009	0.011	1.55
F Fluid2	6256	1.49	1.11	0	0.69	1.48	2.33	3.62
F Fluid3	6256	1.51	1.11	0	0.56	1.53	2.33	3.53

In the subsequent machine learning phase, three ML models were developed using Python and the scikit-learn package. These models included linear regression, artificial neural networks, and an unsupervised clustering (k-means) model. The machine learning process involved data import, preprocessing, splitting the dataset into training and testing sets, model training, and prediction. The k-means model aimed to classify data into "ok," "warning," or "alert" categories without the need for training on labeled data.

The prediction made by this layer depends on the processing and algorithm used, with two possible results. A product flow adjustment as outcomes of the regressor algorithm, and a message box displaying the machine's status produced by the classification and clustering one. The system's status can be classified into one of the following groups according to the evaluation of flow rate (F), inlet pressure (P1), and outlet pressure (P2) combined:

- Ok: The machine is operating properly because the parameters are in the right range of values. The DT model defines "correct" operation as values that deviate from the computed value of the pressure drop, by no more than 10% (in absolute terms).
- Warning: One or more parameters deviate from the computed values of the DT by 10 to 25 percent (in absolute terms), which is outside the defined correct functioning range. This condition could be an indication of operational anomalies, such as transitional phases as the plant changes from one steady-state condition to another and leads to parameter deviations.

- Failure: The machine encounters a critical issue requiring shutdown. This happens when the difference between the real and computed pressure drop is greater than 25%.

There are no accepted standards, despite the classifications of machine functioning status proposed by various authors in the literature. In addition, a wider range of values (75 percent of cases) have been assigned to the failure category in comparison to the warning and ok statuses to improve the detection of plant malfunctions (i.e., failures) rather than the other categories. The TCP/IP receiver G-Code on the server transmits the pertinent data, i.e., the detected status, to the front panel HMI.

In contrast, the regressor model provides a value representing the estimated output variable, i.e., pressure at the outlet section of the HE. The setpoint of the controller can be modified using this value in accordance with the condition of the machine.

4.1.3.6 Testing phase

The developed digital twin has been tested in all its components which are basically the control system and the anomaly prediction tool developed with the ML algorithms. Sections below describe the testing setup, testing methodology and model validation carried out following specific metrics.

4.1.4 PID Control Test and results

The PID controller gains were determined through an iterative trial-and-error method. Initial settings for the integral (I) and derivative (D) terms were zero, and the proportional (P) gain was gradually increased until the output displayed steady oscillations within a range of values that was deemed acceptable. Initial settings for the integral (I) and derivative (D) terms were set to zero, while the proportional (P) gain gradually increased until the output displays steady oscillations within an acceptable range. Increasing the proportional gain allowed the system to respond more rapidly to changes in the process variable, resulting in clear oscillations around the setpoint. Subsequently, the integral term was introduced to diminish steady-state errors and reduce these oscillations. After setting an appropriate P value to achieve the desired response, minor adjustments were made to the integral term to establish a steady state, all while considering the potential increase in overshoot. Finally, the derivative term was fine-tuned to ensure the feedback system remained stable throughout the control process. Following these steps, the PID gains were determined as follows in Table 7:

Table 7 PID Gain Coefficients

K_p	0.045
K_i	0.01
K_d	0.001

At this stage, follows a testing phase which requires:

1. Wi-Fi internet connection at the plant's location, connected to the plant through a USB adapter.
2. An external device, like a laptop, that is online and has a LabVIEW runtime version that is compatible with the one installed on the pilot plant's data acquisition module.
3. A link to the appropriate web server tool

The initial test phase involves the using of the provided by the system administrator for the web service tool to connect to the digital twin Figure 28.

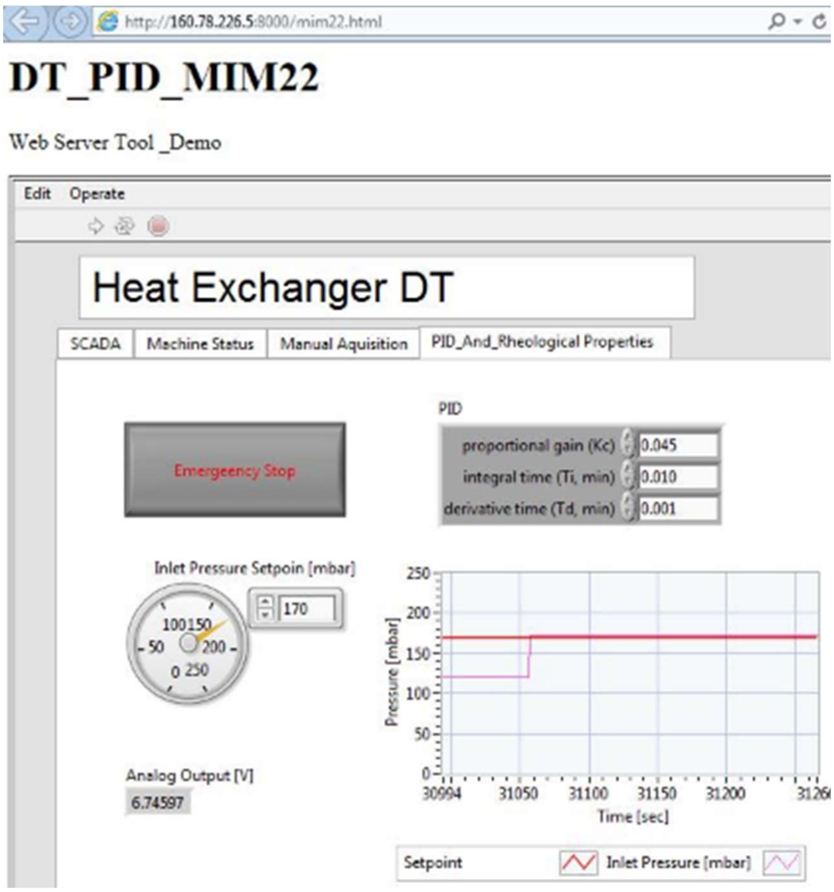


Figure 28 LabVIEW Web Server Tools

As an alternative, setting up a secure login to access the remote desktop. After successfully establishing the connection, users gain control over the system and can monitor its status though the digital twin. Using the remote control and monitoring web server tool, several tests were run to evaluate the PID controller's responsiveness, stability, and software robustness. The following steps were part of the test sequence:

Changing the inlet pressure setpoint from 0 mbar to 170 mbar to start the product pump. Once the system was operating at the desired point, a disturbance was simulated by temporarily changing the setpoint to a higher required inlet pressure of 200 mbar and afterwards changing it back to 170 mbar. As shown in Figure 29, the outcomes of the gain tuning process showed its effectiveness in achieving a quick response, slight oscillation, and negligible overshoot. However, due to the software restrictions present in conventional embedded systems when integrating new technologies, a few minor issues emerged during laboratory tests.

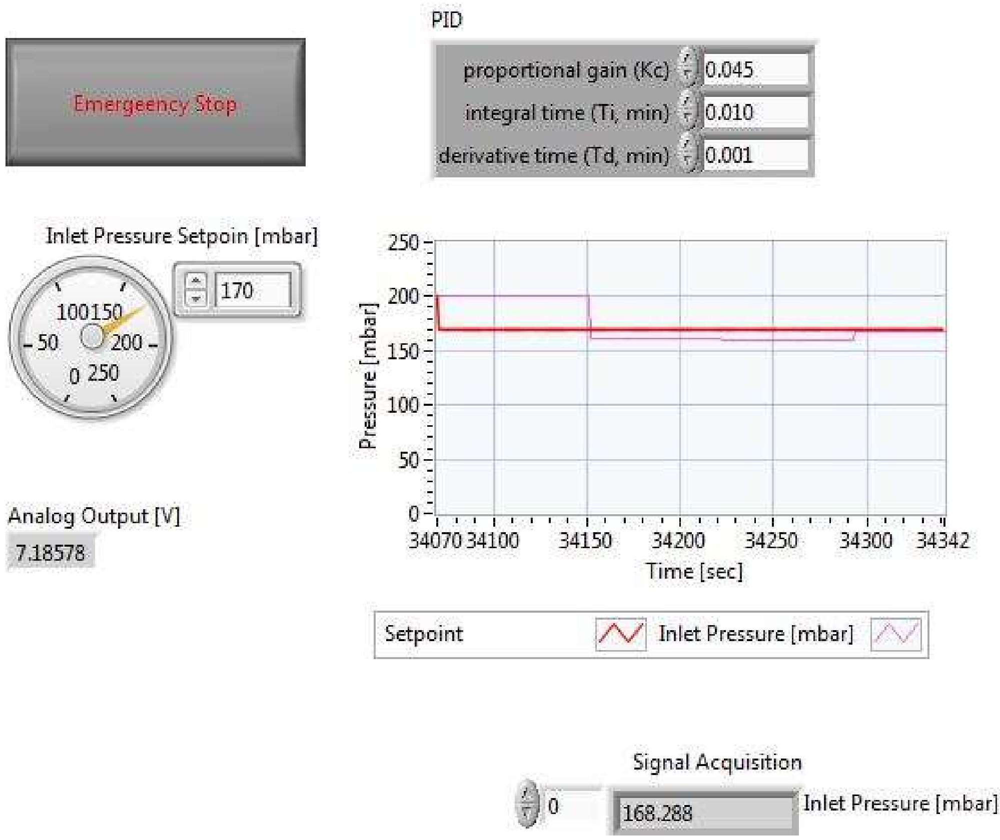


Figure 29 Response from 200 mbar to 170 mbar.

To ensure proper operation of the remote-control system, several prerequisites and settings are necessary:

1. a network connection at the plant.
2. a laptop with LabVIEW runtime for keeping track of the plant's condition.
3. the availability of a web server tool.

A Windows 7 Professional operating system, which has some restrictions on software upgrades, is embedded in the controller. Inadequate hard drive space for installing a modernized browser is one of these restrictions.

Users must use Internet Explorer or a different browser that is not based on Chromium to access the web server (such as Google Chrome, Microsoft Edge, or Opera). Chromium-based browser use may cause access problems. Alternatively, for users who cannot meet the browser requirements, external access to the digital twin can be achieved by utilizing the Microsoft Windows remote desktop application, which is, however, limited to Windows users, or by using a virtual machine capable of emulating a compatible Windows OS environment.

Based on the tests conducted in this study, it becomes evident that future research endeavors should focus on improving the tools available for connectivity. Specifically, enhancements in the LabVIEW environment's internet connectivity can be explored. Additionally, the implementation of an automated procedure for tuning the controller's gains could significantly enhance the system's effectiveness when deployed in real-world scenarios.

4.1.5 ML Test and results

4.1.5.1 Multiple Linear Regression

A multiple linear regression model is the first algorithm that has been developed. A statistical method known as multiple linear regression is used to simulate the relationship between a dependent variable (also known as the response) and several independent variables (also known as explanatory variables). The outlet pressure and product flow are represented, respectively, by the independent variables x_1 and x_2, which are used in this case specifically. The inlet pressure is the dependent variable, that we want to make predictions for it (y). Equation (9) presents the general linear model that explains how the dependent variable and the independent variables relate to one another.

$$y = a_1x_1 + a_2x_2 + intercept \tag{9}$$

The regression coefficients in the equation are denoted by the symbols a_1 and a_2, whereas the intercept is the value where y is equal to 0. Table 8 include the outcomes in obtained to the three fluids analyzed:

Table 8 Outcomes for fluids analyzed.

	a_1	a_2	Intercept	MAE	R^2
Water (fluid 1)	1.65	8.98	44.32	0.71	0.99
Fluid 2	1.27	18.29	57.28	5.745	0.947
Fluid 3	0.77	28.87	70.65	7.467	0.938

We calculated the mean absolute error (MAE) and coefficient of determination as performance metrics for evaluating the models' accuracy (R2). The MAE calculates the difference between two observations that represent the same phenomenon. MAE stands for the mean of the absolute errors when contrasting the actual values in the dataset with those predicted by the linear regression model. And R2 shows how much of the variation in the dependent variable can be predicted from the independent variables. With a maximum perfect fit score of R2 equal to 1, it measures how well the predicted values match the observed data. The findings in Table 8 unmistakably show that R2 validates the created ML algorithm for the case under investigation, as it consistently produces R2 scores above 93 percent for each fluid. Regarding MAE, it is notable that the algorithm performs better in the case of water compared to Fluid 2 and Fluid 3. The MAE for water is roughly 0.71 overall and peaks at 2.66 when the model determines the "warning" status. The effect of temperature on the rheological characteristics of non-water fluids, where additives are present and affect operating pressures, is likely responsible for the regression model's inaccurate prediction results when applied to different fluids. The difficulty may come from the definition of the "warning" status itself, which is expected to represent transitional phases or deviations (in a range between 10% and 25%) from typical plant parameters. This challenge may arise when focusing on the various performances in predicting the "warning" status. Notably, the MAE for evaluating system parameters in the "warning" status consistently stays below 10% for all tested fluids, indicating that the proposed approach performs with promise. Encouraged by these outcomes, we conducted an online test of the model using a specialized procedure. To simulate the three possible operational states of "ok," "warning," and "alert," the pilot plant was turned on and data were continuously collected from sensors for the three fluids. Utilizing data from the pilot plant, the ML model was used in real-time to evaluate its capacity to forecast the expected outlet pressure value and, if necessary, take actions within the plant. When used online, the regression model can produce results like those shown in Figure 30, for instance. This image shows how the online application of the regression model enables real-time calculation of the P1 value (ML-predicted P1), allowing for comparisons with the actual value (Real P1) and the estimate provided by the plant's DT model (DT P1). Analytical calculations are used to compare the P1 value that the ML model predicted with the actual one ("Data comparison" value).

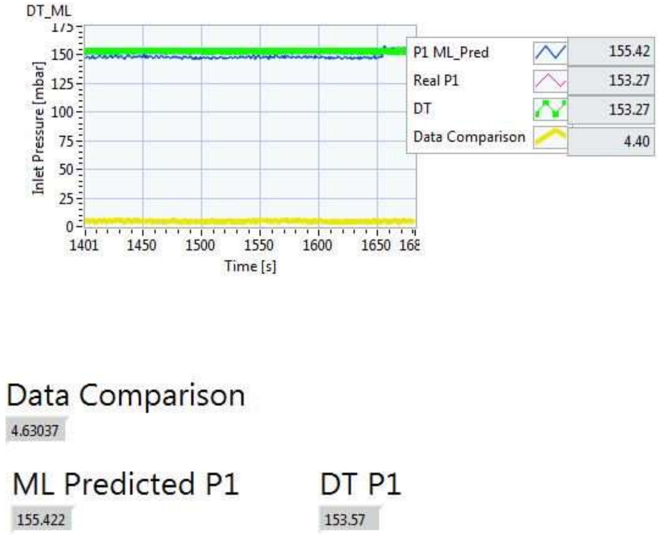
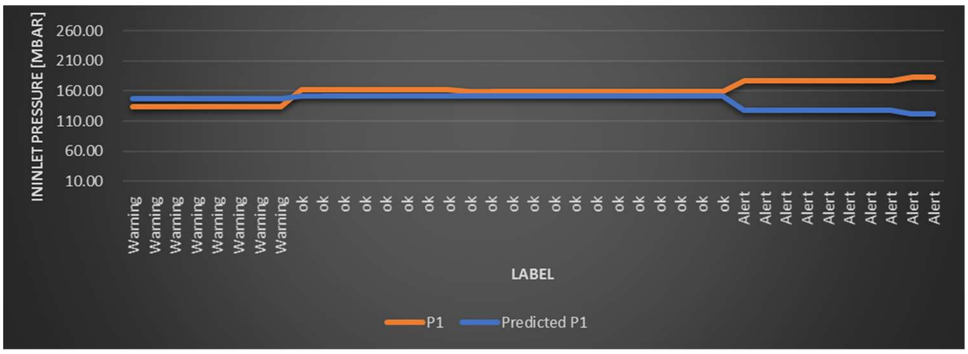
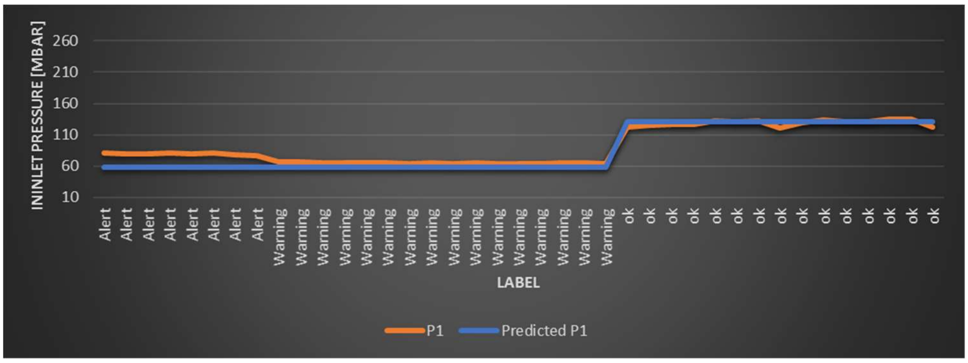


Figure 30 Front panel: online testing of the regressor algorithm for fluid 1

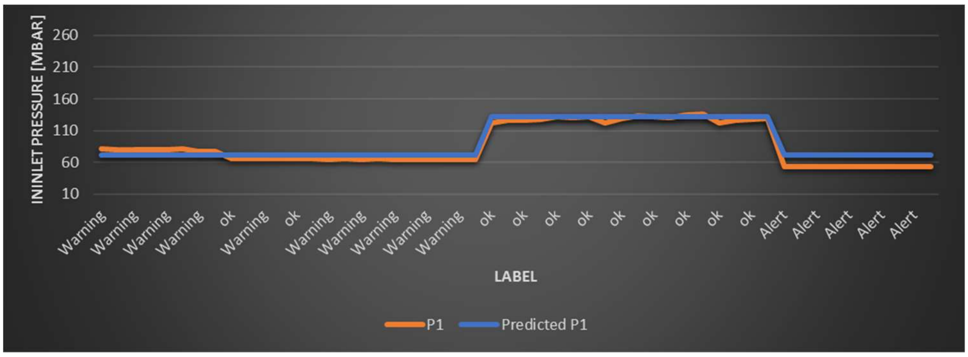
Figure 31 a–c displays the combined outcomes for the three fluids. These graphs, once more, contrast actual data obtained from the plant (P1 series) with predictions made by the ML algorithm (Predicted P1 series The x-axis labels show the machine's status, which reflects its operating circumstances as previously explained. This was contrasted with the outcomes obtained using the DT model. Figure 31 shows that, with errors of less than 5%, the algorithm correctly predicts the inlet pressure when the machine is operating in the "ok" condition. On the other hand, for the other statuses, the algorithm forecasts the anticipated inlet pressure in real-time using measurements of the process flow and outlet pressure. Since the machine is malfunctioning, it is only natural that the predicted value and observed value are different. Therefore, comparing the expected outlet pressure with the actual observed value makes it simple to spot anomalous operation. This result demonstrates how well the ML implementation within the system worked.



(a)



(b)



(c)

Figure 31 Results for multiple linear regression model: (**a**) Water; (**b**) Fluid 2; (**c**) Fluid 3.

4.1.5.2 Artificial Neural Network

MLPC stands for Multi-Layer Perceptron Classifier, the chosen artificial neural network (MLPC). The main goal of MLPC implementation is to offer a real-time classification of the machine's status based on the three parameters taken into consideration. Since there are three predefined categories—"ok," "warning," and "alert"—the output of this algorithm is a string that uses those to describe the machine's status. After preprocessing, the dataset used to train this model is the same as that described in Section 2.3. This includes the fourth column, where labels are given to each set of parameters to indicate the status of the machine. Technically speaking, the MLPC is represented in Figure 32 as two hidden layers, each with 200 nodes. The Adam optimizer, a stochastic gradient-descent method that iteratively updates network weights based on training data, was selected as the solver for weight optimization. A confusion matrix was used to validate this machine learning model. This matrix evaluates the algorithm's alignment with actual data to provide a performance evaluation. This means that it is possible to compute the crucial parameters recall and precision. Recall measures the proportion of correctly classified data points out of the entire set of actual data, while precision measures the ratio of correct predictions to all predictions for each data category. The ratio of correctly predicted samples to the total number of tested samples is the final way to calculate the model's accuracy. The confusion matrix and precision values for the dataset pertaining to water are shown in Figure 33 and Table 9. The numerical results show that the model has a 96 percent accuracy rate. Additionally, the weighted average precision value, which is calculated from the precision and recall values for each data class and weighted in accordance with the variety of instances in each class, has been used to evaluate the MLPC model. The score for this parameter for the case under consideration is 97 percent.

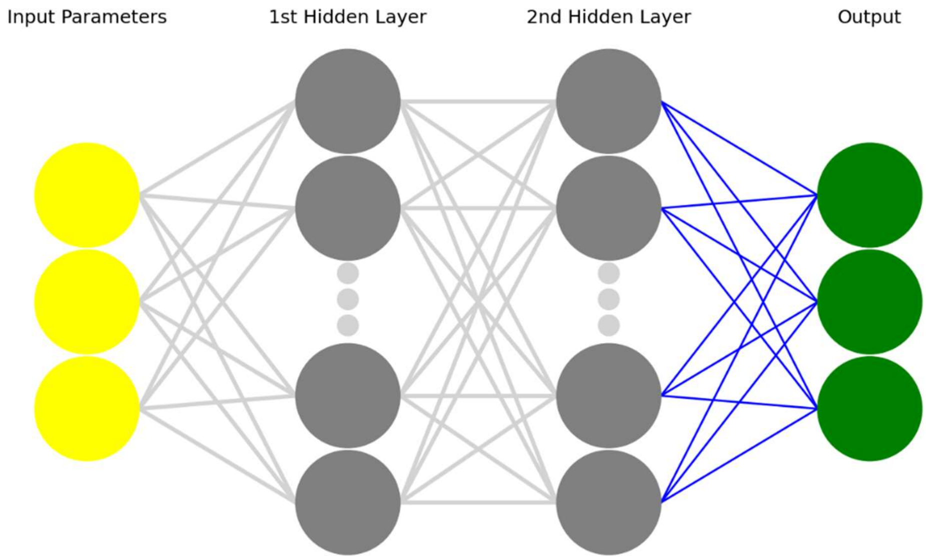


Figure 32 Neural Network structure

Table 9 Average precision values for water.

Category	Precision	Recall	Average Precision
Alert	1.00	1.00	0.83
Warning	0.92	0.27	0.05
Ok	0.74	0.99	0.09
Weighted average precision			0.97
Model Accuracy			0.96

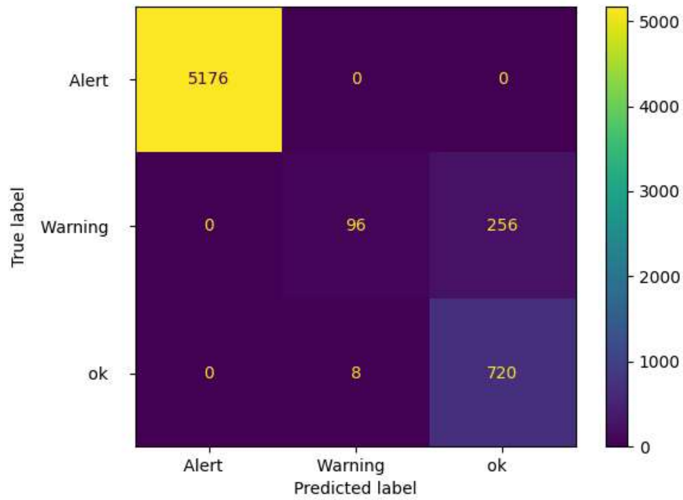


Figure 33 Confusion Matrix for Fluid 1

Table 9 shows that when used to determine the "ok" and "alert" statuses of the pilot plant, the MLPC algorithm exhibits excellent precision and recall. However, when attempting to predict the "warning" status, accuracy is noticeably reduced. Despite the two approaches' divergent logic, this agrees with the outcomes of the linear regression model. The lower predictive performance for the "warning" status for the ML algorithm under consideration may be explained by the relatively small range of values assigned to this status (i.e., a 10–25% absolute deviation from the normal condition), as opposed to the larger range of the "alert" status (i.e., a 25–100% deviation from the normal condition).This disparity affect the rate of "alert" data in the dataset, which may have an effect on how well the algorithm performs. In fact, the ML algorithm must be extremely precise to accurately detect such cases because the range of "warning" data is constrained. Regarding the earlier findings, the MLPC displays impressive accuracy, averaging 99 percent for Fluid 2 (Figure 34 and Table 10), and an accuracy of 93 percent for Fluid 3. (Figure 35 and Table 11). Due to the lack of data categorized in this category, the recall value score for Fluid 2's "warning" status is zero for the weighted average precision.

Table 10 Average precision values for Fluid 2.

Category	Precision	Recall	Average Precision
Alert	0.95	1.00	0.11
Warning	-	-	-
Ok	1.00	0.99	0.88
Weighted average precision			0.99
Model Accuracy			0.99

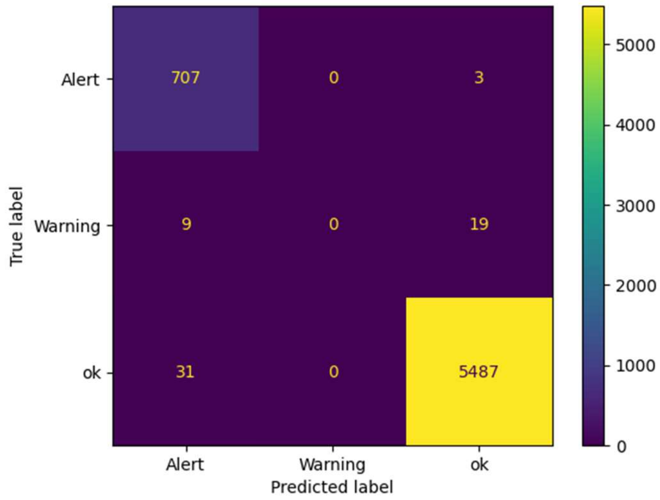


Figure 34 Confusion matrix for Fluid 2

Table 11 Average precision values for Fluid 3.

Category	Precision	Recall	Average Precision
Alert	0.98	0.94	0.29
Warning	0.93	0.23	0.06
Ok	0.92	1.00	0.59
Weighted average precision			0.94
Model Accuracy			0.93

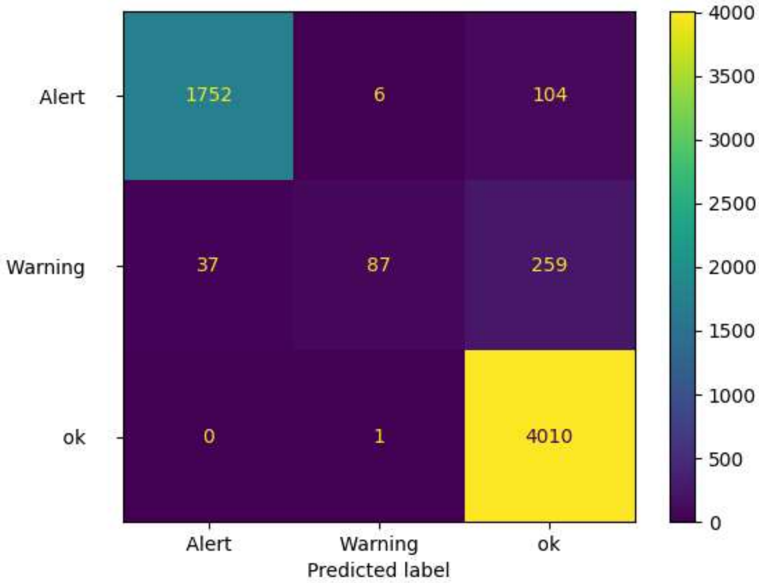
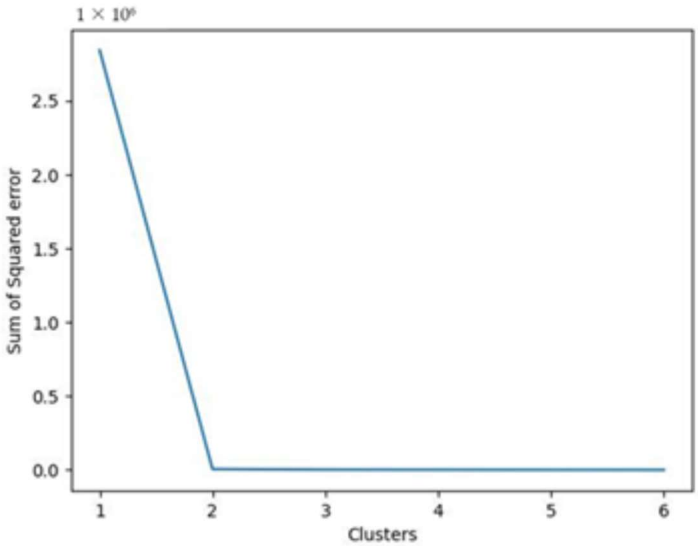


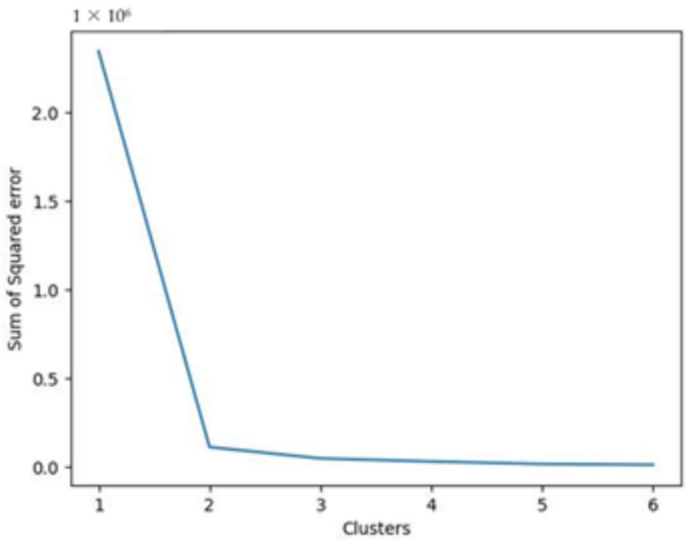
Figure 35 Confusion matrix for Fluid 3

4.1.5.3 K-Means Clustering

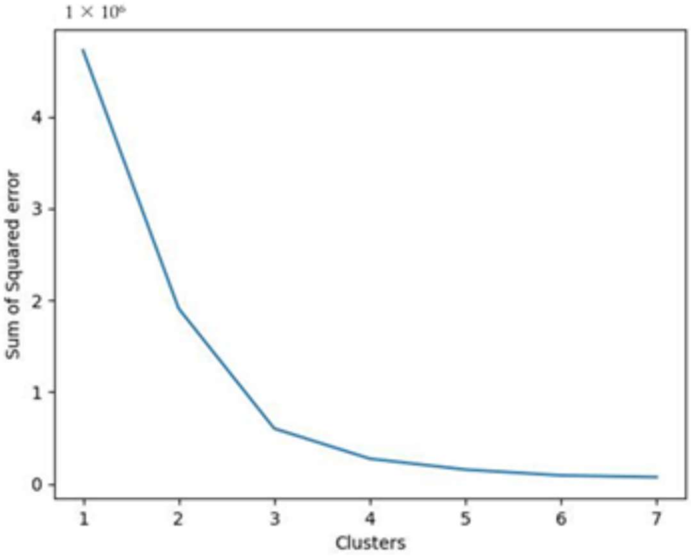
An unsupervised machine learning algorithm called clustering divides up a dataset into groups based on shared traits. A set of samples is grouped into distinct clusters by the k-means algorithm. Each cluster is identified by the mean value (centroid) of the components. Similarity between elements in k-means clustering is determined by how close they are to the centroid of their respective clusters. The dataset used to apply the clustering algorithm is made up of three columns that list the essential characteristics of the examined plant, like the algorithms previously described (i.e., P1, P2, and F). The expected result of this strategy is the classification of the gathered triplets of data into the proper statuses of "ok," "warning," or "alert," in line with the earlier discussion regarding the application of the clustering algorithm. This suggests that the algorithm is anticipated to divide the dataset's elements into three clusters in accordance with the operational statuses. The k-means procedure's number of clusters, however, is not predetermined. Finding the ideal number of clusters (k) for the dataset is the first step in putting this algorithm into practice. The "elbow" method was used to accomplish this. The elbow point on the curve represents the ideal number of clusters to use in this method, which involves plotting the explained variation as a function of the number of clusters chosen. As a result, with the number of clusters ranging from 1 to 6, the sum-of-squares error (SSE) between the data points and their assigned cluster centroids was calculated (in steps of 1). Figure 36-a- c show the elbow curve calculated for the dataset (Water, Fluid 2, and Fluid 3, respectively). The findings suggest that three clusters would be a reasonable number for water and Fluid 2, as this number offers enough stability to explain the variance. The same does not hold true for Fluid 3, where a greater number of clusters (six) are required to achieve stability in SSE. However, the number of clusters was set to three for all fluids to ensure the suitability of using the clustering algorithm in the examined system. The k-means algorithm was used to categorize the dataset's elements with this many clusters to test its ability to determine the system's "ok," "warning," and "alert" statuses. Figure 37 a–c shows a 3D plot of the fluid clusters for the three fluids. The algorithm also saved the classification in a new database column with the name "cluster," which was filled with the values expected by the k-means model.



(a)

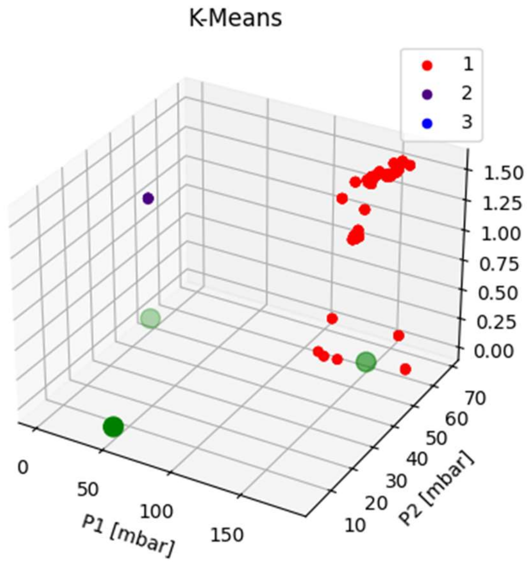


(b)

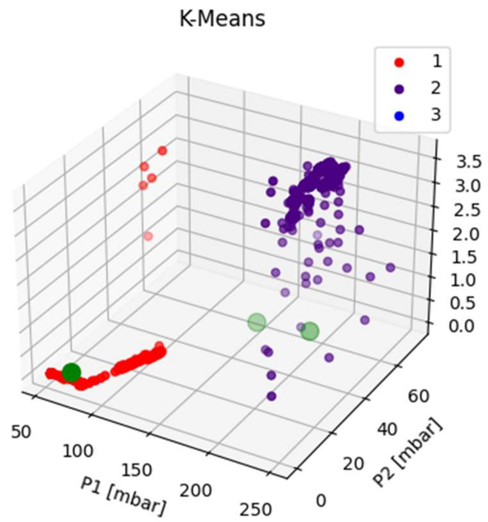


(c)

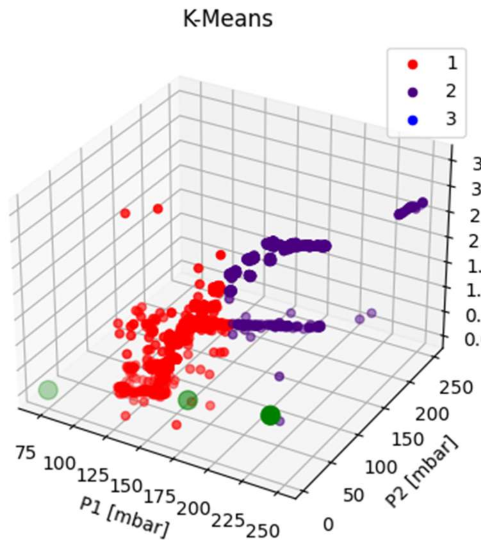
Figure 36 Elbow Grpah for the dataset of the tested fluids



(a)



(b)



(c)

Figure 37 k-means 3D-plot: Fluid 1 (a), Fluid 2 (b), and Fluid 3 (c).

The accuracy of the k-means method was evaluated using the silhouette score as a performance metric. A clustering technique's accuracy can be assessed using this method, which provides a succinct representation of how well each element has been classified. The range of the score is-1 to 1, where:

- 1 representing clearly separated clusters.
- A score of 0 denotes that there is no statistically significant difference between the clusters.
- Clusters with a score of-1 have been incorrectly assigned.

The models produced a silhouette score for water of 0.92, fluid 2 of 0.91, and fluid 3 of 0.76. These results indicate that the algorithm can classify the data correctly even if the number of clusters is forced to be three. The clustering algorithm would not be able to categorize the clusters as "ok," "warning," or "alert," so it is crucial to remember that well-defined clusters do not necessarily imply that they accurately represent the potential machine statuses. A comparison of the original machine status with the three clusters obtained after classification was done to evaluate this aspect. Tables 9–11 present the findings.

According to Table 12, the clusters for the water dataset that the k-means algorithm found can be summed up as follows:

Clusters 1 and 3 are made up solely of "alert" data, each of which has unique properties. While cluster 1 "alert" situations are characterized by little to no product flow, cluster 3 "alert" situations involve triplets of data where the inlet pressure is always zero. In either case, the algorithm appears to have correctly identified these "alert" circumstances. Cluster 2 consists of a mix of "ok" and "warning" conditions, with a small number of "alert" circumstances remaining. This suggests that, like the other ML algorithms put to the test, the clustering algorithm has trouble telling the difference between "ok" and "warning" circumstances, which again points to a problem with identifying the "warning" status. Like this, the clustering of Fluid 2 (Table 13) shows that one cluster (i.e., cluster 2) is clearly defined and only includes typical plant operations. Clusters 1 and 3 include a variety of potential circumstances. Given that most of the data relate to the "ok" status, Cluster 3 is arguably acceptable in practice. However, cluster 1's classification is not accurate enough for practical use because 24 percent of the anomalous circumstances are categorized as typical working circumstances, which could endanger the safety of those using the machine. Regarding Fluid 3 (Table 14), the clusters seem to accurately reflect many of the actual working conditions of the plant, even though the algorithm was forced to use three clusters rather than the ideal number of six:

- Cluster 1: only contains "alert" circumstances.
- Cluster 2: only contains "ok" functioning.
- -Cluster 3 consists primarily of "warning" situations.

This implies that the algorithm correctly identified and categorizes abnormal machine operation. When examining cluster 2, it becomes clear that it contains all "ok" statuses with only minor amounts of warning and alert circumstances. Once more, it appears that the "warning" status has been largely assimilated into the "ok" or "alert" circumstances.

Table 12 Cluster composition vs. machine status for water

	Cluster Composition		
Machine status	1	2	3
OK	-	58%	-
Warning	-	28%	-
Alert	100%	14%	100.0%

Table 13 Cluster composition vs. machine status for Fluid 2.

	Cluster Composition		
Machine status	1	2	3
OK	76%	100%	87%
Warning	-	-	1%
Alert	24%	-	12%

Table 14 Cluster composition vs. machine status for Fluid 3.

	Cluster Composition		
Machine status	1	2	3
OK	0%	83%	0%
Warning	-	7%	1%
Alert	100%	10%	98%

To develop a comprehensive tool for identifying anomalies in the operation of an industrial system, this study has proposed an application that aims to integrate digital twin models, machine-learning algorithms, and Industry 4.0 technologies. The suggested method has been developed for use in a tube-in-tube indirect pasteurization machine for fluid food. Four distinct operating modes were created and put into use in the digital twin of the plant to get the best results possible with the technologies at hand. To ascertain which operating mode most closely matches the system's actual condition, several tests were carried out. The system can be monitored and controlled using the digital twin environment, which is furnished with tools created in earlier studies, both locally and over remote connections. In contrast to fully automated systems, the requirement for manual controller setpoint and fluid characteristic adjustment in the software can be a drawback. Three machine learning techniques (clustering, linear regression, and artificial neural networks) were incorporated into the created online plant monitoring solution to address this issue. The system can be monitored and controlled in-person and over remote connections using the digital twin environment, which is furnished with tools created in earlier studies. However, compared to fully automated systems, the need for manual adjustment of the controller setpoint and fluid characteristics in the software can be a drawback. The developed solution for online plant monitoring included three machine learning techniques (clustering, linear regression, and artificial neural network) to address this problem. The multi-layer perceptron classifier (MLPC) algorithm demonstrated high accuracy in predicting anomalies for categorizing "ok" or "alert" statuses of various fluids tested but had less accuracy in categorizing the "warning" status. Similar conclusions were reached for the K-means clustering algorithm, which effectively grouped "alert" and "ok" statuses but frequently conflated the "warning" status with regular plant operation. This can be attributed to the "warning" status's definition, which covers events that fall into a relatively small range of values and makes accurate detection difficult. Currently, only "ok" situations, which are usually correctly detected, may be identified using clustering and classification algorithms. To facilitate employee intervention at the plant, any deviation from these operating conditions would result in an alert being displayed on the HMI. However, these algorithms still need to be improved and refined before they can be used online.

4.2 Bag Filter pilot plant

A plant with a bag filter is a system employed for air purification from suspended particles, dust, and other impurities. For the case under examination the plant is composed by 31 bags, a main inlet pipe i.e., defined as manifold, a compressor, an outlet pipe which exhaust the cleaned air in the atmosphere and a branch connected to the manifold with collect he polluted air containing dust to be separated within the cyclone filter.

4.2.1 Plant Description

4.2.1.1 Main Piping

At the beginning of the plant, there is a main inlet piping through which contaminated air enters the system. This piping is designed to direct polluted air into the filter's chamber.

4.2.1.2 Compressor

The compressor is a critical component of the plant. Its primary function is to increase the air pressure so that it can effectively pass through the bag filter. This is particularly important to ensure that the air is evenly distributed among all the bags of the filter.

4.2.1.3 Bag Filter

This is the core of the plant. The bag filter (Figure 38) consists of 31 bags (long, cylindrical tubes) aligned inside a sturdy structure. These bags are made of special materials designed to capture suspended particles in the air as it passes through them. The bag filter retains the particles and allows clean air to pass through the bag walls towards the outlet.

Figure 38 Bag Filter Pilot Plant

4.2.1.4 Clean Air Outlet Piping:

The air that has been purified through the bag filter exits the plant through a separate piping, ensuring that clean air is properly directed out of the system. This clean air can be released into the environment or used for other applications depending on the user's needs.

Branch with Cyclone Filter: which collect the polluted air form the shop floor into the main piping to be separated within the cyclone filter. The cyclone filter is a device used to separate solid particles, such as flour, from the air stream. It operates by creating a vortex inside a cylinder, causing heavier particles like flour to be pushed towards the walls and collected, while clean air continues its path.

4.2.1.5 Valve

For the regulation of the system flow, there is a butterfly valve positioned downstream of the filter and preceding the fan, determining its intake section. In addition to this component, it is possible to adjust the fan's frequency using an inverter. The Endress Hauser Deltabar S PMD75 (Figure 39) differential pressure sensor, positioned at the ends of the filter, measures the occurring pressure losses.

Figure 39 Endress Hauser Deltabar

4.2.1.6 Stepper motor

The butterfly valve mounted on the central section of the manifold is coupled with a stepper motor which shaft is connected to the rotor of the motor. Stepper motor (Figure 40) is electromechanical device widely used in precision control applications, characterized by its ability to divide a full rotation into a series of discrete steps. It operates based on the principle of converting electrical pulses into mechanical motion, with each pulse representing one step. Stepper motors feature a rotor with teeth and a stator with coils, typically arranged in a bipolar or unipolar configuration. The motor's movement is controlled by energizing specific coils in a sequenced manner, inducing magnetic attraction and repulsion between the rotor and stator. This sequential energization enables precise angular positioning, making stepper motors suitable for applications requiring accurate and repeatable positioning, such as 3D printers, CNC machines, robotics, and automation systems. Stepper motors exhibit distinct advantages, including simplicity, open-loop control capability, and immunity to feedback issues. However, they may exhibit limited speed and torque compared to other motor types, necessitating careful consideration of application requirements.

Figure 40 Stepper Motor

4.2.1.7 Sensors

The pilot plant has been equipped with sensors able to acquire data and monitor the main parameters of the system which are: (i) Air flow; (ii) Air velocity; (iii) pressure drop measured at the inlet and outlet section of the filtering chamber. The component dedicated to performing these measures are basically two types of sensors, in particular: a differential pressure transmitter called KIMO, and an anemometer.

Differential pressure transmitter

The Kimo, (Figure 41)is a particular kind of sensor, known as a "density-compensated differential pressure flow sensor". This sensor measures air velocity based on differential pressure measurement while considering the air. This type of sensor is used in environmental monitoring, HVAC (Heating, Ventilation, and Air Conditioning), and other applications where precise measurement of air velocity is required. The principle of operation of this sensor is based on Bernoulli's law, which describes the relationship between fluid velocity, pressure, and fluid density in a steady flow. The sensor consists of two tubes or channels, one of which is exposed to the air flow to measure static pressure, while the other is equipped with a differential pressure sensor to measure the difference between dynamic and static pressure. This pressure difference is directly related to air velocity. To account for air density, the sensor uses a compensation method. Since air density can vary with temperature and atmospheric pressure, it is important to consider this variation to obtain accurate air velocity measurements. Density compensation typically involves using a temperature sensor to measure air temperature and then correcting the pressure difference based on this temperature, using the ideal gas law.

Figure 41 Kimo

While for the measure of the air velocity at the outlet section of the filtering chamber it has been installed a hot-wire anemometer. This device is used to measure air velocity or its direction within a flow. It is a type of thermal sensor that exploits temperature variations caused by air velocity to calculate it.

Hot-wire anemometer

The operation of a hot-wire anemometer, (Figure 42)is based on the principle that a thin, heated wire exposed to the air will experience faster cooling due to the moving air. The wire is kept at a constant temperature using electrical resistance. When air flows over the wire, the wire's temperature decreases in proportion to the air velocity. The amount of cooling of the wire is directly related to the surrounding air's velocity. A temperature sensor continuously measures the wire's temperature, and based on this temperature variation, the device can calculate the air velocity. The higher the air velocity, the greater the temperature change of the hot wire.

Figure 42 Hot Wire anemometer

4.2.2 Digital model

4.2.2.1 Plant characteristics

The purity level achieved in the exhaust air atmosphere is involvedly linked to the fluid velocity in the filtration process. Given this premise, the implementation of an advanced control strategy becomes imperative to guarantee both product quality and operator safety. To regulate the velocity of contaminated air within the manifold and to provide real-time monitoring of process parameters, a specialized software application has been developed. This application incorporates a Data Transmission model constructed using LabView, and its overall structure closely resembles that previously described for the pasteurization facility in the context of liquid food production. The critical parameters for this facility include air velocity and piping pressure. These parameters are monitored through dedicated sensors integrated into the plant. Specifically, air velocity is measured using a hot wire anemometer, while the air pressure within the pipe sections is measured using a differential pressure transmitter. Each branch of the pilot plant is equipped with a hot wire anemometer capable of measuring air velocities of up to 30 meters per second and providing an analog output within the standard current loop range of 4-20 mA.

The differential pressure transmitter calculates air flow based on the Bernoulli equation (10), expressed as:

$$Air\ Velocity\ =\ C_m\sqrt{(2\Delta P/\rho)} \qquad (10)$$

Here, C_m represents the differential pressure device coefficient, which, in the case we are examining, is determined to be 0.84 (Kimo Instruments 2022). This device is positioned at the center of the manifold and provides an analog output signal in the standard current loop format (marked as KIMO dP in Figure 43).

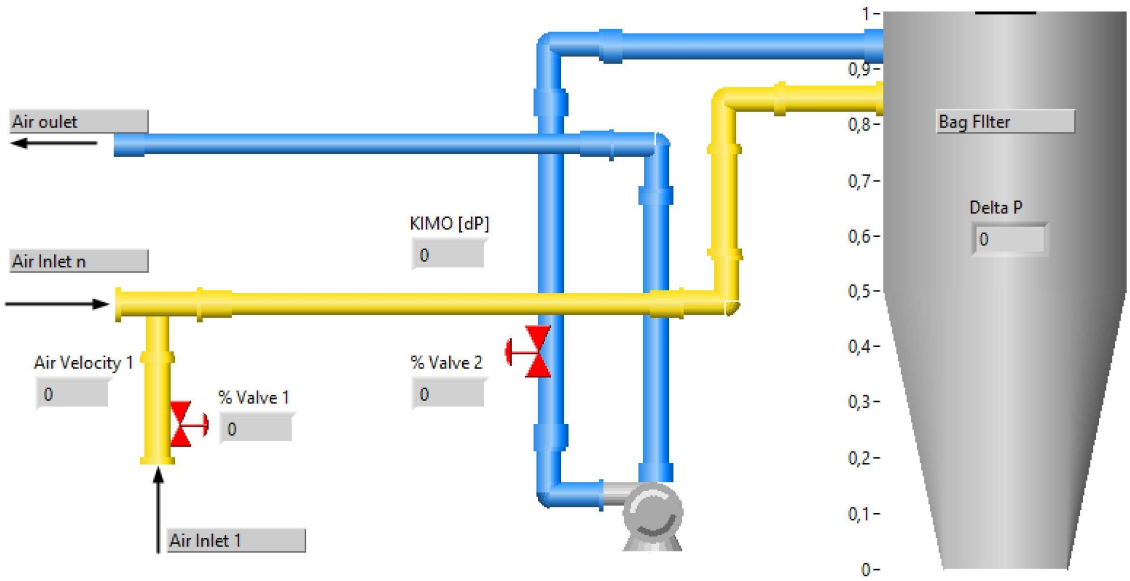


Figure 43 Bag Filter Front Panel

Valves 1 and 2 are actuated by stepper motors, specifically brushless synchronous DC motors with the model M60SH86-TO0512P24C. These motors utilize digital input and output signals for starting and stopping. Two of these devices are installed on the pilot plant, enabling adjustment of the piping section for the air outlet manifold and air inlet 1 (as shown in Figure 43).

Controlling the stepper motor requires a dedicated drive, which serves a dual purpose: it initiates the control signal and supplies power to the motor coils while respecting the defined sequence in the first block. The input interface of the drive communicates the desired command to the motor. The commercial drive employed in this system is the SMD1104LIE.

A digital layer of the system has been developed, encompassing the Human-Machine Interface (HMI), system monitoring, control block diagram, and a virtual model of the machines and sensors specific to this case. The system model follows an experimental approach. More precisely, the plant characteristics have been determined by assuming that the air inlet velocity (AIV) varies with the valve position (VP). The characteristic curve was constructed by collecting various AIV data within a sampling time of 540 seconds (with a sampling rate of 1 Hz). These data were correlated with the valve opening angle, ranging from 0° (fully open) to 80° (almost closed) in 10° steps, sampled every 60 seconds. The resulting curve, (Figure 44) from the experimental data, (

$$AIV \;=\; 0.0035{V_p}^2 \;+\; 0.0502V_p \;+\; 18.553 \qquad (11)$$

Table 15) can be summarized using the following empirical relationship, equation (11):

$$AIV \;=\; 0.0035{V_p}^2 \;+\; 0.0502V_p \;+\; 18.553 \qquad (11)$$

Table 15 Empirical relationship Valve Position-AIV

Valve position [°]	AIV [m/s]
0	18.35
5	18.832
10	18.792
15	18.582
20	18.202
25	17.652
30	16.932
35	16.042
40	14.982
45	13.752
50	12.352
55	10.782
60	9.042
65	7.132
70	5.052
75	2.802
80	0.382

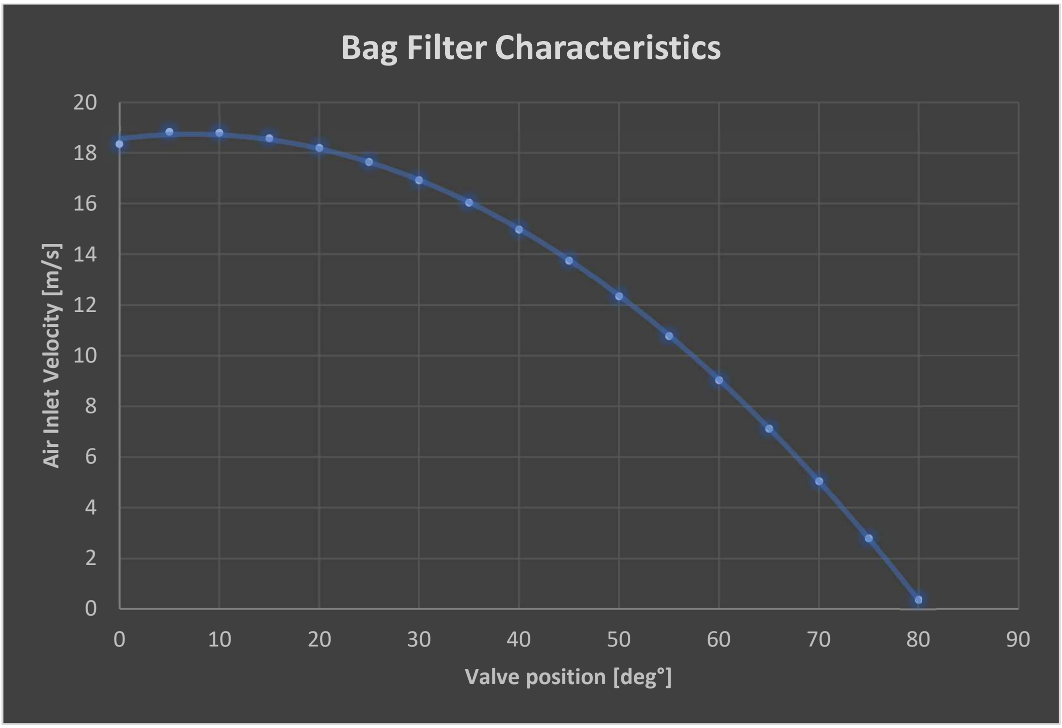


Figure 44 Bag Filetr operating points

4.2.2.2 Data communication system

While for the previous case study it has been adopted a TCP/IP protocol, for the case under examination, the choose falls into Modbus. This protocol is a widely used in industrial automation and control systems. It facilitates communication between various devices, such as sensors, actuators, and controllers, allowing them to exchange data and control signals. In this case, it has been programmed to communicate with a stepper motor to adjust a valve position which adjust air flow into the manifold. Nonetheless, Modbus performs as a standardized way for a supervisory controller (such as a PLC or a computer) to communicate with the stepper motor driver/controller. It defines the rules and formats for data communication, including commands to set the desired valve position and queries to read the current valve position or other relevant parameters. The supervisory controller sends commands to the stepper motor controller, specifying the desired valve position or control parameters. These commands are typically sent in a structured format, such as MODBUS function codes, which the stepper motor controller understands. The code for this protocol has been implemented into LabVIEW environment considering the following steps:

I. The establishment of a connection between DAQ (Data Acquisition) and the drive necessitates the development of a master-slave configuration compliant with the Modbus protocol. The initial step involves creating Modbus library support to facilitate communication between the software and hardware components. This protocol mandates a cyclic query-response mechanism between the host and the connected devices over the same network. To establish a proper connection, a server is configured to communicate via serial ports with the hardware devices, referred to as "slaves." In this setup, the LabVIEW controller acts as the server, while the motor drives serve as the slaves.

II. The server configuration details, involve the use of IP addresses to represent the motor drive names, with assigned values (e.g., 125) as the starting addresses for Modbus slave registers. A refresh rate of 0.1 seconds is allocated for each call-response cycle. In the event of connection errors, the master initiates a retry mechanism with up to four attempts to restore the correct linkage. Once the master setup is complete, users can communicate with the slave registers by reading the portion of the internal Modbus register of the client that contains specific commands or actions. The server can both read and write the holding register of the drive, which is in the fourth memory map of the register. For the specific case being examined, drive parameters and associated addresses required for control tasks are listed in Table 16, addressed at 400001. Notably, the third variable is a 16-bit unsigned word for setting the desired motor position via software and is addressed at 400060. The fourth variable, known as "Target SD," is a 32-bit integer data addressed at SD400009 of the holding register, used to specify the desired motor position, as presented in Table 1. The final variable pertains to the command for shaft velocity, which ranges from -10000 to +10000, with the sign indicating the rotation direction (positive for clockwise and negative for counterclockwise rotation). Each command or position corresponds to a software variable, organized within a library as sub-VIs (Virtual Instruments) in the digital layer. Similar precautions apply to the "TP1" and "TP" variables. To command a target setpoint, bit 7 of register 400009 can be written by entering 27 concatenated with the address.

Table 16 Drive parameters and register addresses

ID	Variable Name	Data Type	AccessType	Register Address
CM1	CurrentMotor1	Boolean	read only	400200.1
ShP	Shaft Position	Int32	read/write	400001
RS	RotateShaft1	UInt16	read/write	400060
TP	TargetPosition	Int32	read/write	SD400009
ShV	Shaft Velocity	Int32	read/write	SD400064

4.2.2.3 Process Control strategy

The control approach in this case follows an open loop. Further steps in the control implementation process involve the master adhering to a precise algorithm to communicate effectively with each variable. Due to the multiple query-response cycles, the master must query the slave with specific time delays for each call response to prevent data communication overlap, leading to the development of a timed sequence block diagram. The delay between one query response and the next is set at 0.5 milliseconds. Initially, the master activates the drive by querying the "RS" (RotateShaft1) address, located in the fourth section of the Modbus memory map, at the second bit. This entails writing an unsigned word of 2 into the register address. Subsequently, the master checks the status of the motor by querying the "CM1" (CurrentMotor1) address. The "ShP" (Shaft Position) variable can be read or written depending on the task at hand. During reading, the master queries the address and returns the motor position as a decimal value in the range of 0 to 10800, representing the current position of the motor shaft in terms of clockwise rotation, with a proportional relation of 30 between a full turn of the shaft and the round corner. To display the value appropriately, an adjustment is required by dividing the read word by 30. The control system is composed of both hardware and software component. In terms of the hardware, the DAQ module, which serves as the system's brain and is connected to the sensors and motor through various cables, has been used to read the analog output of the probes, which is offered in a standard current loop (4–20 mA). Instead, the motor drive is communicated with using a twisted pair ethernet cable. The user monitor shows the status of the process parameter as well as the behavior of the machine. Again, using LabVIEW, the software tool supporting the system was created in G-Code. An open-loop control for the stepper motors has been developed in accordance with prior applications to flour milling plants, considering the peculiarities of the equipment, particularly its low sensitivity to external factors. Twisted pair cable was used to establish the connection between the RS232 port on the motor drive and the corresponding one on the DAQ. The Modbus RTU communication protocol is adopted by the communication layer. This is also generally used as a bridge between the field devices and the control systems and is quickly emerging as a go-to method for remote monitoring on the Internet of Things. It is necessary to create a master-slave connection that complies with Modbus to connect the DAQ to the drive. To connect LabVIEW software and hardware components, the first step is to develop support for the Modbus library. Such a protocol necessitates a cycle of query and response between hosts connected via the same web. A master must be set up to communicate with the host through serial ports to create the proper connection (slave). The LabVIEW behavior as the master, and the motor drives serve as the slaves. Figure 45displays the server configuration, where the IP address stands in for the motor drive name and its value (125), respectively, for the starting address of the Modbus slave registers. The duration of each cycle of a call response is 0.1 seconds. The master may query the slave up to four times to re-establish the linkage if a connection error occurs. By reading the section of the client's internal Modbus register where specific commands or actions are located, the user can communicate with the slave registers after the master has been set. The holding registers of the drive, which is in the fourth memory map of the register, is accessible to the master for reading and writing.

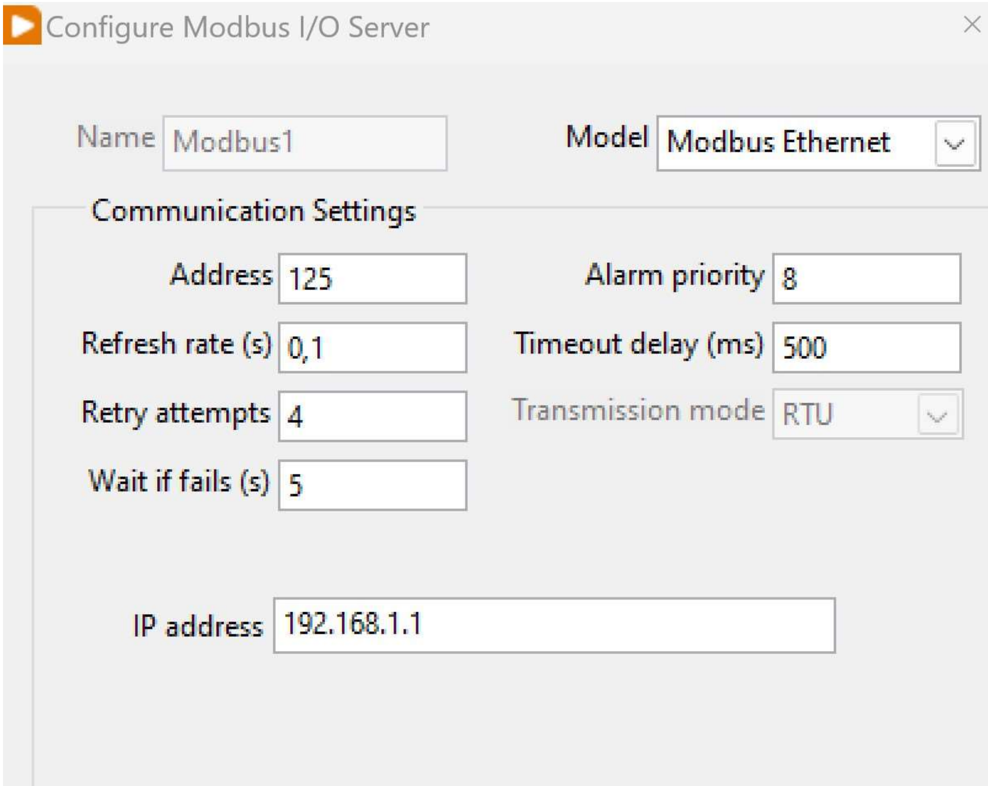


Figure 45 Server configuration

The control is supported by a code section that, as seen in Figure 46, can minimize the overshooting of the controlled parameter. This code is made up of several "if" cycles with two parameters called manual and AIV creating a hierarchy. The first box is based on the former parameter's Boolean (true/false) value, which enables manual control of the system when set to true or open-loop control when set to false (when set at false). There are in fact two subcases controlled by the air inlet velocity parameters, and they can be distinguished by comparing the AIV and the desired setpoint (SP), which has been modified to account for plant characteristics. The following inputs must be set in the motor drive because the controller must slightly adjust the valve position to reach the setpoint, specifically if AIV is not in the range SP1 m/s (Figure 48).

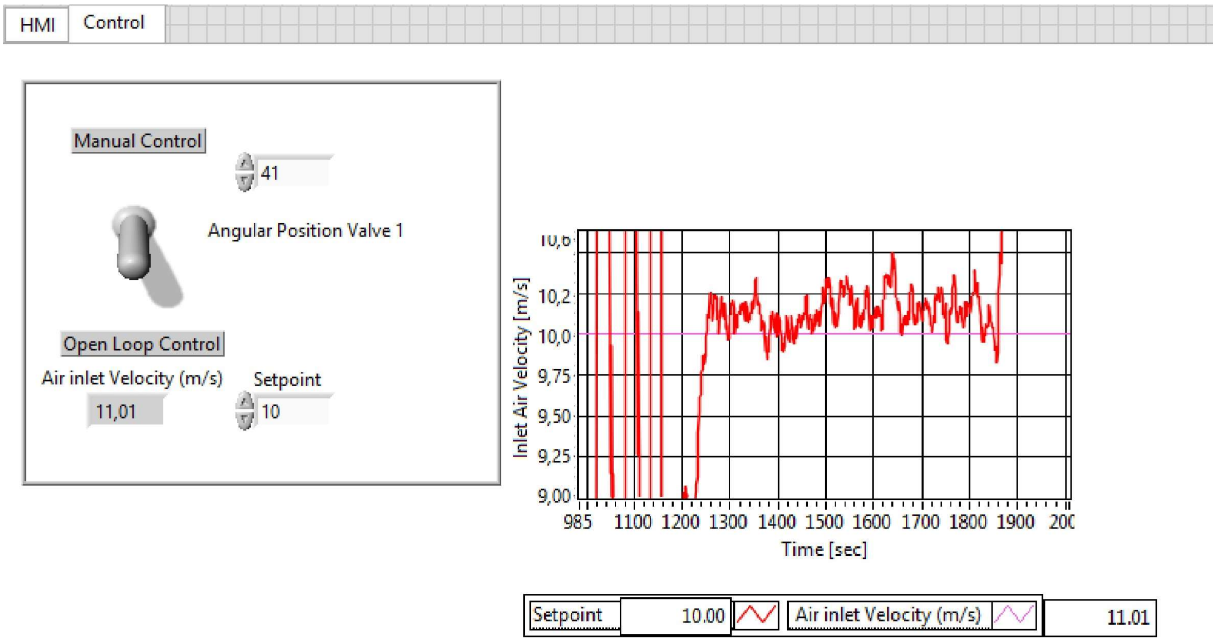


Figure 46 Bag Filter Front Panel Control

i. The motor shaft rotation velocity.

ii. The target task.

iii. The command to achieve the setpoint.

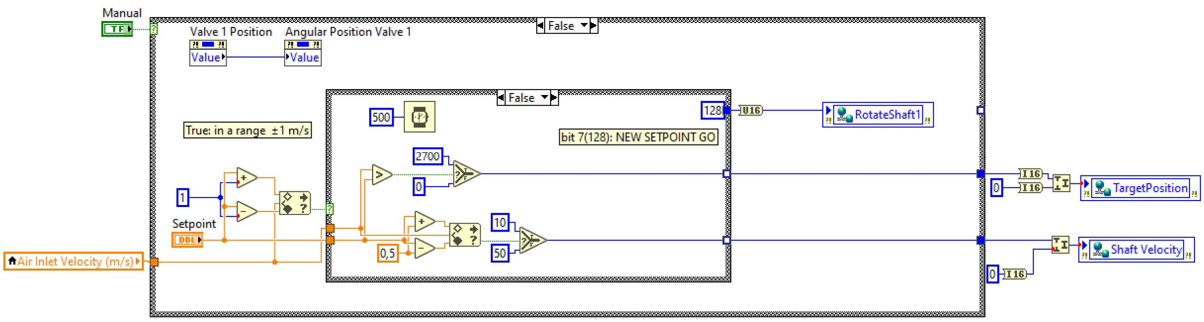


Figure 47 Block Diagram: Setpoint achievieng

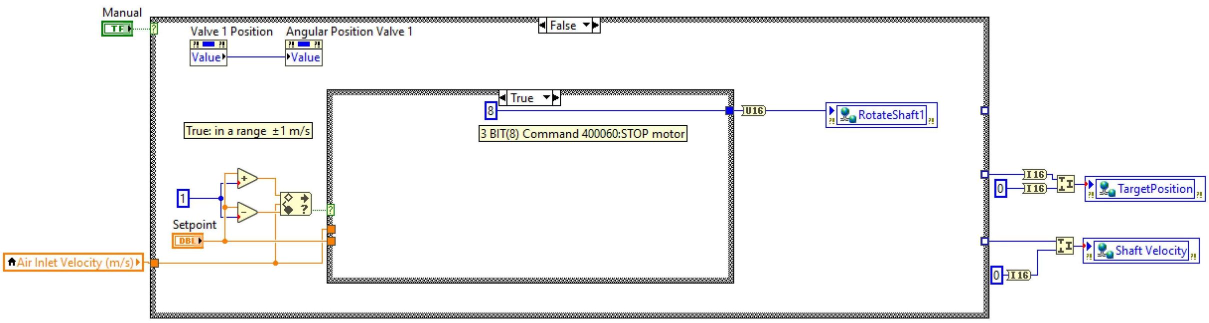


Figure 48 Block Diagram Motro Stop in place

The motor tends to reach the desired setpoint by checking the 7th bit of the Rotate Shaft (RS) register variable up until the target task is in the range between 0-2700, which represents the angular position of the valve from 0° to 90°. The shaft velocity is also controlled according to a further comparison between the AIV and a constant value of 0.5 m/s, which allows for a higher velocity when the AIV is far from the desired value and a slight slowdown when it is getting close to the setpoint. By querying the third bit of the RS once the setpoint has been reached, the valve is stopped in place Figure 47. By querying the third bit of the RS once the setpoint has been reached, the valve is stopped in place (Figure 48).

4.2.3 Machine learning algorithms

Before developing a specific predictive model based on the collected data, it was necessary to study the input parameters and evaluate their relationships.

4.2.3.1 Multivariable Linear regression

Multivariable Linear regression is a statistical method used to study and model the relationship between a continuous dependent variable and one or more independent variables. The dependent variable, also known as the "response variable" or endogenous variable, is the variable that is predicted or explained by the model. It is the parameter you want to analyze or predict based on the values of the independent or predictive variables. In the formulation of a linear regression model, the dependent variable is represented as a linear combination of the independent variables, multiplied by their respective regression coefficients. The goal of linear regression is to find the best estimate of the coefficients that minimizes the discrepancy between the observed values of the dependent variable and the values predicted by the model. The choice of the dependent variable depends on the analysis's objective and specific research questions. It can be a continuous variable, such as income or temperature, or a discrete variable, such as the number of children or the result of an exam (pass or fail). The independent variables, also called "predictive variables" or exogenous variables, are the variables used to explain or predict the variation in the dependent variable in the model. These represent the characteristics or conditions believed to influence the dependent variable. In the formulation of the linear regression model, independent variables are used to estimate the regression coefficients, which represent the effect of the independent variables on the dependent variable. Independent variables can be selected based on theoretical hypotheses, previous studies, or empirical considerations. It is essential to choose them carefully to ensure their relevance to the dependent variable and their ability to provide adequate explanation or prediction. In linear regression, the equation (12) takes the following form:

$$Y = \beta 0 + \beta 1X"1" + \beta 2X2 + \cdots + \beta nXn + \varepsilon \qquad (12)$$

Where:

- Y represents the dependent variable (response variable) that you want to predict or explain.

- X1, X2, ..., Xn are the independent variables (predictive variables) used to explain the variation in Y.

- β0, β1, β2, ..., βn are the regression coefficients, representing the effect of the independent variables on the dependent variable. β0 is the intercept term, while β1, β2, ..., βn are the coefficients associated with each independent variable.

- ε represents the residual error, which is the discrepancy between the observed values of the dependent variable and the values predicted by the model. It is due to unexplained or random factors.

The goal of linear regression is to estimate the regression coefficients (β0, β1, β2, ..., βn) in a way that the linear equation improves the accuracy of predicting the dependent variable Y based on the values of the independent variables (X1, X2, ..., Xn). In the case at hand, multiple regression is considered the most suitable for the study. It allows for examining the simultaneous effect of multiple independent variables on the dependent variable, considering their individual effects and possible interactions. Furthermore, it can provide a better prediction of the dependent variable compared to simple linear regression because it considers multiple factors that could influence the dependent variable. This situation clearly reflects the simulation discussed in the study, as it involves parameters such as speed and pressure, and aims to evaluate their impact on filter pressure drop. Specifically, it involves real-time analysis of the efficiency rate of zinc and lead extraction concerning selected parameters. The variables include temperature, pH, and particle size, received from the IoT module connected to corresponding sensors. Through data analysis using supervised machine learning algorithms, a multivariate regression model is formulated to predict future estimates related to zinc and lead extraction. In general, the assumed variables have predicted the efficiency of zinc and lead extraction statistically significantly. Therefore, the predicted model fits well and can be applied in the mining industry to estimate extraction efficiency and performance.

4.2.3.2 Gaussian Process Regression (GPR)

In this work, to obtain an effective predictive model, the chosen method is Gaussian Process Regression (GPR). A Gaussian process is a type of stochastic process used to model various random phenomena. It is fully described by its mean and covariance function. The mean defines the process's average value in the domain, while the covariance function specifies the correlation between values of the process at different domain points. Gaussian processes are used in many machine learning algorithms, one of which is Gaussian regression. In this case, the Gaussian process is used to model the distribution of possible regression functions for a given set of training data. Generally, analyzing Gaussian processes requires some knowledge of probability theory and linear algebra. However, there are software tools like Matlab that simplify their analysis and application in machine learning problems. Gaussian Process Regression (GPR) is a type of regression analysis that utilizes Gaussian processes to model the relationship between a set of input variables and a continuous output variable. It is a powerful and flexible machine learning technique that can be used for a wide range of tasks, including prediction, interpolation, and extrapolation. A Gaussian process is a collection of random variables, and any subset of these variables is jointly Gaussian. In GPR, it is assumed that the output variable is a sample from a Gaussian process with a mean function and a covariance function. The mean function describes the general trend of the data, while the covariance function describes the relationship between different input values. When it comes to predicting outputs, the model is trained on a set of input-output pairs. The model then uses the covariance and mean functions to make predictions on new input values. The uncertainty of the predictions is expressed by the covariance matrix of the predicted values, providing a measure of how reliable the model is. One of the advantages of GPR is its ability to handle nonlinear relationships between input and output variables, as well as its ability to provide uncertainty estimates for predictions making it suitable for both interpolation and extrapolation. The evaluation is based on its flexibility and power, as well as the ability to provide both predicted results and confidence intervals (uncertainty estimates). This aspect is crucial for quantifying the reliability of predictions. The literature shows that GPR has been widely used for data modeling in various systems. An example is the multi-response GPR model proposed by Wang and Chen, who demonstrated its superiority through simulations and modeling a chemical reaction's response. Aye and Heyns studied an integrated GPR model to predict the remaining useful life of low-speed bearings, achieving reduced prediction errors. Many articles in the literature have focused on extending and improving the GPR model to make it more accurate. In contrast, there is limited attention given to the analysis and improvement of confidence intervals, possibly because the presence of Gaussian noise has little impact on the GPR prediction accuracy, as these noises can be quantitatively modeled in the model.

4.2.3.3 Neural Networks

Neural networks, also known as artificial neural networks (ANN), constitute a subset of machine learning and derive their name from the way biological neurons communicate. They are composed of artificial neurons or nodes that receive inputs and send various outputs to connect with others. Each node has its associated weight and threshold and will send an output only if it exceeds a specific threshold value; otherwise, the connection will be interrupted. ANN relies on training data to optimize accuracy. They are used in AI to organize high-speed data into clusters. Weights are assigned to each input level, determining the variables' importance. Weighted inputs are then summed, and the output is determined by an activation function, which will decide whether to proceed to the next level in the network if the obtained value surpasses the threshold. Normally, the mechanism underlying neural networks is feedforward, meaning there is a unidirectional flow of information. Training can also be achieved through backpropagation, associating each node with its error. The type of ANN described represents the classic functioning logic. However, there are other types, such as convolutional neural networks (CNNs), mainly used for image recognition, and recurrent neural networks (RNNs), consisting of feedback loops to predict future results. Neural networks represent a crucial classification system, thanks to their versatility. They are particularly suitable when problem instances are provided in pairs, and there are many training sets. However, they have some disadvantages. For instance, with a high number of neurons, the network can lose its ability to generalize results and require lengthy and expensive training. The application of ANNs is found in various fields, such as the economic sector, where they have been used to predict bank failures. Another example is provided in article, which uses a convolutional neural network (CNN) to calculate surface roughness in a milling process, training the model using LabVIEW combined with a Python algorithm. Testing the neural network with data like those used in the training set is one of the few methods used to verify the network's effectiveness. In most cases, such traditional testing techniques are adequate for accepting a neural network system. However, in more complex and critical systems, the standard neural network training test approach is not sufficient to provide a reliable method for validation.

4.2.4 ML Testing and results

The testing phase's objectives are to evaluate the predictive model's precision by analyzing the errors that have an impact on the algorithm's predictions, and the robustness of the cleaning system's on/off control. The control functionality has been assessed by stressing the system into out-of-range points of functioning and by observing the system behavior through user monitor while considering the measured and predicted pressure drop through the ML algorithm chosen after the model validation phase. The algorithm accuracy can be achieved by comparing the errors that were recorded during the testing phase with those related to the data on which the model was trained. The approach involves acquiring data to train the ML algorithms. The data acquisition tool in LabVIEW was used to control this process. The measurements were obtained in this phase by running tests at four different air inlet velocities: 15, 17. 5, 20, and 22.5 m/s with a tolerance of 1 m/s. As a result of this stage, the four data sets displayed in Figure 49 were compiled into a singular *.csv file used to train each Python algorithm.

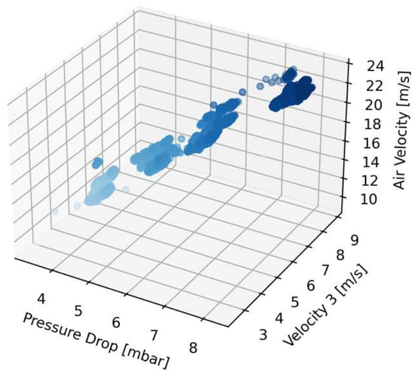


Figure 49 3D Plot Data acquired

The model validation score, which is the coefficient of determination R2, equation (13) has been used to validate each algorithm.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(x - \hat{x})^2}{\sum_{i=1}^{n}(x - \bar{x})^2} \tag{13}$$

$$MSE = \frac{\sum_{i=1}^{n}(x - \hat{x})^2}{n} \tag{14}$$

Where n is the number of samples, x is the average value of the testing data, and x is the predicted value. As shown in equation (14) the Mean Squared Error comes after:

which is the proportion of R2 to the number of samples n. At this point, the collected data were cleaned up and sent to the Python environment, where three machine learning algorithms were programmed. Multivariable linear regression (MLR), Gaussian process regression (GPR), and artificial neural network (ANN) are the first two (ANN). Each algorithm consists of three sections: I the first where data is called into the code; (ii) the second where the dataset is split into a train/test subset, of 70% and 30% respectively; (iii) it follows a third section of the code, devoted to the model prediction; (iv) while the latter section displays model scores and data distribution plots on the screen. The outcomes of this stage's equations (1) and (2) are displayed in Table 17.

Table 17 Score of the ML algorithm developed

Score	**MLR**	**GPR**	**NN**
R^2	0.98	0.96	0.95
MSE	0.032	0.002	0.002

The MLR's higher R2 demonstrates that this algorithm has been integrated into the digital environment. To achieve this, a G-Code section that evaluates the predicted value by solving the MLR function has been created.

$$Y = \alpha + \beta_1 X_1 + \beta_2 X_2 \tag{15}$$

Where Y is the estimated pressure drop value, X_1 denotes the V_3 measurement, and X_2 denotes the measured air inlet velocity. The MLR algorithms evaluate the coefficients, β_1 and β_2, and the intercept for each parameter, assuming the value shown in Table 18 after the equation (15)

Table 18 MLR Coefficients β_1, β_2 and intercept α

α	β_1	β_2
-2.16244884	0.14941325	0.48836683

Following the ML algorithm's integration into the digital environment, a second testing phase was conducted by adjusting the air inlet velocity, measuring the relevant parameters, and assessing the ML prediction. At this stage, the following formula was used to calculate the error between algorithm-estimation and measured pressure drops:

$$error = \left|1 - \frac{Predicted\ Pressure\ Drop}{Acquired\ Pressure\ Drop}\right| \tag{16}$$

Figure 50 presents the results obtained and illustrates the distribution of the error evaluated based on the test performed using the equation (16). The ordinate displays the frequency of each error percentage in a dataset of 4484 samples, and the abscissa displays the classes of error with 1.25 percent steps. Notably, the graph demonstrates that most error percentages are lower than 6%. In fact, under this threshold, errors are present in about 80% of the cases. This shows that, in many cases, the algorithm performs with a high level of accuracy.

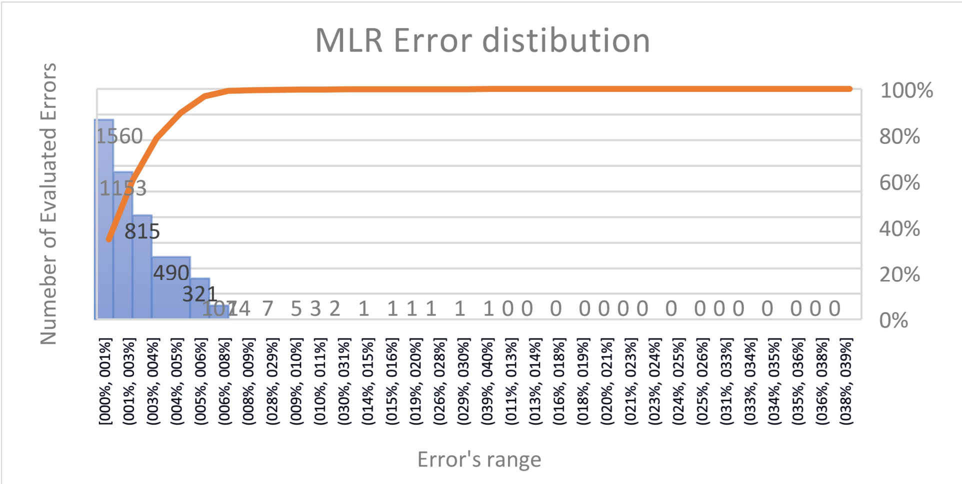


Figure 50 MLR Outcomes

The value derived from equation (15) at this point, i.e., the estimated value of pressure drops under specific conditions, can be compared in the data comparison block. The logic gate that produced the Digital Output used in the actual comparison, on which the control is based, can be used in the future to activate the air-jet compressor. If the error is within the established range or not, the logic gate is based on three cases. No action is planned if the range matches the error, and the error is less than 10%. On the other hand, the beginning of the cleaning cycle using the compressed air system is determined by an error value greater than 10% for a given duration. as a test.

4.2.5 Open loop control test and results

Like the prior instance, plant operating conditions have been used to test the open-loop control created for the bag filter. A constant compressor speed of 35 Hz has been set in the system, resulting in an air inlet velocity of roughly 19 m/s. The open-loop control is activated once the machine reaches that velocity by setting the manual parameter to false. This enables the software to adjust the necessary setpoint by shifting the valve through five different positions.

Figure 51 illustrates the outcomes that were seen. As seen in that figure, once the control is engaged, the air flow tends to stabilize and get closer to the desired velocity. The pressure distribution and velocity profile of the air movement in circular conduct, which has a non-unique value, are where the oscillating trend of the air velocity is found.

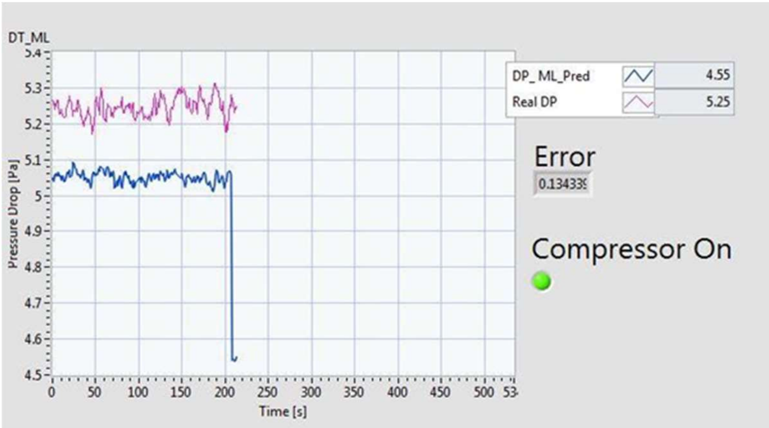


Figure 51 Front Panel: Open Loop results

The main positive finding from the tests is the quasi-stable state of air velocity when the control system is in operation. In contrast, when the setpoint changes, the velocity displays a series of peaks, primarily because the rotor speed has been set using an experimental approach, achieving the desired setpoint while also preventing damage to the valve.

5 Discussion for future research implementation

The two case studies under consideration make it possible to emphasize that, depending on the specifics of the food process, the implementation of a DT-based control system in the food processing industry involves several decisions and can be done in a variety of ways. As result of this work illustrates some important choices to be made and major issues to be addressed, regardless of the specific process.

Stating the goal of the implementation. Clearly defining the intended purpose of the DT model and control system usage is the first step in putting DT-enabled control of a food process into practice. According to (Maheshwari et al., 2023), defining a DT model entail conducting a preliminary analysis of the process, such as by physically inspecting the plant, the machines, or other pertinent resources. In the specific instance of the food processing industry, DT models can support ongoing process monitoring to improve product quality and worker safety.

Both solutions put forth in this paper incorporate a simulation tool that can use thermo-fluid dynamic equations to digitally recreate the food processes. Additionally, to exchange real data from the plant between LabVIEW (or other models/control boards) and the actuators, both solutions follow the fundamentals of a real DT as opposed to a digital shadow. Appropriate sensors are installed in various parts of the equipment (R. Liu et al., 2023).

Modeling a system. The key to DT implementation is reproducing the system in terms of variables and process parameters, but the models used for comparing the controlled parameters can change depending on the process characteristic and are obviously specific of the application context (Semeraro et al., 2023).

In fact, the digital model created for the liquid food plant includes several physical equations that consider the rheological characteristics of the process fluid, along with the thermal equations and machine geometry, allowing for the evaluation of the system's behavior and the prediction of the controlled parameters for various fluids. Contrarily, the model created for the bag filter was based on an experimentally discovered curve and required the creation of a numerical simulation using fluid dynamics for powders.

Determine a control strategy: The best control strategy should be chosen while keeping in mind the characteristics of the process under investigation. The control strategy to be used to act on the system after noticing potential incorrect operation of the equipment with the set values differs between the two case studies.

In general, both closed-loop (feedback) and open-loop (feedforward) strategies can highlight benefits or drawbacks for processes. An open-loop control might be preferred over a closed-loop one way or another because it is a simple solution that is simple to set up in systems where continuous function points of the process can be disregarded. When the controlled parameters can be set in advance within a certain tolerance and within the operating range of the machine, this method is proven to be effective.

Even so, when used for time-varying processes where the controlled parameters must be adjusted continuously, open-loop systems can exhibit some limitations; in this case, a closed-loop (i.e., feedback) system should be preferred.

It goes without saying that, depending on the context, this framework for the implementation of DT-based control systems in the food sector could be further improved by including some aspects of evaluation. To keep this framework as general as possible for this industry, it has only been applied to a few specific decisions in this article.

The fundamental mechanisms of bag filter clogging, such as particle agglomeration, cake formation, and filter media properties, can be further explored in future studies. To predict clogging events, researchers can concentrate on creating predictive models using ML and computational fluid dynamics (CFD). To forecast when and where clogging is likely to happen, these models could consider various factors like particle size distribution, airflow rates, and operating conditions. Investigating intelligent cleaning techniques, such as enhancing pulse-jet cleaning timing and frequency or applying adaptive cleaning based on real-time data, can be advantageous. This might decrease energy use and increase filter life.

In this essay, the implementation of DT models for the regulation of food processes was examined. Two actual cases—a plant for liquid foods and a large flour filter—are used as suitable examples to demonstrate the key steps in DT implementation. The paper describes the DT models and outlines the main implementation difficulties, the better control strategy for process monitoring, and a brief evaluation of that strategy. Finally, the findings from these case studies were compiled into a broad framework that reflected the essential steps for DT development and application for process control in the food industry.

From the scientific point of view, there are still few real-world examples of DT model applications in the food industry, both generally and specifically for process control. As a result, this paper adds to the body of knowledge on the subject and directly contributes by offering two instances of the use and application of DT-based control models. It is therefore anticipated that it will offer empirical examples to further support the growing interest in the DT paradigm.

However, comparisons between this study and the available literature are challenging due to the dearth of such practical implementation. Additionally, the case studies discussed are unique to food production facilities and processes, which makes it even more challenging to compare the results in a straightforward manner. These clearly show the work's limitations, but hopefully future research will evaluate them more accurately.

Notwithstanding, the implementations made it possible to list the main benefits of DT-based control models in comparison to other control strategies that are available in literature and industry. The ability of the DT-based control system architecture to automatically change the setpoint by comparing the expected delta pressure data with the experimental values is particularly important for the pasteurization process. Looking instead at the milling plant, the quasi-stable state of the air velocity when the DT-based control system is active, demonstrating good effectiveness and preventing damage to the valve, is the main strong point emerging from the testing phase.

From a pragmatic perspective, the study's final product, the framework, was developed with the intention of assisting and directing plant managers who wished to approach the implementation of the DT model for the control of food processes. This framework draws some of its inspiration from more general DT architecture proposals found in the literature, but it also adopts different viewpoints from those proposals. Although the architectural characteristics of DT models have been addressed in the literature that is currently available, their application in the field of food has only been lightly studied.

As a result, the proposed approach is unique to the food processing industry, which means that all aspects of the DT design are tailored to the specifics of the food processing industry. Second, since there are still few DT models designed for process control, the control component is also incorporated into the framework.

The proposed framework has been defined as being sufficiently general in nature so that it can be easily customized depending on the case being studied, even though it was built from the ground up using two real case studies. As a result, building on the findings of this study, additional food processes could be examined to assess the suitability of the framework created and to evaluate the effectiveness of DT-based control models of food processes in various contexts.

In summary, this work has introduced an application aimed at the integration of digital twin models, machine-learning algorithms, and Industry 4.0 technologies to create a comprehensive tool for control processes and provide anomaly detection within industrial systems. The focus of this study has been the development and implementation of a solution tailored for use in a tube-in-tube indirect machine designed for fluid food pasteurization, and a bag filter pilot plant. To optimize the performance of this multifaceted approach, four distinct operational modes were devised and integrated into the digital twin model of the plant. A series of empirical tests was conducted to ascertain which operational mode most accurately aligns with the real-world functioning of the system.

While the digital twin environment, coupled with tools from prior research, allows for both in-situ and remote monitoring and control, it does present certain limitations, particularly in terms of manual setpoint adjustments and fluid characteristic configurations. In response to this challenge, three distinct machine learning approaches, including a linear regression model, an artificial neural network, and a clustering algorithm, were incorporated into the solution, primarily for online monitoring of the plant. This endeavour, exploratory in nature, sought to evaluate the efficacy of various machine learning algorithms for anomaly detection within this specific application, an area of research that remains relatively unexplored.

The outcomes from these machine learning tools indicate that the regression algorithm, once integrated into the digital twin environment, holds promise as a means of achieving automatic control over the system. This is attributed to its ability to predict the dependent variable (P1) based on multiple independent variables (P2 and F), subsequently providing a discrete value that can be employed as a setpoint for the PID controller. Simultaneously, the digital twin model operates as a vigilant sentinel over the machine's behavior, overseeing its functioning and capable of halting operations upon detecting a "failure," while displaying the machine's status on the HMI. Conversely, the artificial neural network and clustering algorithms demonstrated marginally less impressive performance. Specifically, the MLPC algorithm exhibited a high accuracy in predicting "ok" or "failure" status but displayed lower precision in classifying the "warning" status. Similar conclusions were drawn regarding the k-means clustering algorithm, which exhibited the capability to distinguish between "failure" and "ok" statuses but struggled with the "warning" status, which encompasses a relatively narrow range of values, making precise detection challenging.

It is important to note that false-negative classifications of the "failure" status, though relatively infrequent, pose a significant concern, particularly in terms of employee safety. Therefore, refinements are essential for these algorithms to minimize false negatives in identifying machine failures. Alternatively, a pragmatic approach might involve the combined use of both methods, significantly reducing the likelihood of concurrent false negatives for "failure" status.

In practical application, the use of clustering or classification algorithms for anomaly detection would be best suited for identifying "ok" conditions, where their performance is generally reliable. Any deviation from this status would trigger alerts on the HMI, signalling the need for intervention. Additionally, real-world implementation should account for the presence of outliers or noisy data, which could adversely affect the performance of clustering algorithms, necessitating careful data preprocessing in practical scenarios.

In conclusion, this research provides a promising foundation for the development of an integrated digital twin-based system enhanced by machine learning algorithms for anomaly detection in industrial settings. The findings not only advance the knowledge in this domain but also underscore the importance of refining these methodologies to ensure the utmost safety and efficacy in industrial operations. By investigating the possibility of integrating them with ML algorithms, future research activities could also address further advancements in the application and adoption of DT in the food processing industry. Predictive and prescriptive analytics will be used to analyze real-time data from the DT model to gain additional insights for improving food process controls.

The monitoring, control, and maintenance optimization of pasteurization system and, a bag filter pilot plant, are addressed in this paper using an integrated Machine Learning solution created with Python and LabVIEW. The suggested system employs a data comparison tool to activate compressor filter cleaning or adjust product velocity in the pasteurizer. Three suitable algorithms for control management were chosen after a thorough review of the literature. Using a Data Acquisition System (DAQ) tool, a dataset was gathered and used for algorithm testing and training. Based on the results of the model validation, the implemented algorithm was chosen. To evaluate the performance of the chosen algorithms and the Data Comparison tool, real data acquisition and validation were carried out. The compressed air cleaning system is activated when a significant difference is found by the control system developed in LabVIEW, which compares estimated pressure drops with directly recorded values.

The difference between estimated and acquired values serves as a warning to the user that the filter sleeve needs to be replaced, or to control product flow. To assess the precision of the acquired and predicted data, various tests at various air inlet velocities were carried out. The findings show that in about 80% of the cases, the machine learning algorithm maintains an error distribution below 5 percent. By implementing Industry 4.0 principles and using digitalization for predictive maintenance, this research helps industrial plants cut costs and downtime. The system architecture, which consists of both hardware and software layers, makes it possible to effectively monitor, control, and optimize maintenance of both plants. The created ML algorithms demonstrate the potential of predictive maintenance in enhancing plant performance, efficiency, and resource utilization after being validated through performance metrics. The maintenance management of industrial plants can be improved by additional study and application of Intelligent Predictive Maintenance (IPdM) and Industry 4.0 technologies.

6 Conclusions

In summary, this academic endeavour has focused on the complex realm of predictive maintenance and process control within food industry manufacturing field. Through the integration of ML in a DT environment, a comprehensive and innovative solution has been elucidated. This study has underscored the critical importance of predictive maintenance in optimizing the process control and predictive mainentance of industrial processes. The results and insights presented here emphasize the potential of employing ML algorithms within the digital twin environment for anomaly detection and automated control. The integration of ML algorithms, such as linear regression, artificial neural networks, and clustering, has shown as a promising avenue for enhancing process control and predictive maintenance. The findings reveal that the linear regression algorithm, when embedded within the digital twin, offers automated control capabilities by predicting and directly setting parameters. However, it is crucial to acknowledge the need for continuous refinement and improvement, particularly in the context of accurately identifying anomalies, a challenge that the artificial neural network and clustering algorithms have grappled with.

The significance of this work extends beyond the theoretical realm, as it has practical implications for industrial pilot plant operations. The application of Industry 4.0 principles and digitalization, alongside the amalgamation of predictive maintenance and ML, offers a pragmatic solution to cut costs, minimize downtime, and enhance overall plant performance. The proposed system architecture, encompassing both hardware and software layers, demonstrates its capacity for effective monitoring, control, and maintenance optimization. This integrated approach provides the foundation for improved industrial plant management, thus contributing to advancements in IPdM and Industry 4.0 technologies.

In conclusion, this study has highlighted the potential of combining ML algorithms with the digital twin model, offering a holistic solution for predictive maintenance and process control in industrial pilot plants. The findings not only enrich the existing body of knowledge but also underscore the practical relevance of these integrated technologies in real-world industrial applications. As we move forward, it is our duty to continue refining and expanding these methodologies, ensuring their seamless adaptation to the evolving needs and complexities of industrial systems.

7 References

Abdelsalam, A., Luglio, M., Roseti, C., & Zampognaro, F. (2017). TCP Wave: A new reliable transport approach for future internet. *Computer Networks*, *112*, 122–143. https://doi.org/10.1016/J.COMNET.2016.11.002

Afram, A., Janabi-Sharifi, F., Fung, A. S., & Raahemifar, K. (2017). Artificial neural network (ANN) based model predictive control (MPC) and optimization of HVAC systems: A state of the art review and case study of a residential HVAC system. *Energy and Buildings*, *141*, 96–113. https://doi.org/10.1016/J.ENBUILD.2017.02.012

Alphonsus, E. R., & Abdullah, M. O. (2016). A review on the applications of programmable logic controllers (PLCs). *Renewable and Sustainable Energy Reviews*, *60*, 1185–1205. https://doi.org/10.1016/J.RSER.2016.01.025

Ayvaz, S., & Alpay, K. (2021). Predictive maintenance system for production lines in manufacturing: A machine learning approach using IoT data in real-time. *Expert Systems with Applications*, *173*, 114598. https://doi.org/10.1016/J.ESWA.2021.114598

Bagyaveereswaran, V., Mathur, T. D., Gupta, S., & Arulmozhivarman, P. (2016). Performance comparison of next generation controller and MPC in real time for a SISO process with low cost DAQ unit. *Alexandria Engineering Journal*, *55*(3), 2515–2524. https://doi.org/10.1016/J.AEJ.2016.07.028

Bianco, N., Mauro, A. W., Mauro, G. M., Pantaleo, A. M., & Viscito, L. (2022). A semi-empirical model for de-watering and cooling of leafy vegetables. *Applied Thermal Engineering*, *208*, 118227. https://doi.org/10.1016/J.APPLTHERMALENG.2022.118227

Boje, C., Guerriero, A., Kubicki, S., & Rezgui, Y. (2020). Towards a semantic Construction Digital Twin: Directions for future research. *Automation in Construction*, *114*, 103179. https://doi.org/10.1016/J.AUTCON.2020.103179

Bottani, E., Vignali, G., & Carlo Tancredi, G. P. (2020). A digital twin model of a pasteurization system for food beverages: Tools and architecture. *Proceedings - 2020 IEEE International Conference on Engineering, Technology and Innovation, ICE/ITMC 2020*. https://doi.org/10.1109/ICE/ITMC49519.2020.9198625

Caro, D. (2016). Industrial data communications protocols and application layers. *Industrial Wireless Sensor Networks: Monitoring, Control and Automation*, 3–23. https://doi.org/10.1016/B978-1-78242-230-3.00001-5

Davila Delgado, J. M., & Oyedele, L. (2021). Digital Twins for the built environment: learning from conceptual and process models in manufacturing. *Advanced Engineering Informatics*, *49*, 101332. https://doi.org/10.1016/J.AEI.2021.101332

de Souza Mendes, P. R. (2007). Dimensionless non-Newtonian fluid mechanics. *Journal of Non-Newtonian Fluid Mechanics*, *147*(1–2), 109–116. https://doi.org/10.1016/J.JNNFM.2007.07.010

Defraeye, T., Shrivastava, C., Berry, T., Verboven, P., Onwude, D., Schudel, S., Bühlmann, A., Cronje, P., & Rossi, R. M. (2021). Digital twins are coming: Will we need them in supply chains of fresh horticultural produce? *Trends in Food Science & Technology*, *109*, 245–258. https://doi.org/10.1016/J.TIFS.2021.01.025

Earle, M. D. (1997). Innovation in the food industry. *Trends in Food Science & Technology*, *8*(5), 166–175. https://doi.org/10.1016/S0924-2244(97)01026-1

Errandonea, I., Beltrán, S., & Arrizabalaga, S. (2020). Digital Twin for maintenance: A literature review. *Computers in Industry*, *123*, 103316. https://doi.org/10.1016/J.COMPIND.2020.103316

Fortela, D. L. B., & Mikolajczyk, A. P. (2023). Detecting Plant-Wide Oscillation Propagation Effects of Disturbances and Faults in a Chemical Process Plant Using Network Topology of Variance Decompositions. *Processes*, *11*(6), 1747. https://doi.org/10.3390/pr11061747

Gai, H., Yang, P., Zhang, Q., Lin, M., Song, H., Xiao, M., Huang, T., & Zhu, Q. (2023). Dual-dense gas-solid circulating fluidized bed reactor. *Fuel*, *337*, 126872. https://doi.org/10.1016/J.FUEL.2022.126872

Gandomi, A., & Haider, M. (2015). Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, *35*(2), 137–144. https://doi.org/10.1016/J.IJINFOMGT.2014.10.007

Gao, Y. (2023). PID-based search algorithm: A novel metaheuristic algorithm based on PID algorithm. *Expert Systems with Applications*, *232*, 120886. https://doi.org/10.1016/J.ESWA.2023.120886

Grieves, M., & Vickers, J. (2016). Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In *Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches* (pp. 85–113). Springer International Publishing. https://doi.org/10.1007/978-3-319-38756-7_4

Guruswamy, S., Pojić, M., Subramanian, J., Mastilović, J., Sarang, S., Subbanagounder, A., Stojanović, G., & Jeoti, V. (2022). Toward Better Food Security Using Concepts from Industry 5.0. *Sensors*, *22*(21), 8377. https://doi.org/10.3390/s22218377

Han, C., Jornet, J. M., Fadel, E., & Akyildiz, I. F. (2013). A cross-layer communication module for the Internet of Things. *Computer Networks*, *57*(3), 622–633. https://doi.org/10.1016/J.COMNET.2012.10.003

Han, X., Wang, Z., Xie, M., He, Y., Li, Y., & Wang, W. (2021). Remaining useful life prediction and predictive maintenance strategies for multi-state manufacturing systems considering functional dependence. *Reliability Engineering & System Safety*, *210*, 107560. https://doi.org/10.1016/J.RESS.2021.107560

Hazrathosseini, A., & Moradi Afrapoli, A. (2023). The advent of digital twins in surface mining: Its time has finally arrived. *Resources Policy*, *80*, 103155. https://doi.org/10.1016/J.RESOURPOL.2022.103155

Henrichs, E., Noack, T., Pinzon Piedrahita, A. M., Salem, M. A., Stolz, J., & Krupitzer, C. (2021a). Can a Byte Improve Our Bite? An Analysis of Digital Twins in the Food Industry. *Sensors*, *22*(1), 115. https://doi.org/10.3390/s22010115

Henrichs, E., Noack, T., Pinzon Piedrahita, A. M., Salem, M. A., Stolz, J., & Krupitzer, C. (2021b). Can a Byte Improve Our Bite? An Analysis of Digital Twins in the Food Industry. *Sensors*, *22*(1), 115. https://doi.org/10.3390/s22010115

Jiang, H., Qin, S., Fu, J., Zhang, J., & Ding, G. (2021). How to model and implement connections between physical and virtual models for digital twin application. *Journal of Manufacturing Systems*, *58*, 36–51. https://doi.org/10.1016/J.JMSY.2020.05.012

Jones, D., Snider, C., Nassehi, A., Yon, J., & Hicks, B. (2020). Characterising the Digital Twin: A systematic literature review. *CIRP Journal of Manufacturing Science and Technology*, *29*, 36–52. https://doi.org/10.1016/J.CIRPJ.2020.02.002

Kannapinn, M., Pham, M. K., & Schäfer, M. (2022). Physics-based digital twins for autonomous thermal food processing: Efficient, non-intrusive reduced-order modeling. *Innovative Food Science & Emerging Technologies*, *81*, 103143. https://doi.org/10.1016/J.IFSET.2022.103143

Karagiannopoulos, P. S., Manousakis, N. M., & Psomopoulos, C. S. (2023). “3R” Practices Focused on Home Appliances Sector in Terms of Green Consumerism: Principles, Technical Dimensions and Future Challenges. *IEEE Transactions on Consumer Electronics*. https://doi.org/10.1109/TCE.2023.3318874

Khan, M., Tahiyat, M., Imtiaz, S., Choudhury, M. A. A. S., & Khan, F. (2017). Experimental evaluation of control performance of MPC as a regulatory controller. *ISA Transactions*, *70*, 512–520. https://doi.org/10.1016/J.ISATRA.2017.04.024

King, A., & Hunt, R. (2000). Protocols and architecture for managing TCP/IP network infrastructures. *Computer Communications*, *23*(16), 1558–1572. https://doi.org/10.1016/S0140-3664(00)00214-0

Kober, C., Algan, B. N., Fette, M., & Wulfsberg, J. P. (2023). Relations of Digital Twin Fidelity and Benefits: A Design-to-Value Approach. *Procedia CIRP*, *119*, 809–815. https://doi.org/10.1016/J.PROCIR.2023.03.126

Kumar, I., Rawat, J., Mohd, N., & Husain, S. (2021). Opportunities of Artificial Intelligence and Machine Learning in the Food Industry. *Journal of Food Quality*, *2021*. https://doi.org/10.1155/2021/4535567

Lakshmi, K. V., Panda, R. C., Manimozhi, M., & Harshad, S. K. (2022). Design and Implementation of Conventional and Adavanced Controllers for Level Control of Coupled Tank Process. *7th International Conference on Communication and Electronics Systems, ICCES 2022 - Proceedings*, 253–260. https://doi.org/10.1109/ICCES54183.2022.9835970

Lee, I., & Lee, K. (2015). The Internet of Things (IoT): Applications, investments, and challenges for enterprises. *Business Horizons*, *58*(4), 431–440. https://doi.org/10.1016/J.BUSHOR.2015.03.008

Leng, J., Sha, W., Wang, B., Zheng, P., Zhuang, C., Liu, Q., Wuest, T., Mourtzis, D., & Wang, L. (2022). Industry 5.0: Prospect and retrospect. *Journal of Manufacturing Systems*, *65*, 279–295. https://doi.org/10.1016/J.JMSY.2022.09.017

Li, L., Lei, B., & Mao, C. (2022). Digital twin in smart manufacturing. *Journal of Industrial Information Integration*, *26*, 100289. https://doi.org/10.1016/J.JII.2021.100289

Liu, M., Fang, S., Dong, H., & Xu, C. (2021). Review of digital twin about concepts, technologies, and industrial applications. *Journal of Manufacturing Systems*, *58*, 346–361. https://doi.org/10.1016/J.JMSY.2020.06.017

Liu, R., Zhang, C., Ji, H., Zhang, C., & Qiu, J. (2023). Training, Control and Application of SMA-Based Actuators with Two-Way Shape Memory Effect. *Actuators*, *12*(1), 25. https://doi.org/10.3390/act12010025

Lyytinen, K., Yoo, Y., & Boland, R. J. (2016). Digital product innovation within four classes of innovation networks. *Information Systems Journal*, *26*(1), 47–75. https://doi.org/10.1111/isj.12093

Maheshwari, P., Kamble, S., Belhadi, A., Mani, V., & Pundir, A. (2022). Digital twin implementation for performance improvement in process industries- A case study of food processing company. *International Journal of Production Research*, 1–23. https://doi.org/10.1080/00207543.2022.2104181

Maheshwari, P., Kamble, S., Belhadi, A., Venkatesh, M., & Abedin, M. Z. (2023). Digital twin-driven real-time planning, monitoring, and controlling in food supply chains. *Technological Forecasting and Social Change*, *195*, 122799. https://doi.org/10.1016/J.TECHFORE.2023.122799

Malkin, A. Y. (2013). Non-Newtonian viscosity in steady-state shear flows. *Journal of Non-Newtonian Fluid Mechanics*, *192*, 48–65. https://doi.org/10.1016/J.JNNFM.2012.09.015

Maxim, A., Copot, D., De Keyser, R., & Ionescu, C. M. (2018). An industrially relevant formulation of a distributed model predictive control algorithm based on minimal process information. *Journal of Process Control*, *68*, 240–253. https://doi.org/10.1016/J.JPROCONT.2018.06.004

McFarlane, I. (1995). Manufactured products. *Automatic Control of Food Manufacturing Processes*, 134–170. https://doi.org/10.1007/978-1-4615-2187-7_5

Melesse, T. Y., Di Pasquale, V., & Riemma, S. (2020). Digital Twin Models in Industrial Operations: A Systematic Literature Review. *Procedia Manufacturing*, *42*, 267–272. https://doi.org/10.1016/J.PROMFG.2020.02.084

Melville, N. P., Robert, L., & Xiao, X. (2023). Putting humans back in the loop: An affordance conceptualization of the 4th industrial revolution. *Information Systems Journal*, *33*(4), 733–757. https://doi.org/10.1111/isj.12422

Messai, A., Mellit, A., Guessoum, A., & Kalogirou, S. A. (2011). Maximum power point tracking using a GA optimized fuzzy logic controller and its FPGA implementation. *Solar Energy*, *85*(2), 265–277. https://doi.org/10.1016/J.SOLENER.2010.12.004

Mikalef, P., Pappas, I. O., Krogstie, J., Giannakos, M., & No, P. M. (2018). Big data analytics capabilities: a systematic literature review and research agenda. *Inf Syst E-Bus Manage*, *16*, 547–578. https://doi.org/10.1007/s10257-017-0362-y

Morgan, M. T., & Haley, T. A. (2019). Design of Food Process Controls Systems. *Handbook of Farm, Dairy and Food Machinery Engineering*, 533–591. https://doi.org/10.1016/B978-0-12-814803-7.00022-1

Mourtzis, D., Angelopoulos, J., & Panopoulos, N. (2022). A Literature Review of the Challenges and Opportunities of the Transition from Industry 4.0 to Society 5.0. *Energies*, *15*(17), 6276. https://doi.org/10.3390/en15176276

Naha, S., & Das, D. K. (2024). Radial basis function neural network controller for power control of molten salt breeder reactor of nuclear power plant. *Annals of Nuclear Energy*, *195*, 110160. https://doi.org/10.1016/J.ANUCENE.2023.110160

Novák, P., & Vyskočil, J. (2022). Digitalized Automation Engineering of Industry 4.0 Production Systems and Their Tight Cooperation with Digital Twins. *Processes*, *10*(2), 404. https://doi.org/10.3390/pr10020404

Nychas, G.-J., Sims, E., Tsakanikas, P., & Mohareb, F. (2021). Data Science in the Food Industry. *Annual Review of Biomedical Data Science*. https://doi.org/10.1146/annurev-biodatasci-020221

Olaizola, I., Quartulli, M., Unzueta, E., Goicolea, J., & Flórez, J. (2022). Refinery 4.0, a Review of the Main Challenges of the Industry 4.0 Paradigm in Oil & Gas Downstream. *Sensors*, *22*(23), 9164. https://doi.org/10.3390/s22239164

Oluwaleye, S., Oguntosin, V., & Idachaba, F. (2021). Conceptual design of smart multi-farm produce dehydrator using a low-cost programmable logic controller and raspberry pi. *F1000Research*, *10*, 810. https://doi.org/10.12688/f1000research.54463.1

Perno, M., Hvam, L., & Haug, A. (2022). Implementation of digital twins in the process industry: A systematic literature review of enablers and barriers. *Computers in Industry*, *134*, 103558. https://doi.org/10.1016/J.COMPIND.2021.103558

Priyanka, E. B., Thangavel, S., Prasad, P. H., & Mohanasundaram, R. (2021). IoT fusion based model predictive pid control approach for oil pipeline infrastructure. *International Journal of Critical Infrastructure Protection*, *35*, 100485. https://doi.org/10.1016/J.IJCIP.2021.100485

Rao, V. S., Praveen Shenoy, K., & Santhosh, K. V. (2021). Design of PLC based automated food processing machine. *International Journal of Mechanics*, *15*, 22–29. https://doi.org/10.46300/9104.2021.15.3

Rostam, M., Nagamune, R., & Grebenyuk, V. (2023). Self-tuning kernel Gaussian method for predictive control systems. *Journal of Process Control*, *128*, 103009. https://doi.org/10.1016/J.JPROCONT.2023.103009

Sarantinoudis, N., Tsinarakis, G., Dedousis, P., & Tsinarakis, G. (n.d.). *Model-Based Simulation Framework for Digital Twins in the Process Industry*. https://doi.org/10.1109/ACCESS.2023.3322926

Semeraro, C., Lezoche, M., Panetto, H., & Dassisti, M. (2021). Digital twin paradigm: A systematic literature review. *Computers in Industry*, *130*, 103469. https://doi.org/10.1016/J.COMPIND.2021.103469

Semeraro, C., Lezoche, M., Panetto, H., & Dassisti, M. (2023). Data-driven invariant modelling patterns for digital twin design. *Journal of Industrial Information Integration*, *31*, 100424. https://doi.org/10.1016/J.JII.2022.100424

Singh, M., Fuenmayor, E., Hinchy, E., Qiao, Y., Murray, N., & Devine, D. (2021). Digital Twin: Origin to Future. *Applied System Innovation*, *4*(2), 36. https://doi.org/10.3390/asi4020036

Soyguder, S., & Alli, H. (2010). Fuzzy adaptive control for the actuators position control and modeling of an expert system. *Expert Systems with Applications*, *37*(3), 2072–2080. https://doi.org/10.1016/J.ESWA.2009.06.071

Stavropoulos, P., Papacharalampopoulos, A., Sabatakakis, K., & Mourtzis, D. (2023). Metamodelling of Manufacturing Processes and Automation Workflows towards Designing and Operating Digital Twins. *Applied Sciences*, *13*(3), 1945. https://doi.org/10.3390/app13031945

Tancredi, G. P., Vignali, G., & Bottani, E. (2022a). Integration of Digital Twin, Machine-Learning and Industry 4.0 Tools for Anomaly Detection: An Application to a Food Plant. *Sensors*, *22*(11). https://doi.org/10.3390/s22114143

Tancredi, G. P., Vignali, G., & Bottani, E. (2022b). Integration of Digital Twin, Machine-Learning and Industry 4.0 Tools for Anomaly Detection: An Application to a Food Plant. *Sensors*, *22*(11), 4143. https://doi.org/10.3390/s22114143

Tao, F., Xiao, B., Qi, Q., Cheng, J., & Ji, P. (2022). Digital twin modeling. *Journal of Manufacturing Systems*, *64*, 372–389. https://doi.org/10.1016/J.JMSY.2022.06.015

Torres, A. G. D., & Galvis, I. C. G. (2017). Innovation and creativity in process control and manufacturing. *International Journal on Interactive Design and Manufacturing*, *11*(2), 173–189. https://doi.org/10.1007/s12008-014-0258-8

Trienekens, J., & Zuurbier, P. (2008). Quality and safety standards in the food industry, developments and challenges. *International Journal of Production Economics*, *113*(1), 107–122. https://doi.org/10.1016/J.IJPE.2007.02.050

van Niekerk, J. A., le Roux, J. D., & Craig, I. K. (2023). On-line automatic controller tuning of a multivariable grinding mill circuit using Bayesian optimisation. *Journal of Process Control*, *128*, 103008. https://doi.org/10.1016/J.JPROCONT.2023.103008

VanDerHorn, E., & Mahadevan, S. (2021). Digital Twin: Generalization, characterization and implementation. *Decision Support Systems*, *145*, 113524. https://doi.org/10.1016/J.DSS.2021.113524

Verboven, P., Defraeye, T., Datta, A. K., & Nicolai, B. (2020). Digital twins of food process operations: the next step for food process models? *Current Opinion in Food Science*, *35*, 79–87. https://doi.org/10.1016/J.COFS.2020.03.002

Wakitani, S., Yamamoto, T., & Gopaluni, B. (2019). Design and application of a database-driven PID Controller with data-driven updating algorithm. *Industrial and Engineering Chemistry Research*, *58*(26), 11419–11429. https://doi.org/10.1021/acs.iecr.9b00704

Wang, B., Tao, F., Fang, X., Liu, C., Liu, Y., & Freiheit, T. (2021). Smart Manufacturing and Intelligent Manufacturing: A Comparative Review. *Engineering*, *7*(6), 738–757. https://doi.org/10.1016/J.ENG.2020.07.017

Wang, M., Lu, T., & Li, Y. (2023). Optimizing air purification for household particulate matters using sensor-based and time-based intervention strategies. *Particuology*, *79*, 78–84. https://doi.org/10.1016/J.PARTIC.2022.11.008

Weldcherkos, T., Salau, A. O., & Ashagrie, A. (2021). Modeling and design of an automatic generation control for hydropower plants using Neuro-Fuzzy controller. *Energy Reports*, *7*, 6626–6637. https://doi.org/10.1016/J.EGYR.2021.09.143

Yu, H., Guan, Z., Chen, T., & Yamamoto, T. (2020). Design of data-driven PID controllers with adaptive updating rules. *Automatica*, *121*, 109185. https://doi.org/10.1016/J.AUTOMATICA.2020.109185

Yu, W., Patros, P., Young, B., Klinac, E., & Walmsley, T. G. (2022). Energy digital twin technology for industrial energy management: Classification, challenges and future. *Renewable and Sustainable Energy Reviews*, *161*, 112407. https://doi.org/10.1016/J.RSER.2022.112407

Zewdie, T. A., Delele, M. A., Fanta, S. W., Alemayehu, M., Alemayehu, G., Adgo, E., Nyssen, J., Verboven, P., & Nicolai, B. M. (2022). Optimisation of onion bulb curing using a heat and mass transfer model. *Biosystems Engineering*, *214*, 42–57. https://doi.org/10.1016/J.BIOSYSTEMSENG.2021.12.009

Zhang, C., Zhou, G., Li, H., & Cao, Y. (2020). Manufacturing Blockchain of Things for the Configuration of a Data-and Knowledge-Driven Digital Twin Manufacturing Cell. *IEEE INTERNET OF THINGS JOURNAL*, *7*(12). https://doi.org/10.1109/JIOT.2020.3005729

Zhang, H., Zhang, G., & Yan, Q. (2019). *Digital twin-driven cyber-physical production system towards smart shop-floor*. *10*, 4439–4453. https://doi.org/10.1007/s12652-018-1125-4