Doctoral School of Economics, Business and Informatics

Production System Efficiency Optimization Using Hybrid AI Solution and Sensor Data

Ph.D. Dissertation

Supervisor Dr Tibor Kovacs, Dr Andrea Ko

Joao Henrique Gomes da Costa Cavalcanti

Budapest 2025

Joao Henrique Gomes da Costa Cavalcanti

Corvinus University of Budapest

Department of Information Systems

Supervisors: Dr Tibor Kovacs, Dr Andrea Ko

INDEX

2

1-INTRODUCTION 8

2-RESEARCH PROBLEM 12

3-RESEARCH OBJECTIVE & RESEARCH QUESTIONS 14

4-THEORETICAL FRAMEWORK 15

4.1-ARTIFICIAL INTELLIGENCE, HYBRID AI AND MACHINE LEARNING 15
4.2-PRODUCTION EFFICIENCY AND OPTIMIZATION 18
4.3-DIGITAL TWINS AND ML BASED DIGITAL TWIN 20
4.4-DOE 22
4.5-DEA 23
4.6-GENETIC ALGORITHMS 25
4.7-FEEDBACK CONTROL LOOPS 28
5-LITERATURE REVIEW 31

5.1- LITERATURE REVIEW USING SCOPUS 31
5.2- RELATED WORKS 36
5.3-SYTEM CONTROL AND FEEDBACK LOOP METHOD 37
5.4-ML-BASED DT 39
5.5-RESEARCH GAP 42
6-RESEARCH METHODOLOGY AND METHODS 43

6.1-RESEARCH METHODOLOGY 43
6.2-DATA COLLECTION 49
6.3-METHODS 49
6.3.1-DEA 50
6.3.2-MACHINE LEARNING 50
6.3.3-GENETIC ALGORITHM 52
6.3.4-Z-SCORE METHOD 53
6.3.5-R SQUARED 55
7-FRAMEWORK FOR PRODUCTION EFFICIENCY OPTIMIZATION USING MACHINE LEARNING BASED DIGITAL TWIN SIMULATION 56

7.1 MODELING A COMPLEX PRODUCTION SYSTEM 57
7.2 OPTIMIZATION TECHNIQUES ON ML-BASED DIGITAL TWIN 61
7.3 AI MODEL OVERVIEW 63

7.4 MODEL INPUTS AND SENSOR DATA..... 66
7.4.1 EXPERT DOMAIN KNOWLEDGE..... 67
7.4.2 SENSOR DATA 68
7.4.3 DESIRED PRODUCTION OUTPUT 68
7.5 DATA PREPARATION AND DEA 69
7.6 ML BASED DIGITAL TWIN MODEL TRAINING..... 70
7.7 GENETIC ALGORITHM OPTIMIZATION 73
7.8 FORMAL MODEL DESCRIPTION..... 78
8-IMPLEMENTATION OF THE HYBRID AI SOLUTION IN ENERGY SECTOR 81
8.1 PRODUCTION SYSTEM VARIABLES 81
8.2 DATABASE PREPARATION AND EXPLORATORY DATA ANALYSIS..... 84
8.3 PROPOSED AI SOLUTION ADAPTED FOR THERMOELECTRIC PRODUCTION SYSTEM 88
8.4 RESULTS..... 93
8.4.1-DEA 93
8.4.2- MACHINE LEARNING MODELS..... 98
8.4.3- GENETIC ALGORITHM 102
8.5 DISCUSSION 105
9 CONCLUSION..... 110
10-REFERENCES..... 114

TABLE OF FIGURES

Figure 1 - GA process

Figure 2 - Trend of publications over the years of 2014 to 2024

Figure 3 – Implementation of DSR steps in this research

Figure 4 - Complex system modelling

Figure 5 - The optimization process of the production on user request

Figure 6 - ML-based DT for production optimization

Figure 7– AI framework overview

Figure 8 - ML model training process

Figure 9 - GA workflow within the AI solution

Figure 10 - Conceptual model of the production system

Figure 11 - Schematic diagram of the thermoelectric plant

Figure 12 – Steam production timeseries

Figure 13 – Wood infeed timeseries

Figure 14 – Wood infeed A Histogram

Figure 15 – Oil infeed timeseries

Figure 16 – Oil infeed HistogramFigure 17 - DEA efficiency estimation and ML training phases of the hybrid AI solution

Figure 18 - The Genetic Algorithm optimisation phase of the hybrid AI solution

Figure 19 - DEA relative efficiency distributions of the two models for year 2020

Figure 20 - Correlation table of selected input, output, process control and state variables of the two models

Figure 21 - Relationship between process control and state variables as well as DEA efficiency achieved.

Figure 22 - Cluster of high-efficiency combinations of wood chip inflows

Figure 23 - Improvement of the fitness through generations of GA

Figure 24 - Efficiencies with and without the AI solution for model 1 and 2 (grey points are observations violating state variable constraints)

LIST OF TABLES

Table 1 - ML model optimal parameters and performance

Table 2 - ML model optimal parameters and performance

Table 2 - Optimal recommendations for an output of 168 MJ/h and 48 MW/hr (model 1 and 2)

Table 3 - State variable values for the optimal solution

Table 5 - Calculation time of the AI solution for model 1 and 2

1-Introduction

In today's rapidly evolving manufacturing and production landscape, achieving operational excellence and efficiency is a constant goal for organizations worldwide. Industry 4.0 has brought a digital transformation, introducing technologies such as the Internet of Things (IoT), big data analytics, and artificial intelligence (AI) (Hayat et al., 2023). These innovations fundamentally reshape business operations, and among them, the concept of a "Digital Twin" (DT) stands out as a transformative paradigm. A DT is a virtual replica of a physical system or object, mirroring its attributes, behavior, and real-time status (Rane, 2023). This dynamic, data-driven model provides a comprehensive view of processes and systems It also provides new opportunities for modelling, simulation.(Aheleroff et al., 2021).

Manufacturing and production enterprises face numerous challenges, including complex supply chains, the demand for customized products, and stringent quality and sustainability standards. Optimizing production processes to enhance resource utilization, quality control, energy management, and scheduling is crucial for maintaining competitiveness and sustainability. Machine learning (ML) has emerged as a powerful tool for extracting insights and actionable intelligence from large datasets, offering significant potential for improving production efficiency, moreover, ML applications in predictive maintenance, process optimization, and demand forecasting can reduce waste, lower operational costs, and enhance overall production efficiency (Bharadiya, 2023).

A key aspect of DT is its ability to model virtual processes that describe how a system changes states. These models can be computational, representing physical relationships, or data-driven, using ML and AI techniques, or a hybrid of both.

Computational models can be complex and expensive to implement, making data-driven approaches more attractive (VanDerHorn & Mahadevan, 2021).

Lean manufacturing, a philosophy focused on eliminating production waste, is a key strategy in this pursuit. Industry 4.0 supports lean manufacturing with advanced technologies like AI, robotics, IoT, and cloud computing, aiming to reduce costs, minimize waste, and improve product quality (Pagliosa et al., 2021). This dissertation addresses gaps in the literature by introducing an innovative Artificial Intelligence Framework for optimizing production efficiency. The framework, incorporating ML-based DT Simulation, genetic algorithms (GA), and data envelopment analysis (DEA), is designed for application in various production systems. I provide practical guidelines and examples for implementing this comprehensive framework in real-world practices, and a practical example on the energy production industry. Additionally, the dissertation includes a formal model description, enhancing the clarity and precision of the presented model.

The research aims to develop an AI framework that provides a cost-effective and accessible solution for integrating ML into DTs for optimization purposes. This framework uses ML algorithms trained on historical or synthetic data to simulate system behavior without relying on continuous real-time data from the physical counterpart. By combining ML, GA, and DEA, the research seeks to optimize production systems by offering recommendations for production setup and process control parameters, improving resource usage and overall efficiency. The originality of this approach lies in the novel integration of these methods, applied and validated on a sensor dataset from an energy production system. The proposed hybrid AI solution enhances production efficiency by using DEA to identify the efficient frontier of the production system, ML to make efficiency predictions through simulations based on these DEA results, and GA to optimize input configurations and process control parameters. The framework is tested with real-world data from a

thermoelectric power plant, navigating the complex interactions between inputs, outputs, and state variables. DEA informs the ML model for accuracy in efficiency predictions, and GA optimizes the system further, ensuring the best possible production setup. This integration of DEA, ML, and GA in a unified framework offers an innovative approach to addressing production optimization challenges, particularly in energy production systems.

A challenge arises when the DEA-ML engine relies on historical data generated through trial and error. If these data predominantly reflect inefficient production settings, the GA may return suboptimal solutions. To mitigate this, the design of experiments (DOE) can generate a broad spectrum of data, pushing the efficiency frontier further and yielding superior results. The hybrid AI solution aims to minimize waste and enhance production efficiency and was tested in the power generation industry, reducing energy production costs and benefiting the environment and society. Real-world testing validates the AI solution's potential, demonstrating its versatility across various domains. The research applies a novel approach and demonstrates through a real-world practical example how such a hybrid AI solution could increase production efficiency in a thermoelectric power plant and demonstrate that it can also be applied in other production systems. The application of this AI framework resulted in a 25% efficiency improvement in Model 1 (70% efficiency) and a 58% improvement in Model 2 (93% efficiency), outperforming traditional loop control methods currently used for resource optimization in the thermoelectric power plant.

The dissertation has 9 chapters. Chapter 1 is the introduction, and Chapter 2 presents the research problem, delving into the specific challenges and issues faced in the current manufacturing and production landscape. It sets the stage for understanding why the study is essential and the gaps it aims to fill. Chapter 3 outlines the research objectives and questions, clearly defining the goals of the research and the specific inquiries it seeks to

address. This chapter ensures that the reader understands the direction and purpose of the study.

Chapter 4 introduces the theoretical framework, providing the conceptual underpinnings and theories that support the research. This framework helps in understanding the relationships between different variables and concepts studied. Chapter 5 offers a thorough literature review and overview of related works, examining existing research and knowledge relevant to the study. This review helps to contextualize the research within the broader field and highlights how it builds on or diverges from previous work.

Chapter 6 discusses the research methodology, detailing both the methodology and the specific methods used in the study. This chapter provides a clear and transparent explanation of how the research was conducted, ensuring its reproducibility and reliability. Chapter 7 presents the framework for production system optimization, introducing the proposed hybrid AI solution that combines DEA, ML, and GA. It explains how this framework can be applied to optimize production systems.

Chapter 8 illustrates the practical application of the proposed framework through a real-world example in the thermoelectric power generation segment. This chapter demonstrates the framework's effectiveness and provides insights into its practical implications and benefits. Finally, Chapter 9 concludes the research by summarizing the findings and discussing their implications. It also suggests directions for future research, offering a pathway for further exploration and development in the field. This comprehensive structure ensures a detailed and systematic presentation of the research, from identifying the problem to the practical application and conclusion.

2-Research Problem

The drive for improved production efficiency is a constant challenge in the highly competitive world of production systems. Various methods, from lean manufacturing to automating repetitive tasks, are used with a common goal: to reduce waste and streamline production. Thermoelectric energy, which is costly both financially and environmentally, is a key example. Despite its high costs, thermoelectric energy remains in demand due to its unique advantages. It acts as a reliable backup energy source, helping balance the grid without being affected by unpredictable weather. Because of this, it cannot be fully replaced by cheaper, cleaner alternatives. Still, thermoelectric energy producers face ongoing pressure to lower costs and reduce environmental impact.

The core challenge addressed in this research comes from the need to find a solution that can ease these issues and improve the efficiency of thermoelectric power generation. The efficiency of thermoelectric power generation is naturally low, and using technology to enhance it is crucial for its feasibility (Karin Kirk, 2022). In the contemporary era of Industry 4.0, the integration of artificial intelligence into production processes has emerged as a transformative force, AI finds applications in various domains, significantly contributing to cleaner and more efficient production, it excels in tasks such as predicting faulty components or machine malfunctions, and even extends to visual identification of objects, bringing in an era of smarter and more efficient production systems (Rai et al., n.d.). However, when confronted with a complex system that encompasses numerous variables, including inputs and process control parameters (such as temperature, pressure, and the state of various valves), determining the optimal combination of these factors for production efficiency becomes challenge.

The optimization of complex systems is a challenging task due to the interplay of numerous factors and intricate relationships between components. These systems often exhibit nonlinearity, uncertainty, and multiple objectives, making it difficult to find globally optimal solutions within a reasonable time frame. Furthermore, the presence of time delays, a common phenomenon in real-world systems, adds further complexity, as these delay terms can introduce instability and make it difficult to characterize the system's dynamics and predict its future behavior (Le Thi et al., 2019). Moreover, Razzaghi (2009) provides insights into the challenges of modeling and solving optimization problems involving time-delayed systems, the use of hybrid functions, a combination of block-pulse functions and Taylor series, can simplify the solution process by reducing the problem to a set of algebraic equations. However, this dissertation proposal highlights the inherent difficulties of handling delay terms and the intricacies involved in finding exact solutions, particularly for complex systems.

In many cases, expert decision-makers with deep domain knowledge in fields like physics, chemistry, and thermodynamics tackle these complexities, using their experience to fine-tune production processes. However, within the intricacies of such systems, there are subtle disruptions or signals that go unnoticed by human perception. Even the most seasoned professionals or the most advanced expert systems tasked with decision-making struggle with modeling these complex systems. As a result, managing production systems becomes a daily battle, not just to ensure smooth operation but also to achieve peak efficiency by minimizing resource use. In the midst of this complexity and data-rich environment of modern production systems, the rise of Industry 4.0 and the incorporation of AI into manufacturing processes bring forward a unique research question: How can an AI solution be developed to model such complex systems? How can it capture the intricate relationships between inputs, outputs, and state variables? How can it provide recommendations for optimal settings, predict outcomes, and improve efficiencies across

diverse production environments? These questions define the core of the research challenge, opening new avenues where the potential of AI in production optimization is still largely uncharted.

By focusing on a specific production system—the thermoelectric power plant—as the experimental setting for this research, the problem becomes more defined. How can efficiency be increased in such a plant? The lessons learned here could extend to other production systems, potentially signaling a new era in the quest for optimal production efficiency.

3-Research Objective & Research Questions

The following research objective and sub-goals set up specific goals within the research:

Research objective: Extend lean manufacturing and Industry 4.0 capabilities by developing a hybrid AI solution to optimize production efficiency. Development of a hybrid DEA-ML-GA framework for production optimization, application of Digital Twin-based simulations for optimizing thermoelectric energy production. Demonstration of efficiency improvements using real-world sensor data. The solution will be tested with real-world data from a thermoelectric power plant and can also be applied in other fields.

The following research questions set up specific sub-goals within the research objective:

1. What is a suitable way to describe and model a complex system, using inputs, outputs and state variables in the AI context?
2. How DEA, ML, and GA can be combined to create an AI solution that uses sensor data to optimize production efficiency?

3. How much efficiency rise can be achieved in an energy production system if DEA-ML-GA method is used to set production parameters? What may be the limitations of such an AI solution?

4-Theoretical Framework

The theoretical foundations of this research are summarized in key concepts. These concepts include the DEA, ML and GA, AI, Hybrid AI, Production efficiency, Optimization, Digital twins, Machine Learning based Digital Twin and DOE

4.1-Artificial Intelligence, Hybrid AI and Machine Learning

AI is a rapidly changing and broad field. The early 20th-century ideas of AI pioneers like Alan Turing and John McCarthy have grown into the exciting and active field we know today (Pakuhinezhad & Atrian, 2024). The story of AI is one of continuous progress, driven by our desire to replicate and enhance human thinking through machines, from the early days of symbolic AI and expert systems to the major leap into machine learning and deep learning, this evolution has shaped the AI research we're familiar with today (Berente et al., 2021).

AI is built on core ideas that cover a wide range of intelligence-related concepts. At its heart, AI involves creating systems that can understand their surroundings, take action, and make decisions to improve over time (X. Wang et al., 2016). This includes areas like problem-solving, where AI systems use search and planning techniques to achieve goals, and machine learning, where systems learn from experience to make better predictions (X. Wang et al., 2016). Machine learning, which includes supervised, unsupervised learning, semi-supervised learning and reinforcement learning is an important part of AI. It also covers deep learning, which uses neural networks, such as convolutional neural networks

(CNNs) and recurrent neural networks (RNNs), to process large amounts of data and make sense of it (X. Wang et al., 2016). AI systems also deal with natural language processing (NLP), enabling computers to understand and generate human language, and computer vision, where they recognize objects and interpret images and videos (Chinnaiyan et al., 2025). AI combines with robotics to create machines that interact with the real world, performing tasks like autonomous navigation or picking up objects, and these robots rely on AI techniques to sense and plan actions, making them smarter and more capable (Rajakumaran, 2023; Russell & Norvig, 2016). AI is based on a range of methods for reasoning and decision-making, including symbolic AI, which uses rule-based systems to help machines reason, and connectionism, which focuses on neural networks to mimic how human brains work, while Bayesian inference provides AI with the tools to handle uncertainty and make decisions when not everything is clear (Rajakumaran, 2023; Russell & Norvig, 2016).Moreover, reinforcement learning enables AI agents to learn from their actions, using rewards and penalties to figure out the best way to behave in their environment, and together, these elements form the core of AI, making it a powerful tool that continues to grow and evolve with new advancements being made regularly (Rajakumaran, 2023; Russell & Norvig, 2016).

In the changing field of Artificial Intelligence, Hybrid AI is a method that combines different AI approaches to use their strengths and the history of Hybrid AI, like AI itself, shows how it has grown and changed over time as it started with early systems that used rule-based methods and later included other techniques like symbolic AI, expert systems, machine learning, and deep learning, showing how these methods work together to make AI more effective (Ibrahim et al., 2022).

The cornerstone of hybrid AI combines various core concepts that leverage the strengths of different AI paradigms, this approach includes both supervised and

unsupervised learning, ensemble techniques, and the use of neural networks for solving problems, it also blends symbolic reasoning with data-driven methods, providing a comprehensive approach to problem-solving and decision-making (Azizi & Azizi, 2019).

ML is an evolving branch of computational algorithms designed to simulate human intelligence by learning from past data (Nayyar et al., 2021). It is one of the main engines of the new big data era where techniques based on machine learning have been applied successfully in diverse fields ranging from pattern recognition, computer vision, spacecraft engineering, finance, entertainment, and computational biology to biomedical and medical applications (Nayyar et al., 2021). Humans interpret signals and data given by the environment and try to assume certain patterns and predict what is going to happen based on what has been seen in the past. Sometimes, after viewing the data, we cannot interpret it correctly due to the quantity of information and the low calculation capacity of the human brain (Mahesh, 2020). In that case, machine learning can be applied to overcome human brain limitations. With the abundance of datasets provided by the new digital era, and industry 4.0, the demand for machine learning is on the rise (Mahesh, 2020).

Data quality is a major concern in machine learning, as poor data can lead to inaccurate models and unreliable predictions (Budach et al., 2022a). Errors in data can stem from various sources, including measurement inaccuracies, manual data entry mistakes, extreme outliers, missing values, duplications, inconsistencies across different datasets, and labeling errors (Elouataoui et al., 2023). While choosing appropriate data cleaning methods and techniques, such as normalization and outlier detection, is essential, visual inspection can also help identify some issues by allowing practitioners to spot anomalies quickly (Budach et al., 2022). However, visual inspection has its limitations; it can be time-consuming, may miss certain errors in large datasets, and relies on the analyst's perception (Budach et al., 2022). Therefore, it should complement systematic assessments and

automated cleaning methods to ensure high data quality for effective machine learning (Budach et al., 2022b). ML pipelines are formed by first collecting data; making sure that the data is collected most efficiently. Data integrity is one of the most important steps, as invalid data can lead to problematic models, moreover, data preparation is needed, when we deal with missing or erroneous data such as outliers (de Hond et al., 2022). In addition to that in this step, data can be transformed by preprocessing methods if needed (de Hond et al., 2022).At the end of this step the dataset quality is increased and choosing and training the model can be applied to it more efficiently (de Hond et al., 2022). In addition to those points, splitting datasets is needed as it is a common practice in ML algorithms and can produce more robust and unbiased models (Chuang & Keiser, 2018; Kebonye, 2021).

Each model has a different set of hyperparameters that should be tuned to increase the model performance, well known used method for tuning ML models is grid search and grid search is a hyperparameter optimization method that simply does a complete search over a given subset of the hyperparameter space (Cavalcanti et al., 2024). This subset of parameters has a well-known possible range of values for each ML model, and choosing the optimal one is empirical; therefore, grid searching through these possible values is a common practice (Cavalcanti et al., 2022). After all the steps mentioned the model is ready and optimized; hence predictions can be done.

4.2-Production Efficiency and Optimisation

In the dynamic world of industrial processes, the quest for production efficiency involves using various methods to make production more effective and boost overall performance. This pursuit is about finding better ways to streamline operations and improve outcomes in manufacturing and other industries.

Historically, production efficiency has evolved significantly, It began with early industrial engineering practices aimed at improving how things were made, over time, it grew to include modern approaches like lean production, which focuses on minimizing waste, statistical process control for monitoring, improving processes andindustry 4.0 technologies (Dave, 2020; Quiroz-Flores & Vega-Alvites, 2022). Industry 4.0 integrates digital tools and automation into production, the history of Production Efficiency not only tracks the development of these methods but also shows how they work together to achieve better results, by combining these approaches, industries aim to reach higher levels of operational excellence, improving their productivity and effectiveness in a variety of sectors (Dave, 2020; Quiroz-Flores & Vega-Alvites, 2022).

Production efficiency focuses on using various methods together to improve processes and outcomes.lean principles play a key role by reducing waste, improving continuously, and using resources effectively, which aligns with the goal of better production efficiency (Jamwal et al., 2021). Statistical process control supports this by using data to monitor and improve performance while advanced automation and Industry 4.0 technologies like IoT, AI, and robotics bring new tools to make production faster, smarter, and more flexible (Jamwal et al., 2021).

Optimization algorithms are techniques used to find the best possible solution to a problem from a set of potential solutions. They aim to improve performance by either maximizing or minimizing a particular objective function, which represents the goal of the optimization process, for instance, linear programming is a mathematical method used to achieve the best outcome in a model where relationships are linear (Sivanandam et al., 2008). It helps in solving problems such as maximizing profit or minimizing costs under given constraints, this approach is commonly used in fields like economics, engineering, and operations research to make decisions based on linear relationships (Sivanandam et al.,

2008). Moreover, GAs are inspired by the process of natural evolution, and these algorithms use mechanisms similar to biological evolution, such as selection, crossover, and mutation, to evolve solutions over time, and are particularly useful for solving complex problems where traditional methods might struggle (Weise, 2009). GA have the ability to solve non-linear which can be very common in complex problems by simulating the process of natural selection, genetic algorithms can efficiently search for optimal or near-optimal solutions in large and complex problem spaces (Weise, 2009).

In general, optimization algorithms are essential for improving decision-making and performance in various applications by systematically exploring and evaluating different solutions to find the best possible outcome.

4.3-Digital Twins and ML based Digital Twin

A digital twin is a virtual representation of a physical object, system, or process, created by collecting and integrating real-time data from sensors, IoT devices, and other sources, it mirrors the physical counterpart in a digital environment, enabling real-time monitoring and analysis (Segovia & Garcia-Alfaro, 2022). moreover, Digital twins provide a dynamic, detailed, and holistic view of their physical counterparts, allowing for predictive modeling, performance optimization, and decision support (Segovia & Garcia-Alfaro, 2022). Key features of digital twins include their ability to simulate and analyze real-world behavior, provide insights into performance, and support data-driven decisions (Tao et al., 2022). DT are used in areas like manufacturing, healthcare, and urban planning to improve efficiency, reduce downtime, and enhance system performance while driving innovation and better decision-making (Tao et al., 2022). Their development has evolved from early computer-aided design and simulations to modern integration of IoT, big data analytics, and

machine learning, showing how different technologies have come together to create the concept of digital twins that mirror physical entities digitally (Tao et al., 2022).

Digital twins are built upon a foundational set of concepts that fuel their capabilities. These include the creation of virtual counterparts of physical assets, real-time data integration from sensors, and predictive analytics (Jiang et al., 2021). These virtual representations, along with real-time data, enable decision-makers to gain insights, predict future performance, and optimize operations, the core principles underlying digital twins encompass data integration, modeling, and analytics (Jiang et al., 2021).

Users of digital twin technology can gain invaluable insights into the current performance of the entity, predict forthcoming maintenance requirements, and optimize operational efficiencies (Fahim et al., 2022). This iterative data exchange is often facilitated and enhanced through the application of advanced machine learning and artificial intelligence algorithms, which empower the digital twin with the capacity to serve as a powerful tool for enhancing decision-making processes, minimizing operational downtime, and fostering innovation in an extensive array of domains (Fahim et al., 2022). these domains encompass but are not limited to manufacturing, infrastructure management, healthcare, environmental monitoring, and various other sectors where the digital twin's utility proves indispensable (Fahim et al., 2022). On the other hand, ML-based DT mimic the evolution of a complex real system through computational and digital means by the auxiliary combination of ML into DTs (Fahim et al., 2022; Sheuly et al., 2022).

4.4-DOE

DOE is a comprehensive and structured methodology used in scientific research, engineering, and quality management to efficiently plan, conduct, and analyze experiments. It involves a deliberate and systematic approach to varying multiple factors or variables simultaneously to understand their individual and interactive effects on a particular outcome or response, DOE is rooted in statistical principles and is characterized by its capacity to efficiently explore complex parameter spaces and interactions (Jankovic et al., 2021). It allows researchers and practitioners to discern which factors are most significant in influencing a given response and to pinpoint the optimal settings or conditions for achieving desired results (Jankovic et al., 2021).

A central feature of DOE is its ability to generate rich and informative data with a minimal number of experiments, which can significantly reduce resource and time requirements (Román-Ramírez & Marco, 2022). Through statistical analysis, the data obtained from DOE experiments reveal patterns, trends, and relationships that might be otherwise challenging to discern through traditional one-variable-at-a-time approaches. DOE is widely applied in various fields, including manufacturing, product development, process optimization, quality improvement, and scientific research (Román-Ramírez & Marco, 2022). It offers a systematic framework for experimentation that is instrumental in solving complex problems, improving processes, and driving innovation. By using DOE, organizations can make informed decisions based on empirical evidence, reduce waste and inefficiencies, and accelerate product development and process improvement efforts (Román-Ramírez & Marco, 2022).

In essence, DOE plays a pivotal role in facilitating data-driven problem-solving and continuous improvement across diverse industries and disciplines, empowering professionals to make more efficient and effective decisions, ultimately leading to enhanced product quality, process efficiency, and overall performance (Fahim et al., 2022; Sheuly et al., 2022).

4.5-DEA

DEA is a field of interdisciplinary research in operations research management science and mathematical economics (X. Wang & Li, 2022). DEA evaluates the relationship between inputs and outputs of decision-making units (DMUS) determining what is the efficiency of each of them. Furthermore, DEA can be defined as well-established technique to identify the efficiency frontier of multiple input–output systems (Cavalcanti et al., 2024). DEA was initially proposed to evaluate the activities of business entities; however, it can be extended to other problems where multiple input–output combinations can be observed with differing efficiencies, the technique is based on mathematical programming and assumes that there is no noise in the data, if the data have noise, then the approximation of the efficiency frontier may be overstated and then other techniques of the same branch may be used, like SFA stochastic frontier analysis (Charnes et al., 1978).

DEA efficiency judges DMU's location, whether it is on the frontier surface of the production possibility set or if it is in a suboptimal inefficient set. Moreover, DEA is a quantitative method that evaluates relative efficiencies of similar comparable DMUs using linear programming to determine the relationship of multiple input/output parameters. DEA was firstly proposed by Charnes et al. (1978) when DEA-CCR model was proposed. DEA can be performed by 2 different models, the DEA-CCR model (Charnes et al., 1978) and

DEA-BCC model (Banker et al., 1984), where DEA-BCC should mostly be applied when DMUs are measured in heterogeneous conditions and scaling.

The DEA-CCR model is associated with a limitation in that it considers the return to scale as constant and cannot discern technical efficiency from that of scale efficiency, due to this reason, if the production is variable return to scale oriented, the results obtained may picture some efficient DMUs as inefficient, hence an efficient DMU in BCC may come as inefficient in CCR model, however, the opposite is not true (Mustafa et al., 2021). To counter this issue, DEA-BCC was created considering variables return to scale additional block fundamentals. CCR model can be described by the equation **1** (Charnes et al., 1978).

$$\begin{aligned} &\boldsymbol{\theta *= min\theta} \\ &\text{s.t.} \sum_{\boldsymbol{q=1}}^{\boldsymbol{n}} \boldsymbol{x_{pq}\lambda_q \leq \theta x_{po} p = 1, 2, 3, \dots, k} \\ &\sum_{\boldsymbol{q=1}}^{\boldsymbol{n}} \boldsymbol{y_{rq}\lambda_q \geq y_{ro} r = 1, 2, 3, \dots, l} \\ &\boldsymbol{\lambda_q \geq 0 r = 1, 2, 3, \dots, l} \end{aligned} \qquad (\mathbf{1})$$

Where $\boldsymbol{x_{po}}$, $\boldsymbol{y_{ro}}$, are the pth input and rth output of the DMU o under study;

$\boldsymbol{\lambda_q}$ stands for those variables that indicate the effect of prominent factors on efficient DMUs, which other DMUs use as references to compare their efficiency with.

$\boldsymbol{\theta *}$, is the comparative technical efficiency of DMU o

On the other hand, the formula for DEA-BCC has a small change on its formula by the inclusion of constant $\boldsymbol{u_0}$ in the objective function and in one of the constraints,

considering the variables return to scale with additional block fundamentals, DEA-BCC formulas are shown by the following equation **2** (Banker et al., 1984).

$$\begin{aligned} &\boldsymbol{max h_0 = \sum_{r=1}^{s} u_r y_{r0} + u_0} \\ &\text{s.t} \sum_{r=1}^{s} \boldsymbol{u_r y_{rj}} - \sum_{i=1}^{m} \boldsymbol{v_i x_{ij} + u_0 \leq 0, j = 1, 2, 3, \dots, n} \\ &\sum_{i=1}^{m} \boldsymbol{v_i x_{i0} = 1} \\ &\boldsymbol{u_r, x_{i0} \geq \varepsilon, \forall, r, i} \end{aligned} \quad (\mathbf{2})$$

Since the DEA method and its model were proposed it has been widely used in different industries(Cavalcanti et al., 2022). By comparing the efficiency of a specific unit with the performance of a group of similar DMUs, it is possible to create a rank of the most efficient DMUs, which can be later investigated, then managers can try to increase the efficiency of inefficient DMUs by coping with the behavior of the most efficient ones (Cavalcanti et al., 2024).

4.6-Genetic Algorithms

GA, a metaheuristic algorithm, is a well-known algorithm which is inspired by the biological evolution process (Nguyen et al., 2020). GA mimics the Darwinian evolution theory, based on the survival of the fittest individuals. GA was proposed by J.H. Holland in 1975 (Holland, 1975). The basic elements of GA are chromosome representation, fitness selection, and biological-inspired operators, the biological-inspired operators are selection, mutation, and crossover, in selection, the chromosomes are selected on the basis of its fitness value for further processing (Nguyen et al., 2020). In crossover operator, a random locus is

chosen, and it changes the subsequences between chromosomes to create off-springs, there are multiple different methods on how to do chromosomes' crossover methods, such as Single Point Crossover, Two-Point Crossover, Uniform Crossover, varying the programming of a chromosome or chromosomes from one generation to the next, furthermore, different selection methods are also available, such as selecting only the strongest and then crossover themselves, best-worst selection criteria and rank-based selection (Hussain et al., 2017).

GA final step, "mutation", similarly to what happens in nature, generates random offspring, creating a "mutation" on it. Mutations are one of the key engines of genetic algorithms since they maintain the genetic diversity from one population to the next population (Katoch et al., 2021). The stop criteria is then checked and if met GA stops otherwise the GA cycle is reinitiated.

The stop criteria in a genetic algorithm are a vital component, serving as the traffic lights of the optimization journey. These criteria define the conditions under which the algorithm should halt its search for an optimal solution. Their importance lies in ensuring the algorithm balances exploration and exploitation, preventing it from running indefinitely. Common stop criteria include a predefined number of generations, a threshold on the fitness improvement, or convergence checks. For instance, when applying a genetic algorithm to optimize a complex engineering design, one might set the stop criteria to terminate the algorithm after a certain number of iterations or when the improvement in the fitness function becomes marginal. Carefully selecting these stop criteria is crucial for achieving efficiency and ensuring that the algorithm converges to a satisfactory solution within a reasonable timeframe, saving computational resources and time in the optimization process. Figure 1 summarizes the GA process.

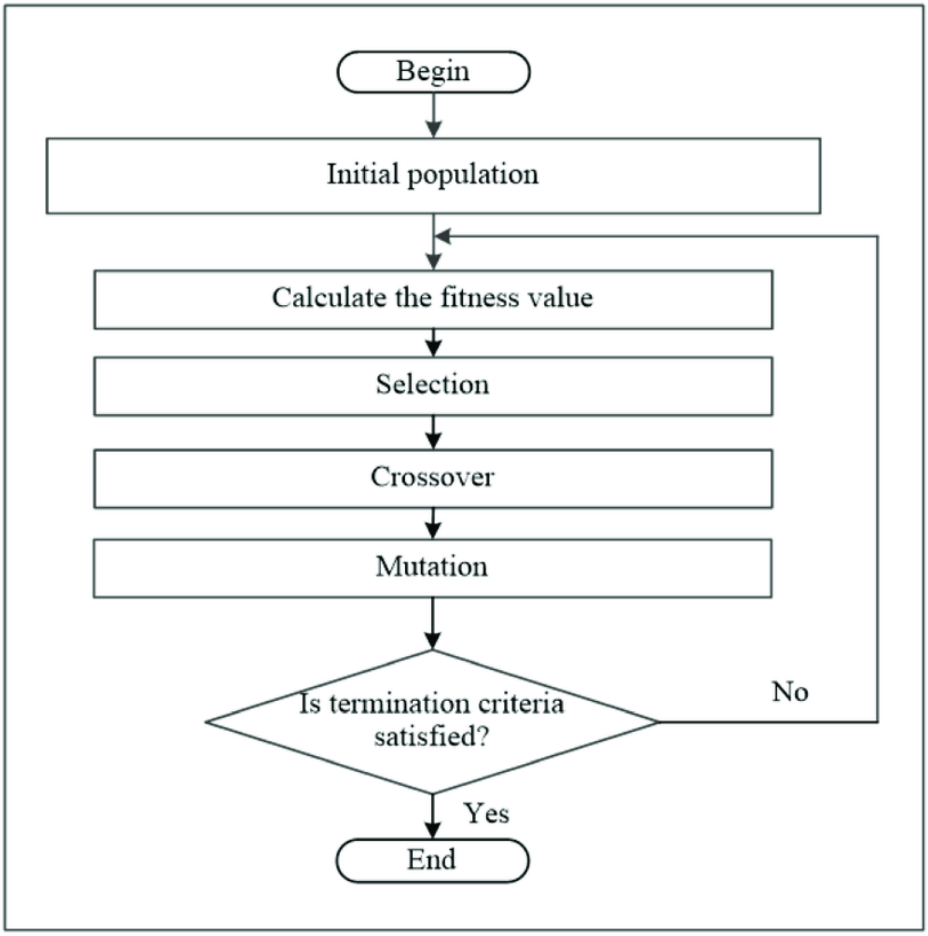

Figure 1 - GA process (Nagunwa, 2024)

4.7-Feedback Control Loops

Feedback control loops are fundamental systems designed to regulate and maintain desired conditions or setpoints within dynamic processes. They serve as the vigilant overseers of processes, continuously monitoring and adjusting various parameters to ensure that the system operates as closely as possible to its intended state. Whether in manufacturing, engineering, healthcare, or countless other fields, feedback control loops play a pivotal role in maintaining precision, stability, and efficiency (Guzmán & Hägglund, 2024).

At the core of a feedback control loop are four essential components. The process is the system or entity whose performance needs to be controlled, It could be a manufacturing, a chemical reactor, or even a medical infusion system, and produces outputs that are subject to variations and disturbances (Bahreini et al., 2021; K. Xu & Pérez-Arancibia, 2020). Sensors are responsible for measuring relevant variables or conditions of the process, such as temperature, pressure, speed, or any parameter of interest , these sensors provide real-time data that characterizes the current state of the process (Bahreini et al., 2021; K. Xu & Pérez-Arancibia, 2020). The controller is the decision-maker in the feedback loop, It receives input from the sensors, compares this data to the desired setpoint or reference value, and determines the necessary corrective action (Bahreini et al., 2021; K. Xu & Pérez-Arancibia, 2020). the controller then calculates the control signal required to bring the system back to the desired state (Bahreini et al., 2021; K. Xu & Pérez-Arancibia, 2020). finally the actuators are responsible for executing the control signal generated by the controller, they manipulate the system by adjusting parameters like valve openings, motor speeds, or any other relevant variables, and bring the process back in line with the desired setpoint (Bahreini et al., 2021; K. Xu & Pérez-Arancibia, 2020).

The operation of a feedback control loop can be understood through the following steps:

- Measurement: Sensors continuously collect data on the current state of the process.
- Comparison: The controller compares the measured data to the desired setpoint. If there is a discrepancy, it calculates the necessary control action.
- Control Action: The controller sends a control signal to the actuators, instructing them to adjust.
- Correction: The actuators implement the control signal, modifying the process and steering it back toward the desired setpoint.
- Reassessment: The process is continually monitored, and this cycle repeats, ensuring that the system remains within the desired performance parameters (Borase et al., 2021).

Feedback control loops are versatile, adaptable, and vital in applications where precision, stability, and efficiency are paramount. They can maintain a constant temperature in an industrial furnace, stabilize the flight of an aircraft in turbulent conditions, regulate the flow of medication in a healthcare setting, or even control the position of a robotic arm in manufacturing. Modern feedback control loops have been enhanced with advanced technologies, enabling them to incorporate predictive algorithms, data analytics, and machine learning. These innovations enable systems to anticipate disturbances and make real-time adjustments more intelligently and accurately. Feedback control loops are the silent guardians of countless processes across industries. They ensure that systems remain on course, adjusting as necessary to maintain precision and stability (Sanfilippo et al., 2020; J. Zhou et al., 2021).

The PID controller, a type of feedback control loop method, with its three core components – proportional, integral, and derivative – operates much like a conductor leading an orchestra. The proportional part instantly responds to any deviations between the desired and actual process conditions, swiftly triggering corrective actions. The integral component addresses any lingering errors that accumulate over time, ensuring a steady and accurate state. Meanwhile, the derivative component anticipates and mitigates sudden changes, adding an element of stability to the system. These components work in harmony to create a symphony of control, ensuring that the many moving parts of a process work seamlessly together. These control models are not limited to one industry but are versatile and can be applied across diverse domains. From regulating temperature in chemical processes to guiding the movements of robotic arms in manufacturing, they serve as the silent but essential force behind efficient and precise operations (Jeng & Lee, 2023; Somefun et al., 2021). However, PID control loops, despite their widespread use, have limitations, particularly when dealing with time-delayed systems. The presence of time delay makes the system's characteristic equation a quasi-polynomial with an infinite number of roots, making stability analysis much more complex. The classical tuning methods for PID controllers, like Ziegler-Nichols, are often inadequate for time-delayed systems (Ziegler & Nichols, 1942). The key difficulty lies in tuning the integral (I) and derivative (D) parameters. While the proportional (P) gain can often be set based on the delay-free system, the I and D gains become more sensitive to the time delay. With an increase in time delay, the stabilizing region in the (kd, ki) parameter space shrinks, often becoming a complex combination of polygons or even the limit of an infinite sequence of polygons. The complexity arises from the fact that larger time delays create more stability boundaries in the (kd, ki) plane, making accurate determination of the entire stabilizing region a challenging task. These factors make designing robust and non-fragile PID controllers for time-delayed systems a significant challenge (Hohenbichler, 2009; Silva et al., 2007).

5-Literature Review

5.1- Literature Review Using Scopus

My proposed AI solution uses DEA, ML and GA. DEA is a widely utilized, well-established nonparametric method for estimating the relative efficiencies from observed, historical data by using linear programming to identify the distance of each DMU to the frontier, composed of a set of the most efficient (Ahmed et al., 2019). Moreover, ML is a multidisciplinary subject that combines knowledge of probability theory, statistics, approximation theory, convex analysis, and algorithm complexity theory (Cavalcanti et al., 2024). ML specializes in how computers simulate or implement human learning behaviours to acquire new knowledge or skills and reorganize existing knowledge structures to continuously improve their performance (Zhu et al., 2020). GA is the last component of the proposed hybrid AI solution. GA is used for optimization and finding the best set of variables that maximizes or minimizes certain output (Al Bataineh et al., 2022). The combination of those methods can create ML-based digital twins capable of running optimization problems, and optimizing the resource usage of production systems.

In the literature review I examined the Scopus database as one of the most important scientific databases for the analysis of relevant scientific articles. Data collection was performed in September 2024 focusing on the previous ten years of articles. The present study used the following keywords to search for the relevant research outputs: "production efficiency", "DEA", "machine learning/ML", “Genetic algorithms/GA” and “digital twins/DT”. I did not obtain any paper containing all keywords, neither anything was found when searching for “DEA”, “GA,” “ML” and “production efficiency”. When searching for DEA, ML and GA 16 manuscripts were found and when searching for “DEA”, “ML” and “production efficiency” 7 documents were found. Finally searching for “digital twins”,

“production efficiency” and “ML” gave 15 results. All the found research is very recent and the oldest are from 2020, with the majority of them published in 2024 and 2023, demonstrating that the topic is new and belongs to uncharted territory. The found documents have some parts involving overlapping methods but none of them uses those for production efficiency optimization thru ML-based digital twins nor combine ML, GA, DEA and DT for such purpose, making this research unique. When keywords were limited to two or three of the previously mentioned, Engineering was the main subject domain area, followed by computer science, business and management and environmental science also. Figure 2 shows the trend on total articles published for the combination of keywords mentioned, a clear uptrend can be seen, reinforcing the relevance growth of the topic.

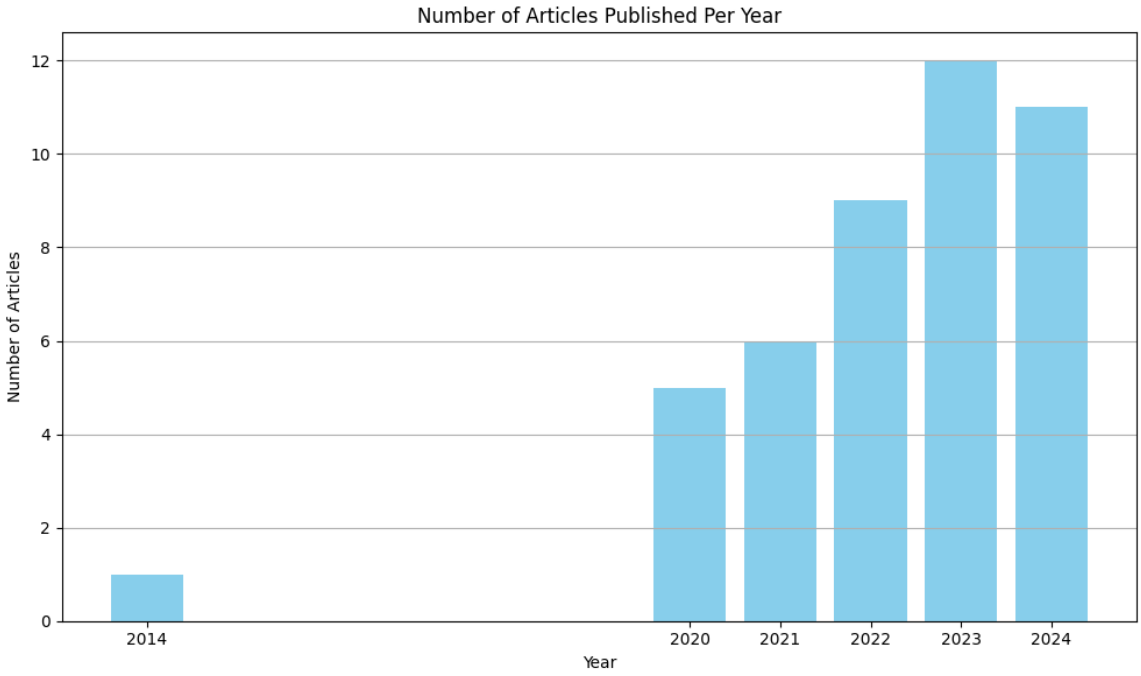

Figure 2 - Trend of publications over the years of 2014 to 2024

The top articles (based on the number of citations) were reviewed and used as a background for this research. This collection of research articles presents a diverse range of applications and methodologies related to intelligent manufacturing systems, primarily within the context of Industry 4.0. Several common themes emerge: the importance of data-driven decision-making, the integration of digital twin technology, and the use of ML techniques for optimization and prediction. C. Liu et al. (2022) proposes a comprehensive framework for collaborative data management in metal additive manufacturing (AM) using a Cloud DT and Edge DTs. It highlights the need for a standardized data model to capture critical data across the AM lifecycle emphasizing data management, a key aspect of this research. Y. Wang et al. (2022) presents a DT model for smart factory production systems, incorporating ML techniques for production prediction and process modeling, exemplifying the application of ML within a DT framework, directly aligning with this research. However, it focuses on predicting production outcomes, while my work emphasizes resource optimization. Expanding DT and ML application into production efficiency. D. Zhang and Gao (2022) talks about a digital twin system designed to improve how chemicals are added in the iron reverse flotation process, which is important for extracting minerals. It uses machine learning and real-time images of the froth to determine the right amount of chemicals needed. A special model monitors the quality of the output, while another model automatically adjusts the chemical dosing. This system helps ensure better product quality, reduces waste, and boosts efficiency and profits in production.

Zhu et al. (2021) combines DEA with four ML algorithms (BPNN-DEA, GANN-DEA, SVM-DEA, and ISVM-DEA) to predict the efficiency of Chinese manufacturing companies. It has some similarities with this research by integrating DEA and ML algorithms for predicting efficiency, however it focusses on financial efficiency, while this thesis targets resource optimization. Taherinezhad and Alinezhad (2023) utilizes a combined DEA and ML approach to evaluate the performance of nations during the

COVID-19 pandemic, similarly Kheyri et al. (2023) develops a DEA-based benchmarking model for car after-sales service agencies and uses ML to predict their performance. Moreover, Diker et al. (2021) combines DEA and ELM for ECG signal classification. This work utilizes DEA and ML for classification, using deep feature extraction with CNN and using DEA to optimize ELM parameters. those manuscripts align with this research by combining DEA and ML, however, focuses on completely different topic than production optimization. Extending these concepts, Yan et al. (2022) introduces a three-stage Super-SBM DEA model coupled with the Extra-Trees algorithm. This model evaluates grain production efficiency (GPE) in the Hexi Corridor, analyzing factors like human-driven and nature-driven influences, which are pertinent to optimizing resource use, similar to the focus of this thesis.

In addition to the articles listed above, Kostov and Hristov (2023) explores optimizing the cycle time of industrial robots used for loading molding machines to boost production efficiency and profitability. It identifies key factors like robot movement, tool selection, part handling, and machine setup, and proposes strategies such as optimizing trajectories, improving gripper designs, and using machine learning. Simulation models and digital twins are highlighted for testing and predicting improvements.

Soori et al. (2024) examines the key aspects, benefits, and challenges of integrating intelligent robotic systems in Industry 4.0. This review provides a broader context for this research, highlighting the role of robotics and AI in Industry 4.0. It covers several areas relevant to this work, including connected automation, collaborative robotics, adaptive manufacturing, and predictive maintenance.

Further contributions to resilience and efficiency optimization come from a range of studies on energy systems that provide valuable insights into enhancing system robustness,

especially under challenging conditions. For instance, Hossain et al. (2021) explore various strategies aimed at bolstering the resilience of power grids in the face of extreme weather events. it focuses on identifying key vulnerabilities within the grid and proposes methods to ensure reliability and continuity of service during natural disasters, offering a framework for how critical infrastructure can be safeguarded against disruptions. Similarly, Martišauskas et al. (2022) introduce a comprehensive set of quantitative indicators that can be used to evaluate the resilience of energy systems. It provides a systematic approach to measuring resilience, allowing for a more objective assessment of how well energy systems can withstand and recover from disruptions. Further, H. Liu et al. (2022) studied sequential, preventive measures designed to strengthen power systems against the cascading failures often triggered by extreme weather. This model prioritizes strategic, preemptive actions to minimize damage, which parallels the resource allocation strategies used to optimize manufacturing processes by anticipating and mitigating potential disruptions before they escalate. Additionally, Q. Zhang et al. (2021) focuses on enhancing resilience through stochastic modeling that prepares distribution systems for potential failures before they occur. This study underscores the importance of uncertainty modeling and pre-event preparedness, which is analogous to how manufacturing systems might optimize their processes by allocating resources in anticipation of potential bottlenecks or disruptions. Both approaches emphasize the critical role of planning and preemptive action in achieving resilience and efficiency. Moreover, Judge et al. (2022), Umunnakwe et al. (2021) and C. Wang et al. (2022) all discuss the implementation of smart grid technologies and power system resilience. They examine frameworks, performance, and key improvement areas, aiming to enhance understanding and achieve more resilient energy systems.

Eskandarpour et al. (2017), Raza and Khosravi (2015) and Z. Wang et al. (2022) focus on enhancing resilience through advanced modeling and predictive techniques. These manuscripts address disaster preparedness and energy management, seeking to develop

strategies that improve system reliability. Montoya-Rincon et al. (2023), J. Wang et al. (2021) and Xie et al. (2020) explore innovative technologies and frameworks for improving power system resilience. They aim to integrate various methods, including machine learning and socio-technical assessments, to enhance decision-making and infrastructure reliability.

These energy-focused studies, while centered on infrastructure resilience, offer valuable parallels and insights into the optimization and preventive strategies relevant to manufacturing and other complex systems. They illustrate the need for proactive measures, strategic planning, and optimization techniques to improve resilience, which are also at the heart of the research being discussed.

5.2- Related Works

When looking closely at how methods used in this research were used in the past, I selected the most relevant articles. . Hafeez et al. (2020) use deep learning to forecast electric load and heuristics-optimized ML model tuning. The model offers guidance for energy production systems by giving maximum energy generation capabilities, therefore avoiding waste of energy that cannot be stored or dispatched to the grid. Similarly, Wen et al. (2019) and Y. Zhou et al. (2020) aim to predict photovoltaic power output by using ML methods and a GA to improve the prediction accuracy of the model, hence enabling the grid administrator to prepare the grid for energy loads and avoiding the waste of energy that has no grid or storage to be dispatched to. On the other hand, Merei et al. (2013) optimized production using GA by changing the component sizes and model settings of solar/wind/diesel energy production systems with different battery technologies. The results indicated that optimization is possible and economical.

Furthermore, Król and Ocłoń (2018) measured costs and energy efficiency for heat and energy generation by evaluating different approaches for combined heat and power plants in Poland. In this kind of power plant, the steam that would be wasted is allocated for another purpose. In this case study, the cost efficiency of the combined heat and power plant is improved if natural gas engines are used. propose a method for the selection of input variables by applying DEA to measure the energy efficiency of a chemical process. Moreover, the approach of X. Xu et al. (2019) was aimed at reusing heat in a solid-state thermoelectric generator (TEG), converting the waste heat to electricity using the Seebeck phenomenon. ML models have been widely utilized in the modelling, design, and prediction of energy systems. Ten major ML models are frequently employed in energy systems: Artificial Neural Network (ANN), Multi-Layer Perceptron (MLP), Extreme Learning Machine (ELM), Support Vector Machine (SVM), Wavelet Neural Network (WNN), Adaptive Neuro-Fuzzy Inference System (ANFIS), Decision Trees (DT), Deep Learning (DL), ensembles, and advanced hybrid ML models (Mosavi et al., 2019). However, the usage of a hybrid AI solution that combines DEA, ML-based simulation, and a GA to optimize energy production systems efficiency was not included in the literature.

Combining ML with GA and DEA with ML has been widely discussed in the literature. DEA and ML were selected to measure and predict efficiency in manufacturing, finances, and supplier selection (Cheng et al., 2017; Hong et al., 2019; L. Liu et al., 2019; Salehi et al., 2020). Furthermore, a GA was integrated into ML models to identify optimal features/hyperparameters (Badnjević et al., 2019; Di Noia et al., 2020; Ko et al., 2017). ML was utilized within the GA to explore advantages in accelerating searches. With a complex fitness function with high calculation costs, using an ML model as a fitness function could reduce the computing time.

5.3-Sytem Control and Feedback Loop Method

Feedback loop control methods, such as the proportional-integral-derivative (PID) controller method, may not be sufficient for controlling more complex processes, especially if they are nonlinear systems (Cavalcanti et al., 2024). Modern control theory heavily relies on the state space representation, where a control system is described by a set of inputs, outputs, and state variables connected by a set of differential equations (Y.-Y. Liu & Barabási, 2016). The state space model can be described as the state and output equations, as shown in Equation 3 (Cavalcanti et al., 2024).

$$\begin{cases} \dot{x}(t) = f(t, x(t), u(t);\ \theta) \\ y(t) = h(t, x(t), u(t), \theta) \end{cases} \qquad (3)$$

Where the state vector x(t)∈ R^N represents the internal state of the system at time t, the input vector u(t)∈ R^R captures the known input signals, and the output vector y(t)∈ R^R captures the set of experimentally measured variables (Cavalcanti et al., 2024). The functions f(·) and h(·) describe the behaviour of the complex system, and Θ collects the system's parameters. Our method is aimed at approximating the function h(·), where the output y(t) is estimated using data envelopment analysis (Cavalcanti et al., 2024). Feedback control loops, such as PID controllers that continuously calculate an error and apply a corrective action to minimize the error, will be used (Chia, 2018; Moshayedi et al., 2019), and at lower levels of the system, controlling simple processes. PID can have a lot of arbitrariness, reducing such arbitrariness with replacing methods that address such issues, can raise production efficiency. For instance ML-based DT using DEA, GA and ML is a potential approach for replacing PID.

Linear programming and simple heuristic approaches are often inadequate for solving complex optimization problems, particularly in nonlinear and dynamic environments. As a result, PID controllers have become a standard alternative, offering

effective control without requiring a detailed mathematical model of the system. PID has been successfully implemented in various sectors, including thermoelectric distillation (Nasir et al., 2022) cooling systems (Koç & Bayhan, 2024), thermal management using Arduino-based platforms (Kherkhar et al., 2022), and efficiency improvement in thermal power plants (Shajahan et al., 2022).

This thesis challenges the widespread reliance on PID control by introducing a novel approach that integrates DEA, GA, and ML, aiming to provide a more adaptive and efficient solution for complex optimization problems. To assess the performance of the proposed AI-based optimization framework, PID control was used as a baseline for comparison. This allowed for a clear evaluation of efficiency gains and demonstrated the potential of the new method to outperform traditional control strategies in handling system complexity.

5.4-ML-based DT

ML-based DT, a new term, has received limited, trendy research attention, signifying its relative novelty in academic discourse. According to Sheuly et al. (2022) There is a notable absence of a comprehensive systematic literature review encompassing various facets of ML-based DT within the manufacturing industry, approached from both bibliometric and evolutionary perspectives.

Precisely defined, a DT represents a mirrored image of a physical process intricately synchronized with the associated process, replicating the operations of the physical counterpart in real-time. The term was initially coined in the early 2000s by Grieves (2014) , whose expertise in product design initially grounded the concept within production engineering. Since its inception, however, the concept has undergone a broadening and a certain degree of flexibility. It is now utilized to characterize a diverse array of digital

simulation models that run concurrently with real-time processes, extending beyond physical systems to encompass social and economic systems, for instance Agrawal et al. (2022) refer to a hierarchy of sophistication levels in DT, which represent different degrees of complexity and capabilities. The hierarchy is designed to guide the selection of an appropriate level of sophistication based on specific needs.

- Descriptive Level (or Description). At this level, the DT provides information in a useful form. It serves as a representation of the physical entity, offering a visual or descriptive replica.

  Diagnostic Level (or Reasoning). Moving up the hierarchy, the DT goes beyond simple representation. It analyzes data to understand and provide insights into why certain events or conditions are occurring.
- Predictive Level (or Prediction). At this level, the DT utilizes operational data to predict possible future outcomes. It goes beyond understanding the current state to forecasting potential developments.
- Prescriptive Level (or Prescription). The highest level involves decision-making. The Digital Twin not only predicts future outcomes but also suggests or decides on appropriate actions based on predefined objectives and preferences. This hierarchy allows users to choose the level of sophistication that aligns with their specific requirements.

Agrawal et al. (2022) discusses challenges in deploying DTs amid technological advancements, especially in sophisticated capabilities like AI and ML, the lack of research has led to the indiscriminate use of AI and ML in DTs, potentially causing missed opportunities and strategic misalignments, to address this, selecting the appropriate level of sophistication in a DT by weighing the pros and cons of AI and ML integration, establishing evaluation criteria, and assessing the implications on organizational processes. Hence, the

proposed AI framework uses the same approach for systematically integrating AI and ML capabilities into DT, providing meaningful results.

In traditional DT, the virtual model is often created to mimic the behavior of a physical system, and real-time data from the physical system is used to update and refine the virtual model, however, in some cases, it may be impractical or costly to have continuous real-time data streaming between the physical system and the virtual models, in this case, ML can be used to create simulations or digital twins that are not reliant on real-time data streaming between a real system and its virtual model, in such situations, ML techniques can be employed to create simulations or ML-based DTs without relying heavily on real-time data, those kinds of DTs when combined with GA as proposed in this research can achieve the most sophisticated levels a DT can reach, the prescriptive level, being capable of predicting future states and giving high quality suggestions for decision making process (Batty, 2018; Botín-Sanabria et al., 2022; VanDerHorn & Mahadevan, 2021)

Some research endeavors have delved into the intersection of ML, DT, and optimization. Y. Liu et al. (2023), for instance, created a ML-based DT aimed at predicting the remaining life of offshore floating wind turbines, proposing it as a predictive maintenance strategy. Meanwhile, Fahim et al. (2022) introduced a novel framework for processing temporal data streams to forecast wind speed and predict generated energy. Notably, those work did not explicitly touch upon optimization and the associated costs of data streaming. Moreover, Dong et al. (2019), He et al. (2023) and Yang et al. (2022) explored the utilization of machine learning and digital twins for optimization problems. Adding to this discourse, Min et al. (2019) proposed a ML-based DT Framework for Production Optimization in the Petrochemical Industry. However, its optimization mechanics focuses on portfolio management derived from market demand data rather than considering production inputs and process control recommendations for reducing

production waste. Additionally, the framework relies on real-time data and petrochemical-specific optimization methods. Notably absent from the discussion are simulation and optimization systems, such as generic optimization algorithms like GA. Moreover, the differences between the existing mentioned researches and the proposed framework lie in the unique approach of the latter.

5.5-Research Gap

This research focused on creating a machine learning-based digital twin to optimize production system resources, builds on ideas from previous studies. Many of these works highlight the potential of digital twin technology to handle large amounts of data, simulate complex processes, and enable real-time monitoring and control, things that are directly relevant to this project. Several papers also show how combining machine learning with DEA can help predict efficiency and improve processes, which can be applied here for resource optimization. What sets this research apart from existing studies is its focus on resource optimization within production systems. While many papers deal with predicting efficiency, this study tackles the challenge of improving how resources are used. Additionally, rather than just applying existing methods, this research aims to create a new machine learning-based digital twin model designed specifically for resource optimization, pushing the current knowledge in this area forward. As previously mentioned, the approach of combining DEA, ML and GA was never implemented according to my literature review.

Traditional optimization techniques, such as linear programming and simple heuristic approaches, often fall short when applied to these types of problems, as they cannot fully address the complexity of how system variables interact in real-time. This is where PID controllers became the dominant standard. PID offers a practical and robust solution

that does not require a complete mathematical model, making it widely adopted across multiple sectors for managing complex processes.

Nevertheless, this research introduces an alternative approach that challenges the dominance of PID control. By integrating DEA, GA, and ML, this thesis proposes a new framework capable of learning and adapting to complex system dynamics, potentially offering superior performance in environments where conventional control strategies may be limited. In summary, this research aims to fulfil a literature gap by introducing an innovative AI Framework for Production Efficiency Optimization Using ML-Based DT Simulation. This framework is designed to be adaptable to various production systems, providing guidelines and examples on practical implementation. The introduction of this new method can lead to a new research line uncharted by the literature and explored in this thesis.

6-Research Methodology and Methods

6.1-Research Methodology

Considering this research's technological character, the use of a methodology that would meet the creation of artefacts aimed at solving the problem outlined seemed appropriate. The epistemological basis for this research was found in the Design Science (DS) to ensure a scientific approach. Design science is a science that designs and develops solutions for problems by creating or improving artifacts that improve human performance, both in society and in organizations, to make sure that the DS methodology fulfils its role, it uses the Design Science Research (DSR) method defined as research that uses design as a method and/or technique. DSR seeks to enhance technology and scientific knowledge via innovative artifacts that solve problems and improve the environment (Moura Júnior, 2021).

DSR has well-defined phases problem awareness, suggestions, development and evaluation. An extra final step is communication when findings and results are published in professional scholarly journals (Dresch et al., 2015; Moura Júnior, 2021; vom Brocke et al., 2020). Figure 3 shows in more detail the implementation of DSR steps in this research.

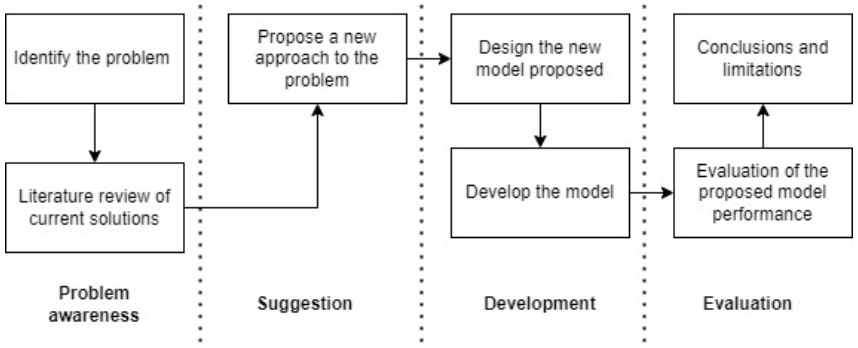

Figure 3 - Implementation of DSR steps in this research

First, the problem was identified by defining the root of the research: production systems need improvement, and increasing performance and efficiency remain a persistent challenge for managers and businesses. More specifically, complex manufacturing systems are difficult to model and optimize due to their nonlinear behavior, interdependencies, and sensitivity to external disturbances. The main approach used for modeling such systems is through modern control theory, which relies on state-space representation. In this framework, a system is described by a set of inputs, outputs, and state variables connected by differential equations (Cavalcanti et al., 2024). However, using differential equations to

describe these systems may overlook various factors and dynamic interactions that are not easily captured by mathematical formulations.

(Cavalcanti et al., 2024)By identifying the problem and being aware of the background and concerns previously discussed by other researchers, a new approach is suggested instead of describing a complex production system with differential equations that explain each and every variable of the system, an AI solution could be capable of learning how the system have behaved in the past and predict how it is going to behave in the future in different circumstances, eliminating the need for mathematical definition of the complex production system. This new approach not only simplifies how to describe production systems, but it can also include variables not included in theoretical differential equations. Another benefit is that the new model can be extended to different production systems without the need for domain knowledge about their production process. Following the suggestions, during the development phase, the new model capable describing complex production systems was designed. The new approach is summarized on figure 4 about how to model complex production system was suggested where outputs of the system are predicted based on inputs, and process control parameters with the usage of machine learning models (Cavalcanti et al., 2024). The model is trained by having as a dependent variable previous output generation values and as dependent variable the inputs and process control parameters (Cavalcanti et al., 2024). Moreover, the efficiency of the system can also be predicted by machine learning models based on inputs, outputs, process control parameters, and efficiencies achieved in the past, on this second model, on the other hand, inputs, outputs and process control parameters are the independent variables and the efficiency is the dependant variable (Cavalcanti et al., 2024). Figure 4 gives an overview of how to model a complex system, without the usage of differential equations and domain knowledge. Modelling complex systems using differential equations and domain knowledge doesn’t seem to be enough, as the implementation of such methods is expensive, and

inefficiency is still identified when analyzing production systems, therefore the usage of AI can potentially improve the production process modelling and later on raise production efficiency by the usage of optimization algorithms.

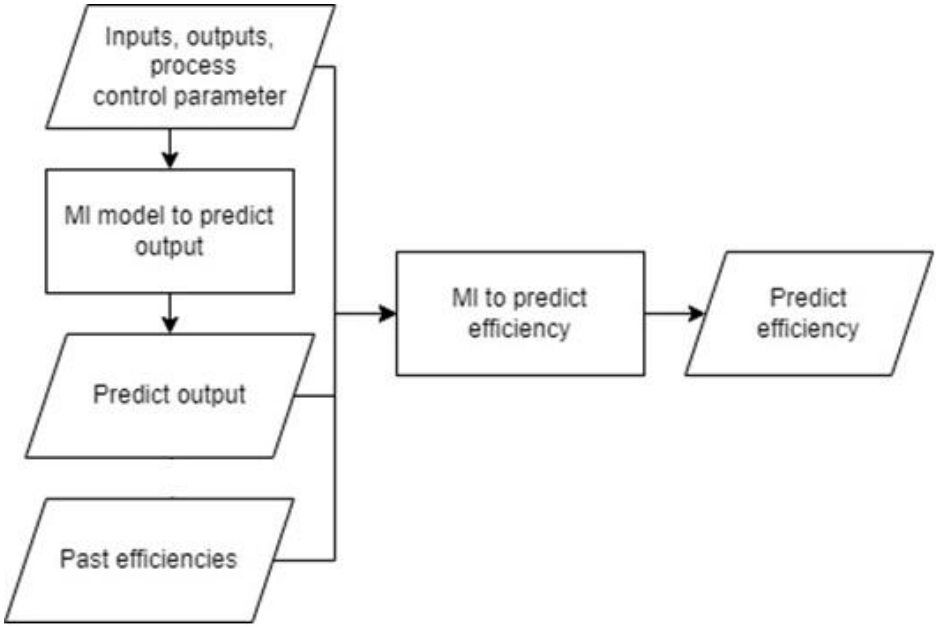

Figure 4 - Complex system modelling

After describing the model, the main goal is to optimize production efficiency, and this can be done by the usage of operations research techniques. Genetic algorithm a metaheuristic optimization algorithm , that is capable to optimise highly nonlinear systems was applied to search for the best production setup by simulating random setups, measuring its output and efficiencies and selecting the solution which fits best for the occasion, based on the predefined minimum output. Figure 5 shows the optimization process of the production on user request.

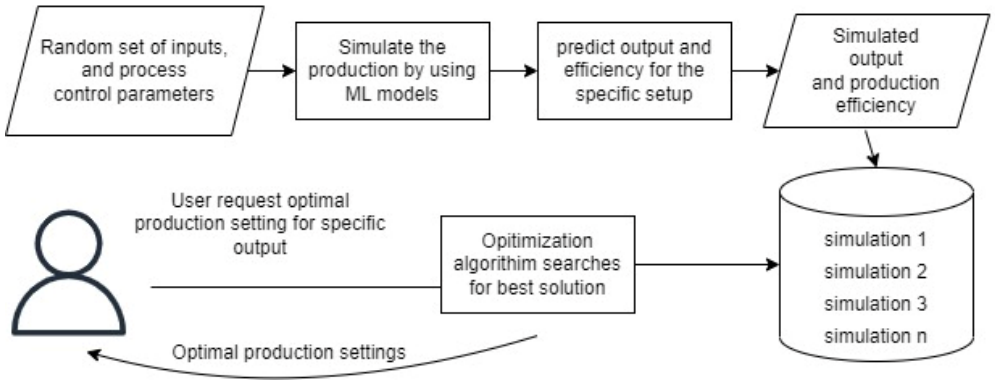

Figure 5 - The optimization process of the production on user request

A random set of inputs and process control parameters is generated, using this, the modelling of the production system previously described can generate simulations of production setups and store them wisely in memory, the optimization algorithm will look through all these simulations and try to select the one that fits better the user conditions.

After the artifact design the next step of the DSR, the development of the artifact, was done. During development, the application life cycle used software engineering methods, which Sommerville (2020) defines as an engineering discipline concerned with all aspects of software production and has scientific rigour due to its systematic approach called the software process that has well-defined steps to produce systems: specification, development, validation and evolution. The first defines what must be done and how it will work, the second encapsulates design and programming, the third seeks to ensure that what has been recommended is what was done and is properly functional and finally the last focuses on scalability and future possibility of modifications according to new needs.

For this research, the agile methodology called SCRUM was used. The SCRUM methodology has three phases: planning, sprint cycle and closure (Schwaber, 1997). In the planning stage, objectives are established according to the architecture of the project, in addition to that, a list of the activities to be developed during the timeframe of the sprint is selected, this list is called backlog. The sprint cycles are equivalent to a version of the system, they last from 2 to 4 weeks and contain a set of previously selected features to be developed. At the end of the cycle, the new version is presented, and a new sprint is triggered. And finally, the project terminates when there are no more items in the backlog. Considering the developers, the team were made up of just one person, myself, who developed new features and wrote code, the management of the SCRUM team and the product owner activities were shared between the thesis supervisors. Following scrum guidelines, at the end of 2 2-week sprint the achievements and goals for the next sprint were discussed with the thesis supervisor, who had domain knowledge about the studied production system specifications. The artifact was developed in Python (versions 3.4 and 3.8) and version controlled by GIT.

Considering the cost of evaluating and testing the solution on a large scale during the production process, another evaluation method was used. The performance of the proposed new hybrid AI solution was measured by comparing the efficiency achieved in the past with the predicted efficiency expected from applying the hybrid AI solution's recommended settings (Cavalcanti et al., 2024), limitations of the artifact were discussed, as calculation time limitations and dependency on past data were the main concerns found, and how to deal with them were addressed for future improvements. Finally, as demanded by DSR, the findings, results and limitations were published in professional academic journals for further and broader discussions with different professionals.

6.2-Data Collection

This research is based on data collected by sensors located in several points of the production process of a thermoelectric power plant. To treat systematically with the dataset containing the sensor measurements of different points 3 steps were used. First business/domain understanding of the production system helped to understand what was measured and what were the important data that should be considered for the research. Secondly, some exploratory analysis of the dataset was done to familiarized with its patterns, identify data quality problems, and explore first insights. Finally, Data preparation/manipulation methods were applied to construct the final dataset as discussed in chapter 8.2.

6.3-Methods

The objective of this study is to propose a hybrid AI solution that is capable of raising the production efficiency in a complex system in our real-world case for a thermoelectric power plant (Cavalcanti et al., 2024). Data were collected from sensors and pre-processed, and an AI solution composed of a DEA-ML-GA mix was applied to it, recommending the production settings for optimizing efficiency and generating a certain desired output (Cavalcanti et al., 2024). The performance of the proposed new hybrid AI solution was measured by comparing the efficiency achieved in the past with the predicted efficiency expected from applying the hybrid AI solution’s recommended settings (Cavalcanti et al., 2024). Auxiliary methods were also used to complement the DEA-ML-GA mix such as Z-score method and R-squared.

6.3.1-DEA

For this research, the DEA-CCR model was selected considering The homogeneity of the DMUs under consideration and the assumption of constant returns to scale (CRS) (Cavalcanti et al., 2024). The DMUs being analyzed operate under similar conditions, ensuring that the inputs and outputs reflect comparable processes and efficiency standards. This similarity allows for a more accurate comparison of efficiency across the DMUs. Additionally, the CCR model's assumption of CRS is particularly fitting, as it posits that increasing the inputs by a certain percentage will yield a proportional increase in outputs. This characteristic is relevant in this context, as the DMUs are of comparable size and can be expected to scale their operations without altering their efficiency. Thus, the DEA-CCR model is a suitable choice for evaluating the relative efficiency of these homogeneous DMUs. Moreover, DEA can be input- or output-oriented; for this research, the input-oriented DEA-CCR model was chosen considering the aim is to minimize the consumption of production inputs, and efficiency was calculated in that sense (Cavalcanti et al., 2024). CCR model can be described by the equation 2 (Mustafa et al., 2021).By comparing the efficiency of a specific unit with the performance of a group of similar DMUs, it is possible to create a rank of most efficient DMUs which can be later investigated, and the efficiency of inefficient DMUs could be increase by copying the behaviour of the most efficient ones(Cavalcanti et al., 2024). pyDEA (, a software package developed in Python for conducting DEA (Raith et al. 2019), was employed to apply the DEA-CCR model.

6.3.2-Machine Learning

This research proposes the use of elastic net (LN method), gradient boosting (RT method) and support vector machine (RT method), which are well-known, successfully implemented ML methods (Cavalcanti et al., 2024). They are widely used to predict

continuous data based on continuous dependent variables (Laref et al., 2019; Mokhtari et al., 2020; Touzani et al., 2018).

Moreover, Robust Scaler, Min-Max Scaler, Standard Scaler, Max-Abs Scaler, Quantile Transformer (Normal), Quantile Transformer (Uniform), and Power Transformer, widely utilized pre-processing methods were selected as potential candidates for the AI pipeline (Singh & Singh, 2022). Grid Search Cross Validation was done by dividing the dataset into multiple folds and training the model on each fold while using the remaining folds for validation based on some scoring metric (Cavalcanti et al., 2024). The proposed train-test split proportion for testing the AI framework using the thermoelectric power plant data in our example model validation is 70/30, as this ratio is commonly used in previous studies. However, other standard proportions can also be applied depending on the specific context (Chuang & Keiser, 2018; Kebonye, 2021). Section 7.6 discusses in more detail how different train-test split ratios can be utilized within the proposed framework.(Chuang & Keiser, 2018; Kebonye, 2021). Mean Squared Error, Root Mean Squared Error, Mean Absolute Error, R-squared, Adjusted R-squared, Mean Absolute Percentage Error, Median Absolute Error, Mean Squared Logarithmic Error, and Explained Variance Score are possible scoring methods that can be used in regression analyses (Cavalcanti et al., 2024). R-squared was chosen for scoring method in this research as it is the most popular and widespread used method for such cases (Chicco et al., 2021).

Therefore, the performance of the models was evaluated based on R-squared, and the best performing combination of ML and pre-processing methods were selected (Cavalcanti et al., 2024). Gradient boosting proved to be the best scoring model for each of the ML models trained (Cavalcanti et al., 2024).

Gradient boosting is a machine learning algorithm that has gained popularity in recent years due to its ability to handle complex datasets and to produce accurate predictions (Cavalcanti et al., 2024). It is an ensemble method that combines multiple weak learners to create a strong learner (Cavalcanti et al., 2024). The algorithm iteratively adds new trees to the model, with each new tree correcting the errors of the previous tree(Cavalcanti et al., 2024). It calculates the negative gradient of the loss function with respect to the predicted values and uses this as the target variable for the next tree (Cavalcanti et al., 2024). The final prediction is a weighted sum of the predictions of all the trees, where each tree is given a weight proportional to its contribution to the overall accuracy of the model (Cavalcanti et al., 2024). Gradient boosting can handle complex datasets and produce accurate predictions (Z. Zhang et al., 2019). Scikit-learn, a free ML software library for the Python programming language was used to create the ML pipelines, including ML models and pre-processing methods (Cavalcanti et al., 2024).

6.3.3-Genetic Algorithm

The basic elements of the GA are chromosome representation, fitness selection, and biological-inspired operators (Katoch et al., 2021). Chromosomes are composed of a set of different values, which are referred to as genes, and represent one possible solution for the optimization problem (Katoch et al., 2021). The fitness function is applied to assign a score for all the chromosomes in the population. The biological-inspired operators are selection, mutation, and crossover (Katoch et al., 2021).

In selection, the chromosomes are selected based on their score calculated by the fitness function; there are different selection methods, such as the rank-based selection technique, roulette wheel, and tournament selections (Hussain et al., 2017; Orong et al., 2018). Since the rank-based selection technique led to better performance, it was applied to the hybrid AI model (Cavalcanti et al., 2024). In this strategy, individuals in the population

are first ranked based on their fitness values, which reflect how well they solve the problem at hand (Hussain et al., 2017; Orong et al., 2018). The ranking process provides a way to differentiate the individuals based on their relative fitness, allowing for the creation of a probability distribution that takes the ranking into account (Hussain et al., 2017; Orong et al., 2018). The probability distribution is then used to select a set of individuals for reproduction, with the fittest individuals having a higher probability of being selected (Hussain et al., 2017; Orong et al., 2018). However, this method also allows for the selection of lower-ranked individuals, providing a chance for diversity in the population and avoiding premature convergence to sub-optimal solutions (Hussain et al., 2017; Orong et al., 2018). This helps to balance the exploration and exploitation of the solution space, leading to a more thorough search for optimal solutions (Hussain et al., 2017; Orong et al., 2018).

The crossover operator, which is a random locus that changes the sub-sequences between two chromosomes to create offspring, is chosen (Lawal et al., 2021). Mutation randomly flips chromosomes based on a probability rate, low mutations could lead to local optimum and very high mutations could lead to random non-optimal results, hence different mutations were applied, and results were checked to assert there was convergence towards a global optimum (Lawal et al., 2021).

6.3.4-Z-score Method

Outlier removal from the dataset was an essential step in data preprocessing. This procedure was undertaken to eliminate data points that have the potential to introduce significant distortions in calculations and, subsequently, lead to incorrect conclusions(Cavalcanti et al., 2024). Outliers are those data points that markedly deviate from the overall data distribution, and they can stem from various sources, including measurement errors, missing data, or inherent natural variability(Cavalcanti et al., 2024).

Measurement errors, a common occurrence in various domains, particularly in industrial IoT systems, pose a significant challenge. In these systems, even a single anomalous event can lead to the generation of inaccurate data from a multitude of sensors (Y. Liu et al., 2020). The resulting outliers, if not addressed, can adversely affect the accuracy and reliability of any subsequent statistical analyses and predictive models(Cavalcanti et al., 2024).

To address this issue, the dataset underwent an outlier removal process utilizing the z-score method. The z-score, a widely-used statistical metric, quantifies how far a particular data point deviates from the mean of the dataset (Cavalcanti et al., 2024). It is calculated by subtracting the mean value from the data point and then dividing the result by the standard deviation (Cavalcanti et al., 2024). The z-score serves as an indicator of how extreme or typical a data point is within the dataset (Cavalcanti et al., 2024).

In the context of outlier detection, a commonly accepted threshold in statistics is a z-score of ±3. In practical terms, this means that any data point that falls outside the range of 3 standard deviations from the mean is considered an outlier (Cavalcanti et al., 2024). This threshold serves as a practical guideline for identifying data points that exhibit significant deviations from the typical behavior of the dataset, ensuring that they are effectively filtered out during the data preprocessing stage (Cavalcanti et al., 2024).

This proactive approach to outlier removal is integral to maintaining the integrity of the dataset and ensuring that subsequent analyses and predictions are grounded in accurate and reliable data. It serves as a safeguard against the potential distortions and erroneous conclusions that outliers can introduce.

6.3.5-R squared

R-squared, symbolized as R^2, is an important statistical measure used in machine learning to evaluate the performance of predictive models, especially in regression analysis. It helps to determine how well a regression model can explain the variance in a dependent variable by considering the influence of independent variables (Shaw et al., 2023; Wan et al., 2021). Every dataset has some variation in the dependent variable, which is the outcome or prediction target in machine learning (Shaw et al., 2023; Wan et al., 2021). This variation can be partially explained by including independent variables in the model, while some of the variation remains unexplained (Shaw et al., 2023; Wan et al., 2021). R-squared quantifies the proportion of the total variance in the dependent variable that the model can account for (Shaw et al., 2023; Wan et al., 2021).

In machine learning, R-squared is important because it indicates how well the model can capture and explain the variability in the dependent variable (Priya Varshini & Anitha Kumari, 2020; Redell, 2019). The value of R-squared ranges from 0 to 1: a value of 0 means the model cannot explain any of the variance, while a value of 1 means the model explains all the variance (Priya Varshini & Anitha Kumari, 2020; Redell, 2019). For example, an R-squared value of 0.7 means the model explains 70% of the variance, leaving 30% unexplained (Priya Varshini & Anitha Kumari, 2020; Redell, 2019). A higher R-squared value usually indicates a better model fit, meaning the independent variables in the model effectively explain a significant portion of the variance in the dependent variable (Priya Varshini & Anitha Kumari, 2020; Redell, 2019). This suggests the model is good at making predictions. However, high R-squared values can be misleading (Priya Varshini & Anitha Kumari, 2020; Redell, 2019). A very high R-squared might indicate overfitting, where the model fits the training data very well but performs poorly on new, unseen data (Priya Varshini & Anitha Kumari, 2020; Redell, 2019). Therefore, it's important to use R-squared

alongside other performance metrics and to understand the specific context and goals of the machine learning problem (Priya Varshini & Anitha Kumari, 2020; Redell, 2019).

In summary, R-squared is a key measure in machine learning for assessing how well a regression model explains the variance in the dependent variable. It is used in predictive modeling and regression analysis to guide model selection and performance evaluation, helping to ensure the chosen model aligns with the objectives of the machine learning task.

7-Framework for Production Efficiency Optimization Using Machine Learning Based Digital Twin Simulation

This chapter was written based on publication Cavalcanti et al. (2024). The framework I present in this exploration aims to leverage Digital Twin technology supported by ML algorithms for real-time monitoring, prediction, and optimization of manufacturing processes. The objectives of the framework are as follows:

- To provide an in-depth understanding of how ML-based DT can be created and used in production systems.
- To highlight the power of ML in creating ML-based DT for predictive purposes.
- To introduce a structured framework that integrates DT technology, DEA, ML and GA for optimization of production efficiency and why those methods were selected for better results.
- To offer practical insights and use cases that illustrate this framework's real-world benefits and implications.

This Chapter provides a comprehensive overview to production efficiency optimization using the proposed framework. Each chapter explores specific facets, from the

fundamentals of Digital Twins and Machine Learning to their integration, implementation, and real-world applications.

In an era where the global manufacturing industry stands at a crossroads, balancing profitability with sustainability and innovation, the "Framework for Production Efficiency Optimization Using Machine Learning-Based Digital Twin Simulation" illuminates a path forward. It guides organizations towards a future where production processes are efficient, intelligent, adaptive, and intrinsically intertwined with the digital realm. Moreover, this research addresses gaps in the literature by introducing an innovative Artificial Intelligence Framework for optimizing production efficiency. The framework, incorporating ML-based DT Simulation, GA, and DEA, is designed for application in various production systems. I provide practical guidelines and examples for implementing this comprehensive framework in real-world practices. Additionally, I include a formal model description, enhancing the clarity and precision of the presented model.

7.1 Modeling a Complex Production System

The proposed framework articulates the modeling of a complex system, incorporating inputs, outputs, process control parameters, and state variables to formulate an AI representation of the production system. In a practical context, this is achieved by accessing a dataset of sensor signals, processing the information, and generating an AI representation of the production system. In essence, the framework creates a ML-Based DT to represent the intricacies of the production system. Input parameters serve as the primary sources that sustain the production system, while process control parameters function as settings to fine-tune the production processes. State variables encompass measurements derived from the production system, while output parameters encapsulate the principal outcomes of the production process. Leveraging previous records of these variables, the

ML-Based DT can simulate the behavior of a production system based on historical data. In summary, the ML model learns from past data and predicts how the production system would behave under specific conditions.

To illustrate this concept, consider a hypothetical scenario where the production system is akin to a distillation process for bioethanol. Input parameters, in this context, would include the quantities or flow rates of fermented corn syrup and steam as primary energy, where the former should be measured both as ethanol equivalent and mass flow. Process control parameters might encompass settings like steam pressure settings and reflux flow rates. State variables could be represented by measurements of temperatures at various points of the process or the chemical composition of the final products. Finally, output parameters would measure the quantities or flow rates of the finished products, again both as ethanol equivalent and mass flow.

Accessing data from past distillation runs, e.g., cross sectional data of hourly production empowers the ML-Based DT to undergo training, enabling the simulation of the production system's behavior. This training process involves learning from historical data, providing the capability to predict how the production system would respond under various conditions. This facet underscores the core essence of the framework, wherein the ML-based DT emerges as a potent tool for representing and comprehending complex production systems. A crucial distinction between process control parameters and input parameters lies in their impact on production costs. While process control parameters do not directly account for production costs, input parameters do. This differentiation proves vital for potential optimization endeavors where the goal is to minimize inputs. When the input and output parameters are measured in different units (like in our examples quantities of steam and ethanol are not comparable), they could be weighted by their monetary value, enabling

optimization financially the production system. This consideration becomes pertinent in later stages if optimization strategies are envisaged.

The machine learning-based digital twin, through its ability to discern and learn from past data, forms a pivotal element in this comprehensive framework designed for a nuanced understanding of intricate production systems. State variables assume the role of secondary outputs within the production system, potentially constraining production. In the context of the distillation example, a relevant state variable could be certain quality parameters of the finished product, like the amount of impurities. These quality parameters can be subject to limitations based on specification limits and should act as constraints for the production system.

The digital twin then describes and models the complex system, incorporating inputs, process controls, outputs, and state variables through the power of artificial intelligence. This ML-based DT is subsequently capable of simulating the production system. Drawing from previous data, it can determine the outputs and state variables (secondary outputs) of the production system when a certain mix of inputs and process controls are randomly set. This process essentially creates a digital replica of the real production system, deploying machine learning to discern and predict its behavior under specific conditions.

In essence, the machine learning-based digital twin becomes an invaluable tool for comprehending and representing the intricate dynamics of complex production systems. It's worth noting that both previously mentioned production systems could theoretically be described using mathematical equations. These differential equations would describe the equilibrium of the various volatile components at different stages of the distillation column, the heat flow that is supplied through steam and that would be required or recovered through

the evaporation and condensation through the distillation as well as the heat losses of the system. For relatively simple production systems, it may be feasible to formulate mathematical equations capable of describing their behavior under various production setups. However, when dealing with more intricate and complex systems, using an ML-based DT emerges as the practical and viable approach to creating a digital replica of the production system.

The ML-based DT, as demonstrated in the preceding example, can be generalized, as depicted in the illustrative Figure 6. This generalization underscores its adaptability and applicability to a broad spectrum of production scenarios. Unlike traditional mathematical equations, which may struggle to encapsulate the complexity of dynamic and multifaceted production environments, ML-Based DTs excel in capturing and simulating the nuanced interactions within these systems. Thus, when confronted with the intricacies of modern and sophisticated production setups, the ML-based DT could stand out as the preferred methodology for creating an accurate and versatile digital representation of the production system.

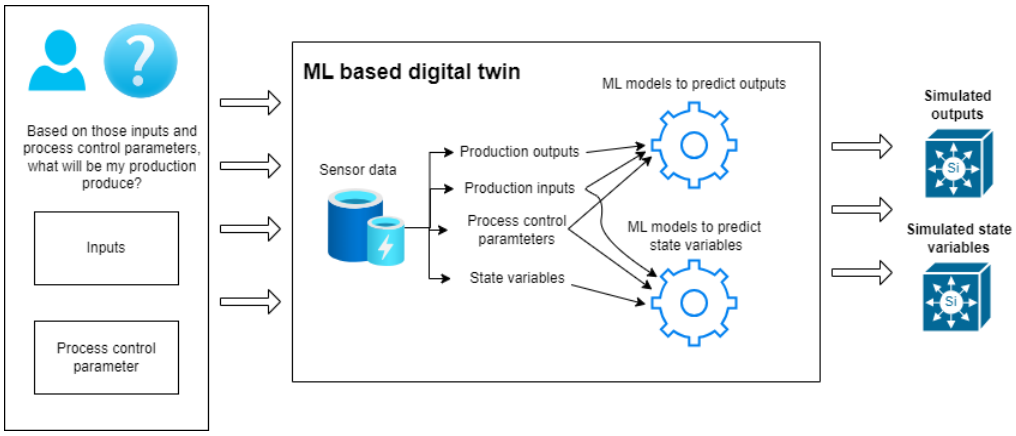

Figure 6 – ML-based DT for production optimization.

As illustrated in Figure 6, the complex process of modeling intricate production systems can be accomplished through the creation of a ML-based DT, adeptly trained on sensor data encompassing production outputs, inputs, process control parameters, and state variables. This multifaceted approach involves training two distinct ML models: one for predicting production outputs and another for predicting state variables. The ML model designed to forecast production outputs is trained using historical data of production outputs, inputs, and process control parameters. Concurrently, the ML model responsible for predicting state variables is trained using past data involving production inputs, process control parameters, and state variables. Once the ML-based DT is constructed, it gains the capability to simulate the production outputs and state variables for a randomly selected set of inputs and process control parameters.

Furthermore, with knowledge of the simulated output and input, it becomes feasible to calculate a theoretical efficiency. The intricate process of calculating production efficiency is further explored in the subsequent chapter, employing the implementation of a combination of DEA, and ML, known as DEA-ML. This combined methodology enhances the depth of analysis and provides a comprehensive understanding of the production efficiency within the system.

7.2 Optimization Techniques on ML-based Digital Twin

Upon the establishment of a ML-based DT, the capability arises to conduct multiple simulations of the production system under diverse conditions. This involves the random simulation of input parameters and process control parameters, enabling the ML model to predict corresponding output and state variables for these simulated values. The subsequent step involves exploring a spectrum of potential outputs derived from specific inputs and process control parameters, thereby facilitating the identification of an optimal production

setup. The goal of this proposed method is to minimize resource utilization while adhering to constraints, inherent to the production system, set by state variables, and simultaneously maximize output. This methodology integrates the strengths of the ML-based DT and optimization techniques to search for the most efficient solution.

It is crucial to note that certain production systems may lack clarity regarding what should be maximized or minimized, especially when multiple inputs and outputs exist. Standard minimization problems may neglect the consideration of process control parameters, variables set by the production planner that might not impact or are irrelevant to production costs but significantly influence output, enhancing quality and quantity. Consequently, efforts to minimize the usage of such parameters might prove futile when seeking optimal production settings. To address these challenges, DEA is recommended. Stochastic Frontier Analysis (SFA), may serve as a potential substitute for DEA but requires the specification of a functional form for the production process, such as Cobb-Douglas or Translog. Unlike DEA, which uses linear programming to construct an empirical efficient frontier without assuming a specific functional structure, SFA is model-dependent. This requirement reduces its adaptability in comparison to DEA, which is inherently more flexible and data-driven. These distinctions, along with further explanation, have been addressed and added to the thesis. DEA offers an efficiency measurement for a complex production system, that can take into consideration multiple inputs and outputs, even if they are measured in different units. Using the value of these inputs and outputs enables to calculate the relative efficiency for each historical production setup, that can be appended to the dataset. This relative efficiency becomes pivotal in predicting and maximizing the efficiency of simulated inputs, process control parameters, and outputs generated by the ML-based DT.

With the ability to simulate various setups, GA can be employed to search for optimal production settings. In this thesis GA is used for optimization but other optimization algorithms could be used and replace GA, for example Fuzzy Logic, Bayesian Optimization or Reinforcement Learning, can be used as alternative optimization techniques. Therefore, the creation of an AI solution for production system efficiency optimization is achieved by amalgamating three well-known mathematical methods: DEA for calculating production efficiencies, ML-DEA for creating an ML-based DT, and GA for searching for optimal production settings. This comprehensive approach capitalizes on the strengths of each method to address the complexity of optimizing production systems.

7.3 AI Model Overview

The proposed AI solution operates by analyzing a dataset of sensor signals that encompasses production inputs, process control parameters, state variables, and outputs. To optimize the production system, the user is required to define constraints for inputs, process control and state variables, setting limits that restrict the production process (e.g., ensuring that the output product is within the specification limits or the steam pressure is below the safety threshold). Additionally, the user needs to specify the desired output, indicating the production target under optimized settings. The AI solution processes this information to generate recommendations for optimal settings and configurations, with the overarching goal of maximizing production efficiency.

Comprising three main components, DEA, ML-Based DT, and GA, this AI solution forms a robust Hybrid AI system. These components collaboratively function to predict the relative production efficiencies associated with diverse settings and configurations. The GA component leverages these efficiency predictions to explore various settings and configurations based on user inputs, ultimately selecting the optimal one. A high-level

overview of the system is illustrated in Figure 7. This integrated approach allows the AI solution to offer actionable insights into optimal production settings, ensuring that the production process aligns with user-defined constraints and targets. The synergy of DEA, ML-based DT, and GA components contributes to the system's effectiveness in providing recommendations that enhance overall production efficiency.

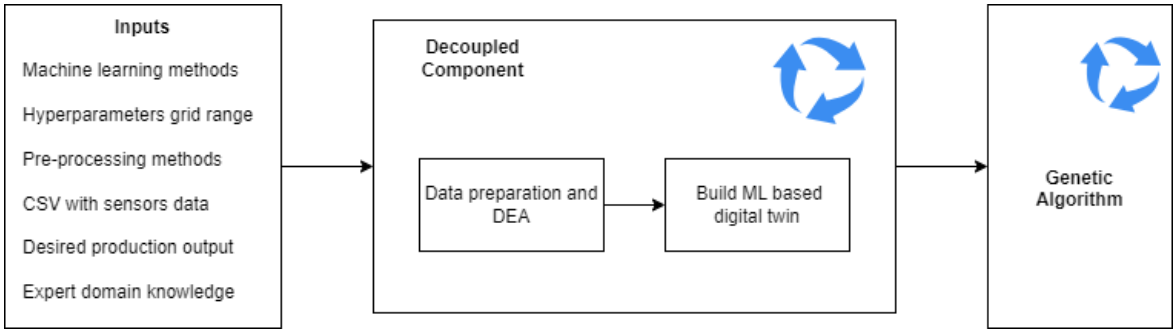

Figure 7– AI framework overview

The first step to build the model is to determine the Inputs of the system, when the user should define initial settings prior to the model's calculations. There are 6 inputs required by the hybrid AI solution for its configuration:

- Expert domain knowledge
  - to choose the right sensor data set,
  - define input-output weights and
  - variable constraints
- Sensor data of inputs, outputs, process control and state variables
- Machine learning methods
- Hyperparameters grid range
- Pre-processing methods

- Desired production output

Once the initial model inputs are prepared, the AI engine initiates its operations. The process begins with data preparation and DEA calculations using historical data. Expert knowledge is required to choose the appropriate sensor data and to define the input-output weights in case they are measured in different units. This phase involves removing outliers and computing the relative production efficiencies. Subsequently, the AI engine proceeds to the second step, where ML models are trained to predict/simulate relative production efficiency, production output, and state variables based on a given set of inputs and process control parameters. This step results in an extended version of the ML-based DT. Importantly, DEA and ML calculations can be decoupled and operate independently from GA, significantly reducing computational load. DEA and ML need to be recalculated periodically, with GA reusing the results from the previous step to iteratively search for optimal solutions. The frequency of DEA-ML calculations is user-determined, guided by expert domain knowledge, more frequent updates on DEA-ML fresher and more descriptive is your virtual model of the DT.

In the final step, GA simulates various production inputs and process control parameters, selecting optimal configurations to achieve the desired production output. GA employs ML models as fitness functions, using them to calculate production efficiencies. The ML predictions of production outputs and state variables act as constraints on GA solutions, aligning with the user-defined targets and allowable ranges for state variables. As a culmination of these steps, the optimal production inputs and process control parameters are identified, ensuring the realization of the desired production output in the most efficient manner. This integrated approach underscores the synergy between DEA, ML, and GA components, collectively contributing to the efficiency and effectiveness of the AI solution in optimizing the production system.

7.4 Model Inputs and Sensor Data

The core of the AI solution is a dataset filled with sensor signals, which act like the fuel that helps it learn. This dataset gathers important insights and needs to cover a wide range of conditions: from low to high loadings, as well as process control and state variables showing both efficient and inefficient performances. Preparing this dataset might require running the process over a long period and selecting times when it operated under these varied conditions. Sometimes, we might even need to run the process in ways we wouldn't normally, just to gather the data needed to build the AI model.

The proposed AI solution relies on a rich dataset of sensor signals, including continuous data streams like infeed speed, pressure, temperature, and binary signals that show the status of key features or inputs. While these signals form the backbone of the dataset, we can also add data from external sensors or public sources, like weather forecasts. This broader perspective helps the AI adapt to a wider array of factors.

One of the main challenges is selecting the most representative features from all this data. We need to avoid irrelevant, noisy, or redundant data, which can cloud the learning process. This careful feature selection is crucial for creating a dataset that truly captures the essence of the problem without unnecessary extras.

The sensor dataset is not just a collection of past inputs, outputs, and process control parameters; it's a complex mix of intertwined information that often requires the expert eye of someone familiar with the domain. User expertise is indispensable here. Practitioners need to use their deep understanding of the problem to assess and categorize the data appropriately. In essence, the sensor dataset is where the user's knowledge and the AI's analytical power come together to create a clear and accurate understanding of the process.

7.4.1 Expert Domain Knowledge

Expert domain knowledge about the production system is needed for the definition of what are inputs, outputs, process control parameters, and state variables from the sensor dataset. That way the AI solution can differentiate variables and address them correctly in its engine. Expert domain knowledge plays a pivotal role in shaping the success of any AI-driven production system. Domain experts must identify and define the critical elements within the sensor dataset. These elements encompass not only the basic inputs and outputs but also the nuanced process control parameters and state variables that drive the entire production system. Expert domain knowledge is required to understand the mechanics and the key drivers of the production process, selecting those process control variables that influence the output and the overall efficiency of the process, as well as those state variables that act as a constraint and/or indicate the performance of the process. This knowledge can be supplemented with techniques that are used in developing machine learning models: measuring relationship between variable and identifying variable importance. In the vast amount of data available from sensors, a big challenge is carefully choosing the most important features. The person handling this needs to be cautious, watching out for data that doesn't contribute towards the goal of optimization, is noisy, or redundant. Expert domain knowledge also required to define the weights of the input and output parameters, especially when they are measured in different units. These weights influence the optimization strategy therefore need to be aligned with the business strategy. Finally, the experts need to declare what are the minimum and maximum acceptable values each of the input, process control and state variables. These values will constrain the optimal solutions space given only feasible solution for the optimization problem.

The significance of this expert domain knowledge cannot be overstated, as it serves as the bedrock upon which the AI solution is built. With a comprehensive understanding of

the system's intricacies, AI can distinguish between the various data points and attributes, ensuring that it assigns the correct significance and context to each of them within its underlying engine. Wrongly determining which data is relevant can lead to increased calculation times and wrongly classifying it into inputs, outputs, process control parameters and state variable can compromise the AI engine, hence also compromising optimal solution recommendations given by it.

7.4.2 Sensor Data

The foundation of our AI solution is a dataset filled with sensor signals. This dataset is where we collect important insights, and it needs to cover a wide range of process states: inputs and outputs from low to high levels, as well as control and state variables that include both efficient and less efficient performances. Preparing this dataset may involve covering an extended time period and selecting those times when the process operated under these varied conditions. Additionally, preparing the dataset may require running the process under conditions that are not typically used, to gather data for building a ML-based DT capable of simulating different scenarios.

7.4.3 Desired Production Output

Finally, it is essential to clearly define the desired production output, as this represents the production goals. These desired outputs can include values from production plans, contractual obligations, or even production forecasts. This information can be used to validate the ML-Based DT and understand the efficiency improvements it suggests. The desired output is important because it will serve as a constraint in the GA. The GA will generate random production inputs and process control parameters, then the ML model will predict the expected output and efficiency for that setup. Even if one of the simulations generates an efficient setup, it is not useful if the output is outside the desired range, so this

setup will be discarded, and the GA will search again for a better setup within the desired output range. Therefore, the desired output will constrain the optimal production setup suggested by the GA. Another option that can be implemented, depending on the user's preferences and the specific details of the production system, is to create a function that penalizes the efficiency of setups that exceed or fall short of the desired output instead of completely discarding those GA solutions.

7.5 Data Preparation and DEA

Data preparation and DEA calculation are the first components of the AI solution. During this step, two predefined inputs are needed from the user; the recorded values of the sensors data and expert knowledge to determine which from those sensors are measurements of production inputs and outputs. At this phase, CSV data is used to perform DEA and calculate the productive efficiency of the DMUs, the different valid configurations and combinations of input–output values), which will then pass through an outlier removal performed by the interquartile range (IQR) or Z-score methods.

The choice between Z-scores and IQR depends on specific research questions and dataset characteristics. Outlier removal is a vital pre-processing step for machine learning models, enhancing accuracy by eliminating data points that could introduce error variance and reduce statistical test power (Nnamoko & Korkontzelos, 2020; Perez & Tah, 2020). In contrast, keeping outliers in the dataset increases error variance and reduces the power of statistical tests; therefore, outlier removal is recommended for most ML models (Brownlee, 2020). The pre-processed sensor dataset, now accompanied by relative productive efficiency values, serves as the outcome of this step. This dataset is pivotal for ML model training in subsequent phases.The scalability of the DEA method in handling a large portion of datasets extracted from the manufacturing sensors is limited by the number of variables

the LP solver is allowed to handle e.g., the Python package (PyDEA), using the PuLP package and IBM ILOG CPLEX as a solver is limited to 10000 variables as an academic license. Data must be stratified and down sampled to meet these limitations, if PyDEA is used as it was used in this research. If other package is used for calculating DEA its limitations should be respected. Moreover, Large datasets can lead to heavy, time-consuming DEA calculations and a balance between dataset size and calculation time must be taken into consideration depending on user's needs. More data means more descriptive modelling of the production system consequently higher efficiencies can be found by the DEA, however time-consuming calculations can not fit some users preferences.

The data preparation and DEA can be decoupled from other phases, and it does not need to be done every time the user wants an optimal production setup recommendation from the AI solution. User should determine how frequently DEA should be calculated, considering that newer sensor data can bring new knowledge to the model.

7.6 ML Based Digital Twin Model Training

After the DEA and data preparation phase, the subsequent step in building the proposed AI solution involves ML model training. During this phase, the previously treated dataset is used as input to identify the optimal ML model capable of predicting production relative efficiency and output based on a set of inputs and process control parameters. However, it is crucial to perform a train-test split before training begins. Dataset splitting is a standard practice in ML algorithms, where a portion of the data is exclusively reserved for validation, enhancing the robustness and impartiality of the models (Bai et al., 2021; Vabalas et al., 2019). Common proportions for train-test splits include 60/40, 70/30, 75/25, and 80/20.

The choice of the train-test split ratio can vary based on user preferences and dataset characteristics. In practice, the selection of the ratio depends on the dataset size and specific goals. A common practice involves using an 80-20 or 70-30 split (training-test) for reasonably large datasets. In cases where data is limited, a larger proportion may be used for training (Kebonye, 2021; Rácz et al., 2021).

As mentioned earlier, users have the flexibility to choose from multiple ML methods, potential hyperparameter values, and pre-processing methods. All these parameters must undergo validation, and their scores should be assessed. The best model is then selected through an exhaustive search method known as grid search. Grid search is a hyperparameter optimization technique that thoroughly explores a given subset of the hyperparameter space. This subset comprises well-defined possible ranges of values for each ML model, and the selection process is empirical. Therefore, conducting a grid search through these potential values is a common practice.

It is essential for users to carefully choose the number of possible values within the common optimal range, as increasing this number can significantly extend the training time. While GA has been discussed as an alternative to grid search, with many studies achieving positive results using this approach (Liashchynskyi & Liashchynskyi, 2019), the presented AI solution opts for a grid search. GA, in this study, is reserved for a different purpose, not for hyperparameter tuning. Specifically, GA is employed solely to find the optimal production setup, with computational resources focused on this task. Further details about the GA phase will be provided in the subsequent section.

In the context of this AI framework, it's important to note that the grid search extends not only to the model's hyperparameters but also includes the chosen ML methods and preprocessing methods discussed in sections **6.3.2** respectively. Therefore, potential values

for optimizing the model go beyond hyperparameters alone. This extension of the grid search is a common technique and is supported by the Python library scikit-learn (Pedregosa et al., 2011).

Grid search, in this scenario, explores various combinations of hyperparameters, preprocessing techniques, and ML methods. It assesses the performance of these combinations using a selected scoring method. The choice of the scoring method is essential for evaluating regression analyses, and this research adheres to the suggestion from Chicco et al. (2021) that recommends R-squared as the standard statistical measure for regression evaluations across scientific areas. As a result, R-squared is the chosen scoring method for the grid search to identify the best-performing model. Figure 8 schematizes the ML model training process and illustrates how grid search operates. All preprocessing methods are combined with all ML methods, and various values for each ML model's hyperparameters are tested to determine the best-performing ML model.

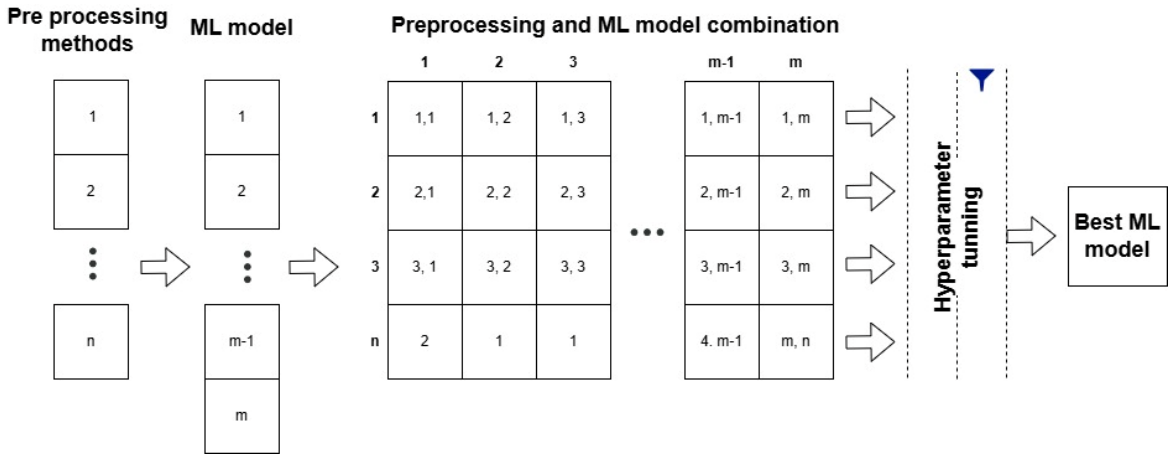

Figure 8 - ML model training process

Following the grid search phase, the output involves two optimally tuned ML models: one designed to predict production output based on process control parameters and production inputs, and another ML model intended to predict DEA relative efficiency. The latter model is built on process control parameters, inputs, and the predicted output from the first ML model. Furthermore, "x" ML models are created to predict state variables based on inputs and process control parameters. The predicted state variables from these models serve as constraints in the subsequent phase, which involves GA. As depicted in Figure 6, this process is repeated twice, plus the number of state variables times.

Similar to the data preparation and DEA calculation phases, the ML-based digital twin model training can be decoupled from other stages. It doesn't need to be executed each time the user seeks optimal production setup recommendations from the AI solution. The frequency of ML model training is determined by the user, taking into account that newer sensor data may enhance the model's knowledge and lead to more reliable simulations.

The trained ML models and DEA results can be stored in memory for easy accessibility in the next step (GA). If the user decides that the existing data is outdated and that new data should be included, new ML and DEA calculations can be performed. The updated results can then be stored in memory, ensuring they are readily available for the GA to access at any time.

7.7 Genetic Algorithm Optimization

The ultimate component of the proposed AI system is the GA. In this phase, the GA generates diverse production inputs and process control parameters randomly and then identifies the optimal configuration using the pre-trained ML model. This ML model was previously trained to simulate both production output and DEA relative efficiency. The

relative efficiency is employed in the GA fitness function, which is responsible for assessing the quality of each potential solution by determining the efficiency based on the ML model's prediction.

Process control parameters and production input parameters are randomly generated to form an initial population for the algorithm. The user must define the range of possible values for these parameters to create the population's gene space. Choosing an appropriate population size requires careful consideration. A limited population size restricts the search space, potentially leading to a local optimum. Conversely, an excessively large population size expands the search scope and increases computational burden (Vie et al., 2020). Therefore, the population size should be selected judiciously to strike a balance. Using the GA population values, the production output can be simulated by utilizing the previously trained ML model for predicting production output.

Subsequently, the second ML model, calculated in the previous step, functions as the fitness function of the GA. The initial randomly generated production setup has its predicted efficiency calculated by the second ML model. Each prediction from the second ML model should consider the desired production output as a constraint or penalize the efficiency based on user criteria before incorporating it into the GA's fitness function. This calculation results in a subset of different production setups and their respective efficiencies, categorized between the worst and best setups, comprising values of different combinations of inputs and process control parameters.

Furthermore, the user must choose a crossover method, such as Single Point Crossover, Two-Point Crossover, Uniform Crossover, or variations in the programming of a chromosome or chromosomes from one generation to the next, depending on preferences and dataset characteristics, aiming for better GA performance. Similarly, different selection

methods should be chosen, including selecting only the strongest and then crossing over themselves, best-worst selection criteria, and rank-based selection.

In the proposed AI system, the GA is subject to constraint boundaries, ensuring that it generates only feasible solutions. The prediction of state variables relies on randomly generated production inputs, process control parameters, and the ML models trained in the previous step. Feasibility is determined by assessing whether all predicted state variables fall within the user-defined constraining range. If all predicted state variables are within this range, the solution is considered feasible. Additionally, the predicted output simulated by the AI model can act as a constraint or may penalize the predicted efficiency based on a penalizing function defined by the user. The choice of penalizing function may vary across different production systems, considering that excess or lack of products can impact business efficiency to varying degrees.

For each GA generation, the algorithm collects a set of feasible solutions along with their efficiencies for crossover, creating offspring that inherit a mix of production setups (process control and input parameters) from their parents. Similar to natural evolution, the generated offspring are randomly affected by mutations, ensuring genetic diversity from one population to the next. In this research, mutation involves some of the offspring genes inherited from their parents undergoing random mutations at a rate defined by the user. If a mutation occurs, another ratio of the chromosome's genes is randomly mutated. The mutation rate in GA should be set to a low value, as high mutation rates lead to a primitive random search with no convergence, while excessively low mutation rates can result in local maxima. Previous research suggests mutation rates between 10% to 40% as suitable candidates for GA tuning (Lawal et al., 2021). If the issue of local maxima persists, the user can consider increasing the mutation rate, while if no convergence is found, mutation rates can be decreased (Hassanat et al., 2019a).

After mutation, the GA iterates to the next generation, where the previously trained ML model is once again utilized. The new population undergoes the same process as the initial population, and production efficiency is predicted by the ML model for each chromosome of the newly generated population. Feasible solutions are then collected for further evaluation.

In the final phase of the proposed AI system, GA checks for any termination criteria, and if met, the optimal solution is provided; otherwise, the algorithm returns to the selection phase, where chromosomes are chosen, matched with crossover, and potentially mutated. This iterative process continues until at least one of the termination criteria defined by the user is satisfied. There are two types of termination criteria: calculation timeout and a limitation on the number of GA generations.

Calculation timeout may be necessary due to the fast-paced and dynamic nature of production environments. Waiting too long for a decision can render the decision useless and inapplicable. Alternatively, limiting the number of generations and monitoring if the efficiency converges to some value can be another termination criterion. Balancing the number of generations is crucial to avoid unnecessarily long calculation times, striking a balance between computational resources and efficiency convergency.

GA operates independently of the previous two steps, allowing the user to run GA multiple times for different inputs while using the same DEA and ML model results. Decoupling GA from other phases can significantly reduce calculation time but might impact prediction performance. Therefore, expert knowledge is required to determine how frequently DEA and ML models should be calculated. Considering that sensor data can change significantly over time, prediction errors may occur if production behavior has

changed significantly. In such cases, the user should consider recalculating DEA and ML with the new sensor data.

If computational resources are not a constraint, decoupling can be reconsidered to keep the AI solution always up-to-date and operating at optimal efficiency. Figure 9 provides a summary of the GA workflow within the AI solution.

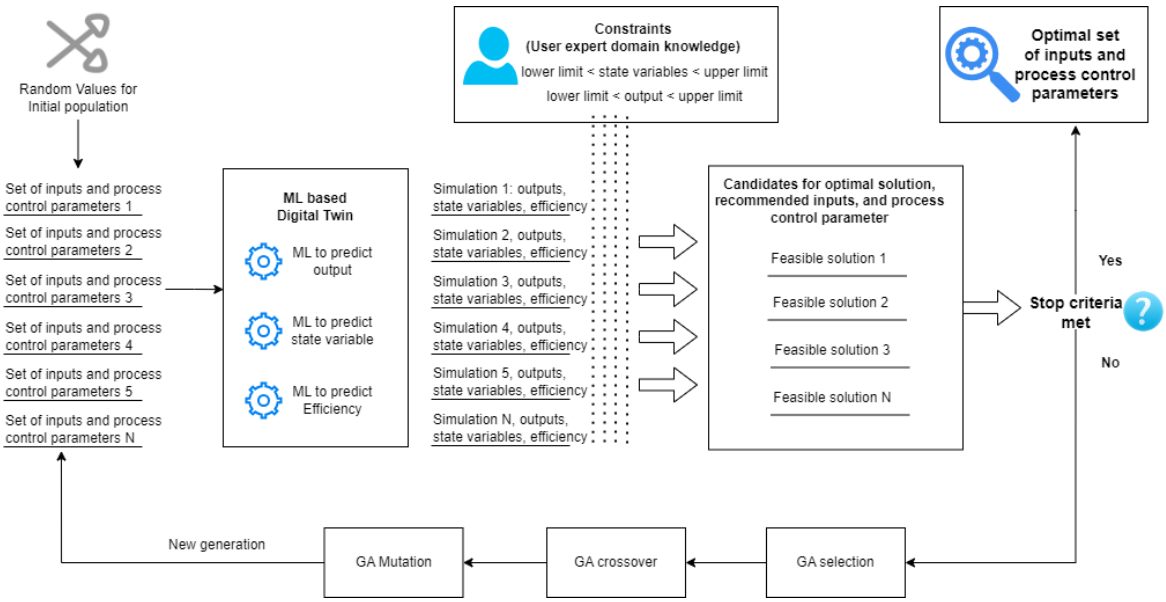

Figure 9 - GA workflow within the AI solution

As illustrated in Figure 9, the GA workflow begins with the generation of random initial values for inputs and process control parameters, forming the initial population. These values serve as input to the ML-based DT of the production system, producing N simulated values based on the GA population size determined by the user. The simulated values encompass production outputs, state variables, and production efficiency.

Following simulation, the initial population undergoes filtering based on constraints defined by the user's expert domain knowledge. Unfeasible inputs and process control parameters (GA chromosomes) are removed from the solution space. Feasible solutions are then evaluated, and if deemed satisfactory or if any stop criteria are met, the best solution is selected as the optimal one. Otherwise, the GA selection, crossover, and mutation phases are initiated, generating a new generation of the population. This cyclical process continues until a stop criteria is satisfied.

7.8 Formal Model Description

$$Input\ parameter\ set: I = \{i_1, i_2, \dots, i_k\}$$

$$weights\ of\ the\ input\ parameters: w^i = \{w_1^i, w_2^i, \dots, w_k^i\}$$

$$Output\ parameter\ set: O = \{o_1, o_2, \dots, o_l\}$$

$$weights\ of\ the\ output\ parameters: w^o = \{w_1^o, w_1^o, \dots, w_l^o\}$$

$$Process\ control\ parameter\ set: P = \{p_1, p_2, \dots, p_m\}$$

$$State\ variable\ set: S = \{s_1, s_2, \dots, s_n\}$$

$$Input\ parameter\ bounds: I^b = \begin{cases} i_1 \le i_1^+, i_2 \le i_2^+, \dots, i_n \le i_k^+ \\ i_1 \ge i_1^-, i_2 \ge i_2^-, \dots, i_n \ge i_k^+ \end{cases}$$

$$Process\ control\ parameter\ bounds: P^b = \begin{cases} p_1 \le p_1^+, p_2 \le p_2^+, \dots, p_n \le p_m^+ \\ p_1 \ge p_1^-, p_2 \ge p_2^-, \dots, p_n \ge p_m^+ \end{cases}$$

$$State\ variable\ bounds: S^b = \begin{cases} s_1 \leq s_1^+, s_2 \leq s_2^+, \dots, s_n \leq s_n^+ \\ s_1 \geq s_1^-, s_2 \geq s_2^-, \dots, s_n \geq s_n^+ \end{cases}$$

$$Desired\ output\ parameter\ set: O^d = \{o_1^d, o_2^d, \dots, o_l^d\}$$

$$Recommended\ input\ parameter\ set: I^r = \{i_1^r, i_2^r, \dots, i_k^r\}$$

$$Recomended\ process\ control\ parameter\ set: P^r = \{p_1^r, p_2^r, \dots, p_m^r\}$$

$$Estimated\ DEA\ efficiency: \theta^e$$

$$Estimated\ process\ control\ parameter\ set: S^e = \{s_1^e, s_2^e, \dots, s_n^e\}$$

Contemporary control theory heavily relies on the state space representation, wherein a control system is delineated by a collection of inputs, outputs, and state variables interconnected through a set of differential equations (Y.-Y. Liu & Barabási, 2016). The state space model can be outlined through the state and output equations, as presented in Equation 3. In this context, the state vector x(t)∈ R^N denotes the internal state of the system at time t, the input vector u(t)∈ R^R captures the known input signals, and the output vector y(t)∈ R^R encapsulates the array of experimentally observed variables. The functions f(·) and h(·) elucidate the dynamics of the intricate system, with Θ encompassing the system's parameters. Our approach aims to approximate the function h(·), seeking to estimate the output y(t) through the utilization of data envelopment analysis.

$$\begin{cases} \dot{O}(t) = f(t, I(t), P(t);\ S(t), \theta(t)) \\ O(t) = h(t, I(t), P(t); S(t), \theta(t)) \end{cases} \quad (3)$$

The ML-based DT approximates $h()$ function. $f()$ function describes the dynamic behaviour of the system is outside of the scope of this research.

Moreover, system efficiency can be estimated using historical data and DEA.

$$max\ \theta(t) = \frac{\sum O(t) *\ w^o}{\sum I(t) *\ w^i}$$

Where $\theta(t)$ is the DEA efficiency of the system at (t) measurement point

s.t.

$$\frac{\sum O(t) *\ w^o}{\sum I(t) *\ w^i} \leq 1 \qquad t = 1, \dots, T$$

Workflow:

1. Data cleansing: remove those measurement points where sensor data were wrongly collected and remove outliers.
2. Estimate $\theta(t)$ using DEA at constant return to scale (DEA_{crs}) for every (t) measurement point.
3. Train ML $h_1()$ model, that approximates $O(t) = h_1(I(t), P(t);\ \theta(t))$
4. Train ML $h_2()$ model, that approximates $S(t) = h_2(I(t), P(t);\ \theta(t))$
5. Use GA to search for best input-process control parameters for the desired output, using $h_1()$ ML model.

$$\max \theta \qquad O^d = h_1(I^r, P^r;\ \theta^e)$$

s.t.

$$I^r \subseteq I^b, P^r \subseteq P^b \text{ and } S^e \subseteq S^b$$

6. Use $h_2()$ ML model to estimate state variable set S based on the output of GA search.

$$S^e = h_2(O^d, I^r, P^r;\ \theta^e)$$

8-Implementation of the Hybrid AI Solution in Energy Sector

This chapter was written based on publication Cavalcanti et al. (2024).

8.1 Production System Variables

Two hybrid AI models were built and tested to evaluate the benefits of process optimisation. Four different types of sensor data are distinguished in the dataset containing information about the production system: ***input parameters*** are the main sources that feed the production system, ***process control parameters*** are settings used to adjust production, ***state variables*** are measurements resulting from the production system, and ***output parameters*** are the main result of the production shown in Figure 10. The input efficiency of the production system can be defined as the ratio of inputs to outputs.

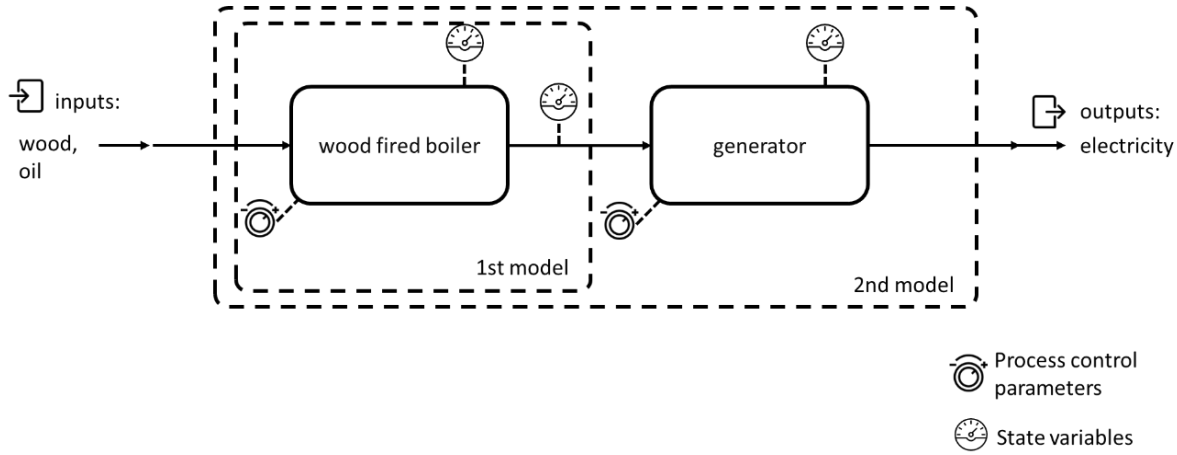

Figure 10 - Conceptual model of the production system

In the real-world example, the thermoelectric power plant, there are 3 inputs used for combustion, for both models: wood infeed A (t/h), wood infeed B (t/h), and oil infeed (kg/h). Wood A and Wood B receive the same raw material, but each feeds the furnace from different points.

The process control parameters that are used to adjust the performance of the combustion process in our dataset are the fluid bed primary air pressure (bar), burner primary air pressure (bar), burner secondary pressure (bar), and burner tertiary air pressure (bar).

State variables that describe the secondary outputs of the system are listed as follows: steam pressure (bar), steam temperature (°C), furnace temperature A (°C), furnace temperature B (°C), pre-ECO flue gas temperature A (°C), pre-ECO flue gas temperature B (°C), flue gas temperature A (°C), flue gas temperature B (°C), flue gas NOx conc. (ppm), flue gas CO conc. (ppm), and flue gas flow rate (m3/h). Temperatures measured by different sensors located in different positions are alphabetically enumerated (A, B, etc.). These

temperatures are consequences of the production process and are important indicators for the process, as they can act as constraints for production. Each of these indicators may have a range of desired values and running production with process indicators outside of their allowed range can cause accidents and/or environmental damage and should be avoided.

The output parameter in the first model is the generated steam production (MJ/h), while the output for the second model is the generated electricity (MW/hr). The output of the first model (steam production with the corresponding steam pressure and steam temperature) could be taken as the additional process control parameter for the second model. Figure 11 illustrates the process plant of our real-world case, indicating the various sensor measurements.

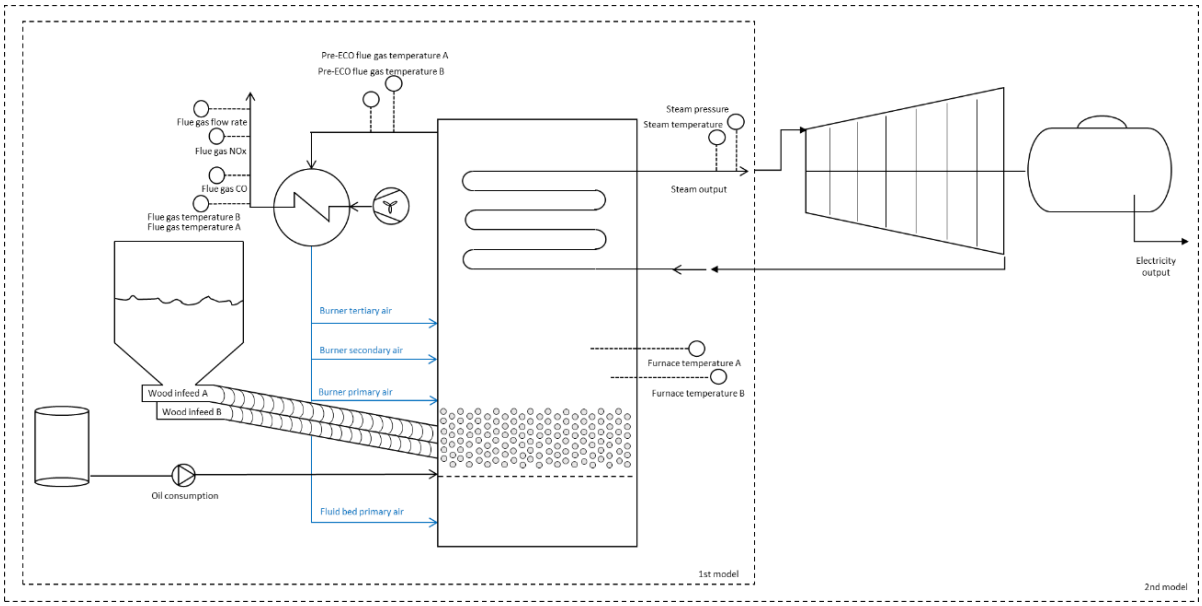

Figure 11 - Schematic diagram of the thermoelectric plant

8.2 Database Preparation and Exploratory Data Analysis

The input dataset contains values of sensor measurements (input parameters, output parameters, process control parameters, and state variables) of a thermoelectric power plant as a regular time series dataset for every day and hour during 2020. As the data were not validated, a data cleaning process was needed as the first step of the research. Data duplications identified in the raw dataset were removed, data duplications could happen during the data collection due to errors in the information system and data warehousing. Moreover, measurement errors such as negative values or values close to zero did not reflect reality, these errors occurred when sensors were malfunctioning; hence, these values were also removed from the dataset, another possibility for missing values would be that the thermoelectric power plant was not working on, to maintenance or other unknown reasons related to production issues. Figure 12 shows timeline of steam output. Where we can seem some timeframes where there is no production at all.

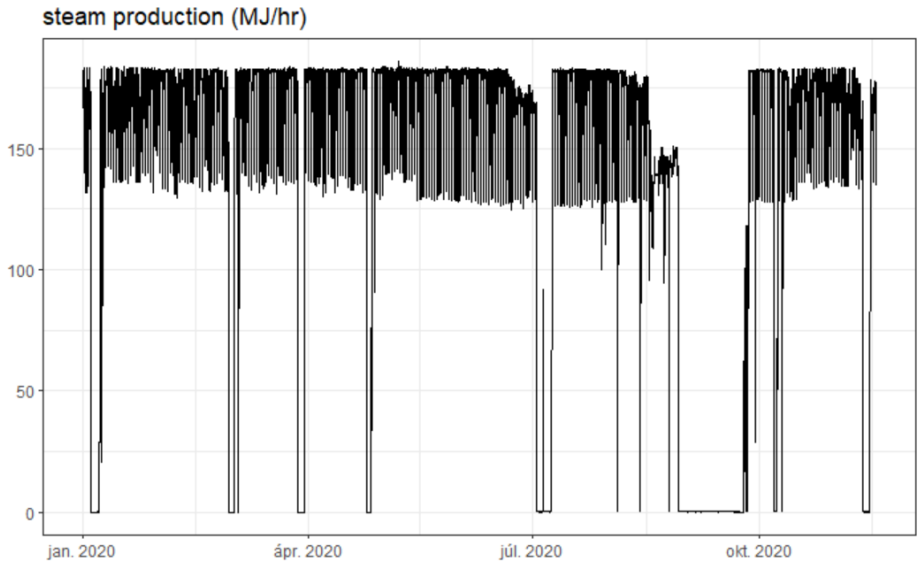

Figure 12 – Steam production timeseries

On the other hand, for some records in the dataset, the sum of production inputs measured by sensors (wood infeed A, wood infeed B, and oil infeed) was equal to 0, which seems to be the incorrect measurement, as it is impossible to generate electricity without using the mentioned production inputs; therefore, these cases were also removed from the dataset. This could be related to no input used at the moment and production was stopped for operational reasons. Figure 13 shows timeseries of the wood infeed of the production system. Some zero and near zero negative values are measured matching with the zero values of the steam output.

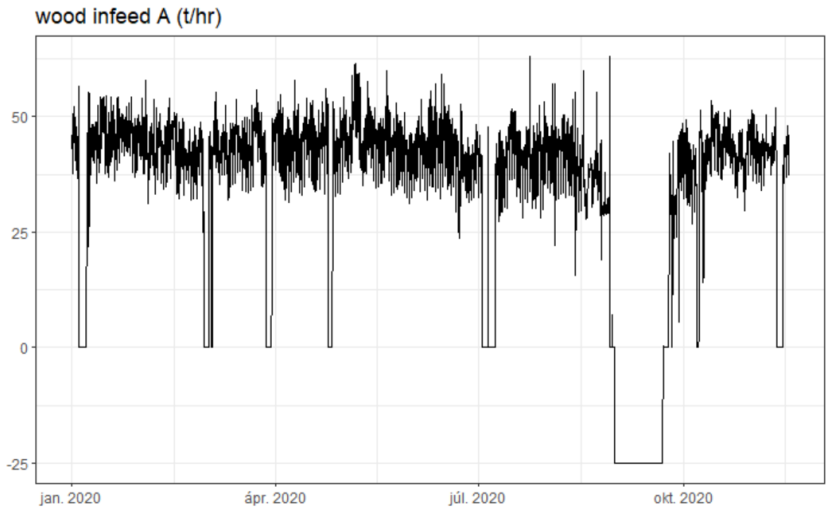

Figure 13 – Wood infeed timeseries

Figure 13 on the other hand shows the histograms of steam output which reveals considerable number of measurements with negative values and zero values, more than 1000 records. Those values were removed from the dataset considering it could be production paused, or simply defects on the sensors measurements.

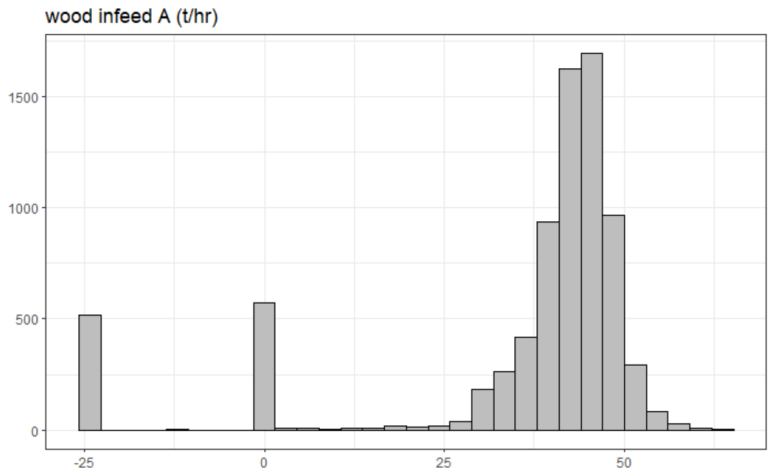

Figure 14 – Wood infeed histogram

Missing or erroneous data could be handled in several ways: if a value is incorrect, it can simply be removed; if data is missing, an average of the preceding and following values can be taken. Alternatively, data can be generated based on a cumulative distribution function (CDF) derived from the existing data or even interpolation could be done. However, considering that the dataset is a timeseries, but the time itself it not a relevant variable. In the specific case of this research, some sensor data contained nonsensical (e.g., negative) values, which were removed, as negative measurements were not meaningful in the given context. Due to the nature of the experimental setup, occasional missing data points were not critical, as the exact timing of measurements was not a determining factor for the outcomes.

On the other hand, oil infeed has a very different distribution as wood input Figure 15 shows the timeseries of oil infeed usage, showing that it is varying, apparently randomly from 0 to 3 kg/hr.

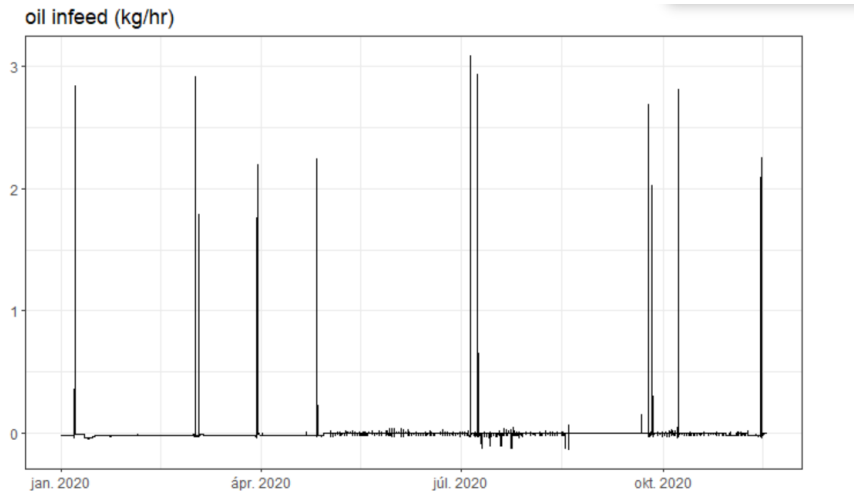

Figure 15 – Oil infeed timeseries

Similary, to other inputs variables there are some negative values which were removed from the dataset, however zero values are a possibility in the production system. So those were kept, only when the sum of wood infeed and oil were zero then this was considered a wrong value and removed from the final dataset.

Figure 16 shows the histogram of the Oil infeed usage, we can see some negative values are present and most of the time the oil usage is avoided so zero values of oil usage is a common approach in the production systems. However, there were sometime when high oil infeed was used.

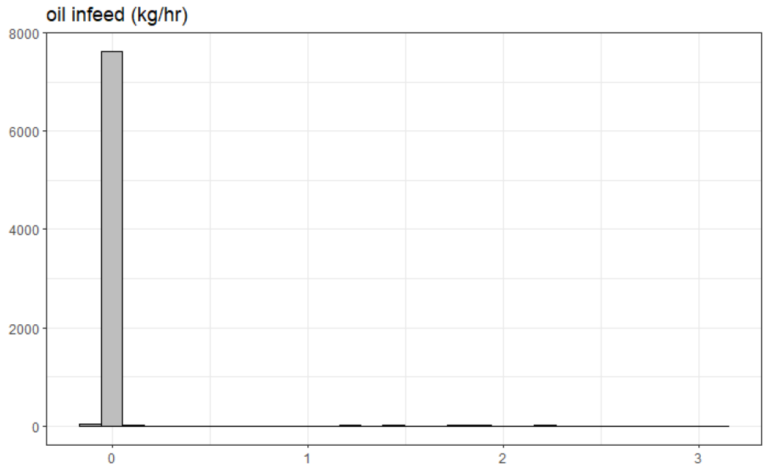

Figure 16 – Oil infeed histogram

Moreover, for more robust analysis, and considering only the production under normal conditions of operations, Outlier removal was applied by the Z score method for each set of sensor data types, resulting a removal of 231 records. The final cleaned dataset was used for analysis, and considering that for more robust analysis it is better to remove some noisy data from the dataset then trying to replace or predict this values using missing values replacement methods, which could result in non-real data and the ML-DT would not be a real representation of the production system, but some approximation.

8.3 Proposed AI Solution Adapted for Thermoelectric Production System

In this research, a unique combination of DEA, GA, and ML was implemented, creating a hybrid Artificial Intelligence solution capable of optimizing the production efficiency of a thermoelectric power plant using manufacturing sensor data. All components of the hybrid AI solution have their own purpose: DEA estimates the production system's efficient frontier and calculates DEA efficiency for all time periods - measured states, ML approximates the relationship between the input, output, process control parameters, and

state variables and predicts production efficiency through simulation, finally the GA proposes the optimal input and process control parameter settings corresponding to the desired output. Implementing such an AI solution can reduce production costs by optimizing resource usage and delivering economic and environmental benefits. The implementation of such a unique and new AI model can be utilized in other production plants, as a digital twin, providing decision support for the SCADA (supervisory control and data acquisition) system.

The proposed AI solution as applied to the thermoelectric power plant reads a dataset containing sensor measurements. Using domain expert knowledge, variables of the dataset were selected and classified into 4 data types: inputs, outputs, process control parameters, and state variables. Outliers were removed using the z score method followed by the DEA efficiency calculation. Constant return to scale with free disposability assumptions were used for the DEA efficiency calculations. The DMUs in the DEA consist of hourly measurements of inputs and outputs. For modelling the plant's behaviour, multiple ML models were created with their respective hyperparameters and pre-processing methods to predict production efficiencies for different inputs and process control parameters. Corresponding production outputs and state variables were also predicted by the ML models created. At the same time, the prediction of production efficiency was performed also considering the previously predicted output in addition to inputs and process control parameter values. All prediction ML Models were validated, and the validation are presented in section **8.4.2.** Figure 17 summarizes the DEA and ML model training phases of the AI solution.

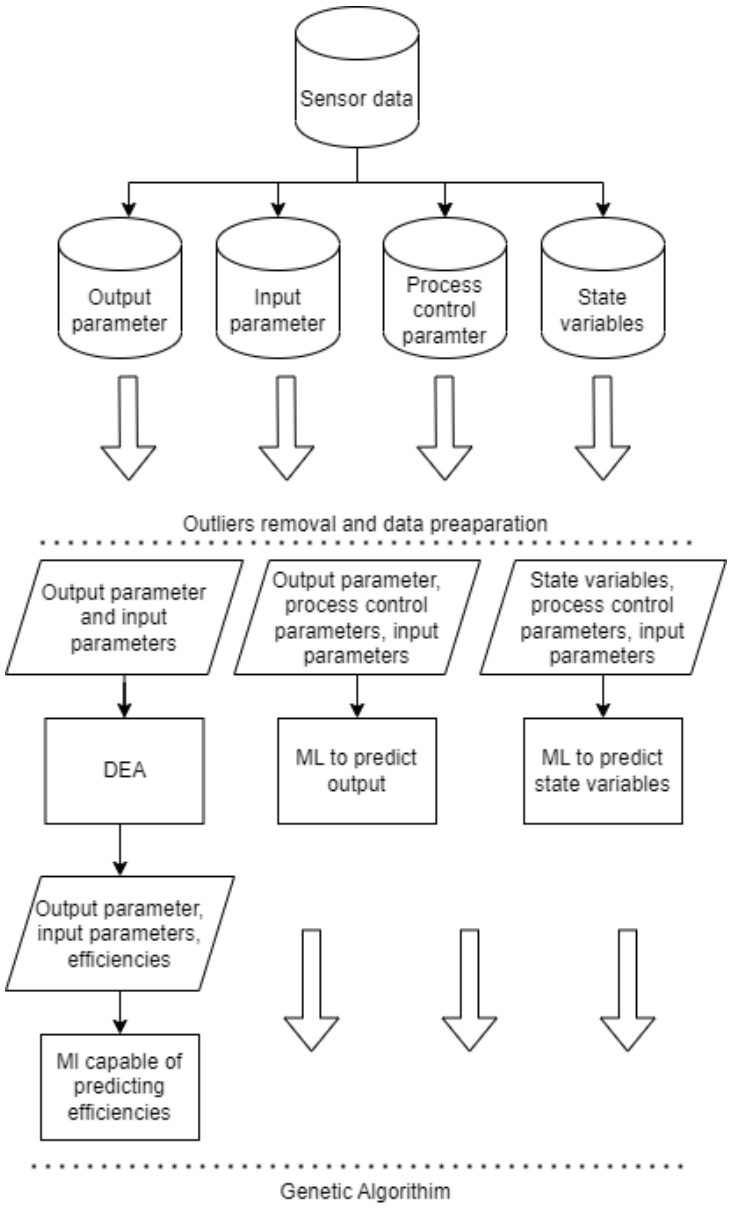

Figure 17 - DEA efficiency estimation and ML training phases of the hybrid AI solution

The GA component of the model searches for the feasible input and process control parameter combinations for any given output and predicts efficiencies to find the optimal one. The fitness function of the GA is the previously trained ML model, capable of predicting efficiencies. Additionally, state variables have a range of maximum and minimum desired values considered as constraints in the GA. For each solution generated by the GA, state variables were predicted using previously trained ML models, and solutions with state variable values beyond the desired range were considered unfeasible.

Furthermore, considering that in energy generation contracts, there are specific minimum values of energy to be generated and not achieving this minimum amount can lead to the breach of the contract, the predicted energy generation for each possible solution should always be greater than the contracted energy generation plan. Hence, if the GA solution predicts an energy generation slightly smaller than the desired/contracted output, this solution is considered infeasible. In contrast, if the predicted energy is greater than the desired/contracted output, the solution is considered feasible, and the excess energy production is considered waste and penalizes the efficiency of the solution. Figure 18 shows how the GA engine is programmed.

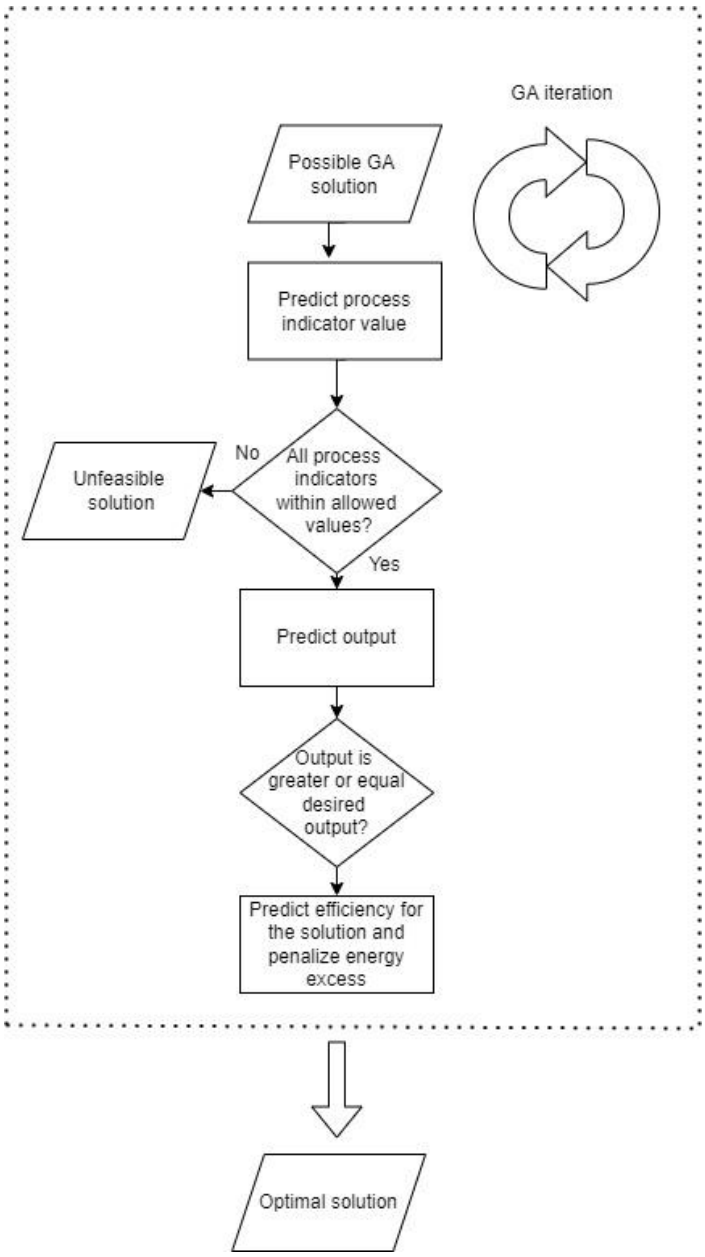

Figure 18 - The Genetic Algorithm optimisation phase of the hybrid AI solution

Exceeding energy generation is penalized according to Equation 4, where efficiency is equal to 1 minus the division of planned output by the predicted output.

$$e = 1 - (Planned/Predicted\ ) \quad (4)$$

The combination of the three main components of the model, DEA, ML, and GA, works as a powerful AI solution capable of predicting the relative production efficiencies and searching different settings and configurations to select the optimal one (Cavalcanti et al., 2024).

8.4 Results

8.4.1 DEA

For this research, DEA efficiency was calculated considering the inputs of the production system wood infeed A, wood infeed B, and oil infeed and the output the steam production for model 1 and electricity production for model 2. DMUs are measurements of inputs and output at a certain time, measured for each hour of the day. The result of DEA was appended to the raw dataset; hence, the dataset is extended to contain a new column with the DEA efficiency calculated for each record.

Since DEA only calculates relative efficiency, the best efficiency achieved may not mean the best possible efficiency, but the best efficiency identified in the historical dataset. Figure 19 shows the histogram of DEA relative efficiency scores throughout year 2020. The goal of the AI solution proposed in this research is to reduce efficiency loss and push it as much as possible to 100%. It is worth to note that the distribution of efficiencies for model 2, with electric output is bimodal. The distribution with lower efficiencies was for the winter

period. Further modelling may be required to separate the two distinct states and develop separate ML models.

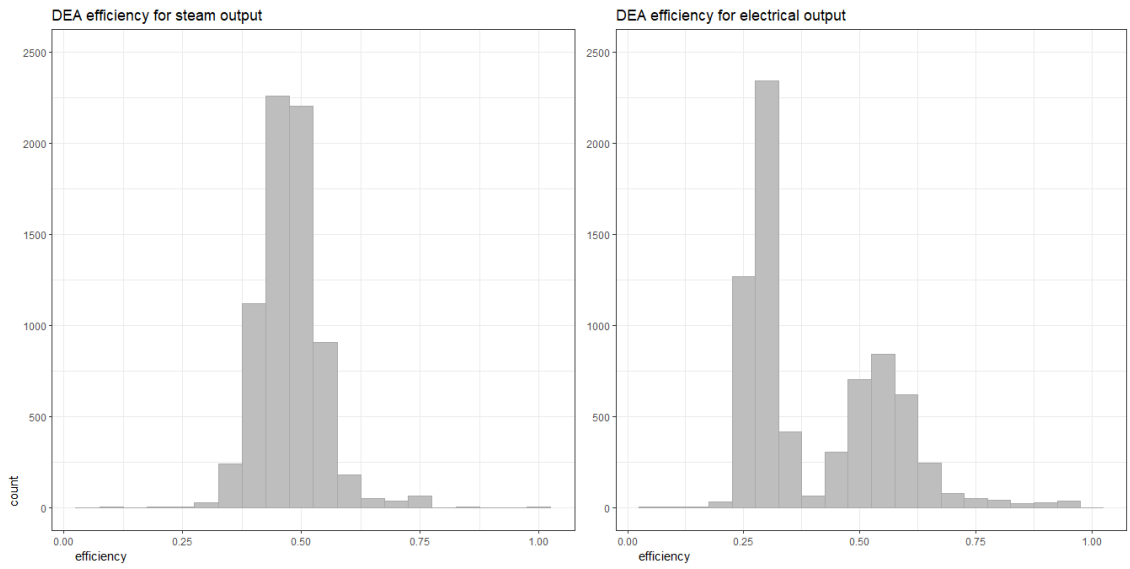

Figure 19 - DEA relative efficiency distributions of the two models for year 2020

The measure of the correlation coefficient provides information about the closeness of two variables; high/low correlations indicate high/low impact of the x variable on the y variable, whether causal or not (Senthilnathan, 2019). To obtain a better overview of how different sensor data can positively or negatively correlate with each other and with the DEA efficiency, correlations between them were obtained. Figure 20 shows the correlation coefficients between selected inputs, the output, control parameters, state variables, and DEA efficiency for model 1 and 2 respectively. It is possible to determine which parameters have the most impact on each other and on the relative efficiency score by obtaining the highest positive and negative correlation values in the table. To improve the visualization,

a red–blue colour scale was applied: blue means a higher positive correlation in contrast with red, representing high negative correlations.

The inputs wood and oil infeed have the strongest negative correlation with efficiency for both models; hence, a larger number of inputs resulted in lower efficiency. In addition to the input parameters, the process control parameters fluid bed primary air pressure (bar), burner secondary pressure (bar), burner tertiary air pressure (bar) showed a negative correlation with efficiency, in contrast with the burner primary air pressure (bar), which showed a slightly positive correlation with efficiency. The state variable flue gas temperatures showed a considerable positive correlation with efficiency, indicating that the efficiencies are higher when these gas temperatures are increased. Another interesting conclusion from the correlation table is that while the steam production is positively correlated with efficiency the correlation between electric output and DEA efficiency is almost insignificant. However, as the production system has output limitations, increasing production output deliberately is not possible.

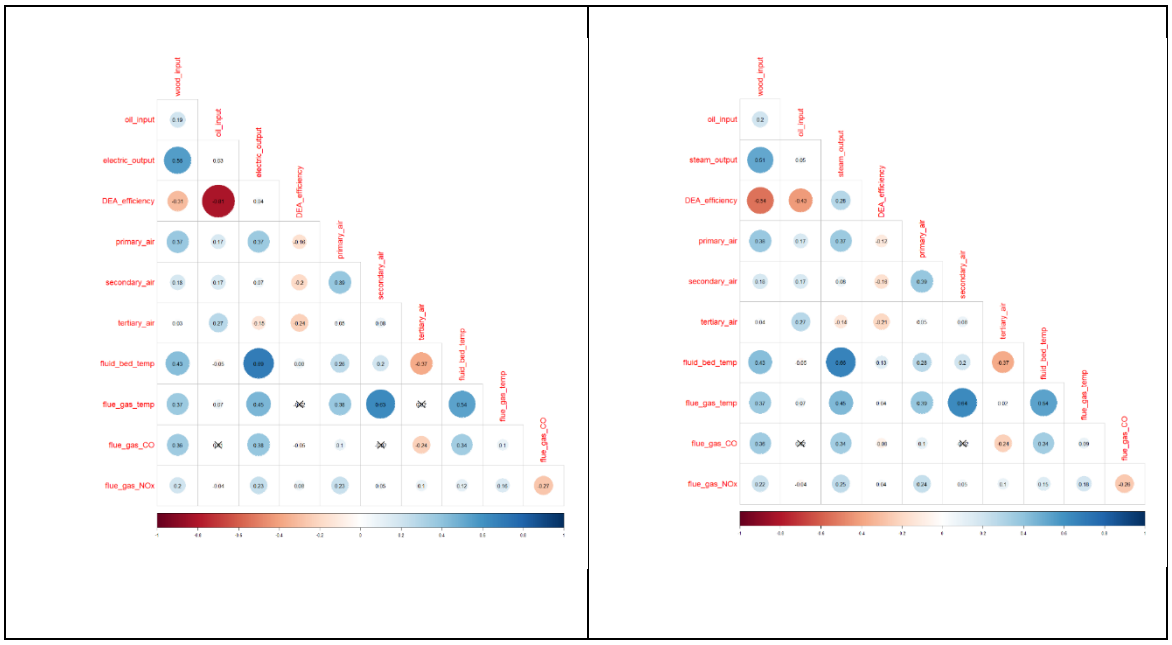

Figure 20 - Correlation table of selected input, output, process control and state variables of the two models

To better understand and illustrate how certain inputs, process control parameters and state variables interact, four dimensional charts were produced, including DEA efficiencies. Figure 21 illustrates how a process control parameter: the primary fluid bed air pressure affects DEA efficiency and the state variables: furnace fluid bed temperature and the NOx concentration of the flue gas (visualised with the colour scale ranging from dark brown through red to yellow). It looks like that the highest DEA efficiencies are achieved at a certain combinations of fluid bed pressure and fluid bed temperature with a corresponding, optimal flue gas NOx concentration. Both lower and higher NOx concentrations indicate lower efficiencies. The GA phase of the hybrid AI model will search for the optimal production setup and recommend input and process control parameters within the constraints of desired output and state variables.

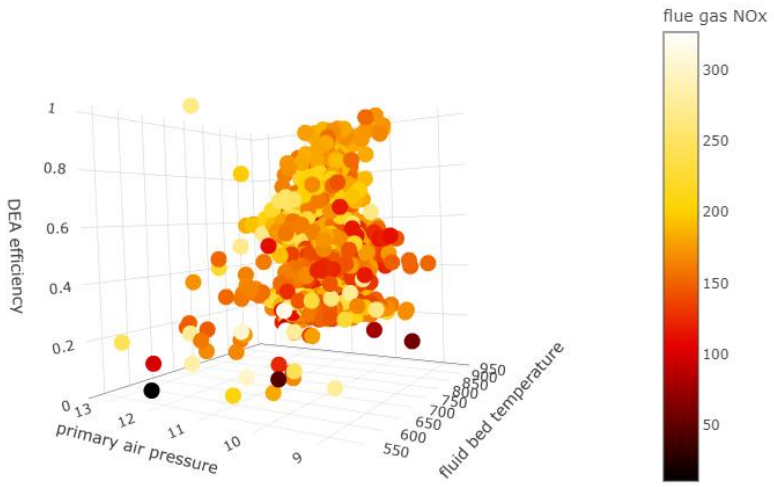

Figure 21 - Relationship between process control and state variables as well as DEA efficiency achieved.

To better understand how the inputs (Wood infeed A, Wood infeed B, and Oil infeed) affect production output (Steam production) and production efficiency, a four-dimensional graph was plotted. Figure 22 shows the efficiency frontier generated by the DEA of the 3 inputs and steam output against the efficiency represented in the colour scale. Wood infeed A and Wood infeed B were employed as one measure since they represent the feed of wood chips for burning, and the means of the two values were selected for dimensionality reduction and better visualization. Figure 22 shows a cluster of high-efficiency combinations of wood chip inflows for different output values. The efficiency frontier cluster, coloured on a bright-dark scale and represented by the brightest area of the plot, contains the settings of optimal inputs that the model will search for and recommend depending on the desired steam output. Process control parameters will also be considered in the recommendation; therefore, the final efficiency frontier calculated by the model will

include inputs and process control parameters in the GA search for the optimal production setup.

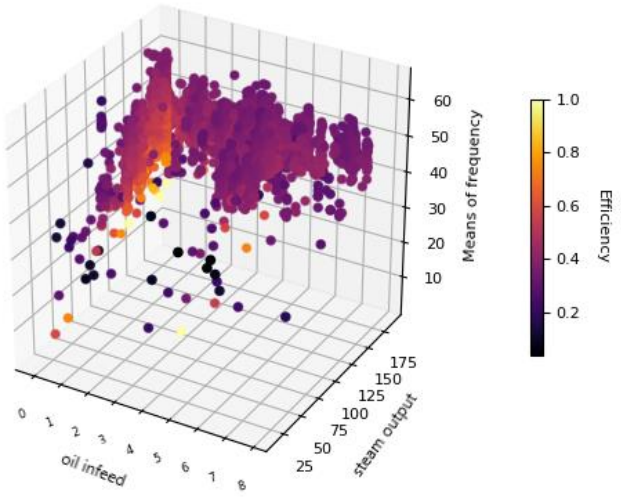

Figure 22 - Cluster of high-efficiency combinations of wood chip inflows

8.4.2 Machine Learning Models

After the DEA calculation, three kinds of ML models were created to predict production output, state variables, and production efficiency. Production output and state variables were predicted based on inputs and process control parameter values. On the other hand, the prediction of production efficiency was performed considering the previously predicted output in addition to inputs and process control parameter values.

The AI solution uses grid search to search various combinations of pre-processing methods (robust scaler, min-max scaler, standard scaler, max-abs scaler, normal quantile transformer, uniform quantile transformer, and power transformer), ML methods (gradient boosting, elastic net, and support vector machine), and hyperparameters' possible values. Grid search is a hyperparameter optimization method that performs a complete search over a given subset of the hyperparameter space and selects the best solution based on a performance score. This subset of parameters has a well-known possible range of values for each ML model, and the process of choosing the optimal value is an empirical process; therefore, a grid search of these possible values is a common practice (Shekar & Dagnew, 2019). The proposed AI solution extends the capability of grid search and uses it not only for hyperparameter optimization but also for ML and pre-processing method selection.

R-squared (R^2) was the chosen scoring method for the grid search for the optimal model. In addition, the mean squared error (MSE) was calculated to further understand the model's errors. The hyperparameter space used for the grid search varied depending on the ML model. Gradient boosting had possible hyperparameter learning rates of 0.05, 0.1, 0.15, and 0.2. Moreover, the elastic net model had possible hyperparameters alpha and L1 ratio, constituting a linear space from 0 to 1 and logarithm space from -1 and 1, respectively. The support vector machine, on the other hand, had c and gamma hyperparameter values of 0.1 or 2. All possible combinations of models, pre-processing methods, and hyperparameters were tested, R scores were measured for each possible combination, and the best scoring models were selected, producing a set of optimal models capable of predicting steam outputs, relative efficiencies, and state variables.

Gradient boost was the method that had the best performance for each trained model; hence, predicting efficiency, steam/electricity output and state variables will use gradient boosting as the base method. Table 1 shows the learning rate, optimal pre-processing

method, and R^2/MSE scores of each trained ML model. ML models for predicting state variables values can be used for both models 1 and 2, on the other hand, predicting efficiencies and outputs (steam and electricity) requires distinct ML models. ML models often operate as black boxes, and demonstrating the inner workings of a specific model was not within the scope of this thesis. The emphasis is placed on the framework itself, rather than on any particular ML model. The framework is designed to be flexible, allowing for the integration of various ML models depending on the use case.

The optimal pre-processing method had some variation; on the other hand, the learning rate had, in the majority, a value of 0.2 with a few cases of 0.15 R^2 scores. Some low R^2 values were observed for flue gas NOx conc., flue gas CO conc., steam pressure, and steam temperature. However, considering that the possible range of values constraining the production is quite large compared with the MSE and that the production constraints are flexible, meaning that certain deviations from the ideal range do not cause such an impact, the models can be considered to have sufficient performance for this specific production system. Since feature selection for the models was performed using expert domain knowledge, raising the efficiency of those MLs can be achieved by adding new unknown, relevant independent variables at the cost of more calculation time. Another way to raise the performance of ML models is to increase the possible values space for hyperparameters, such as the learning rate, or even to test different ML models. However, as previously mentioned, these ML model performances were considered sufficient for this specific case.

Dependent variable	Learning rate	Pre-processing method	R2	MSE
Steam pressure (bar)	0.20	Power Transformer	0.2808	6.9069
Steam temperature (°C)	0.20	Quantile Transformer (Uniform)	0.3029	5.4983
Furnace temperature A (°C)	0.20	Min-Max Scaler	0.6103	646.2882
Furnace temperature B (°C)	0.20	Power Transformer	0.5485	449.4722
Flue gas temperature A (°C)	0.20	Quantile Transformer (Normal)	0.5168	319.0069
Flue gas temperature B (°C)	0.20	Power Transformer	0.8686	27.0330
Flue gas NOx conc. (ppm)	0.20	Max-Abs Scaler	0.2363	473.2964
Flue gas CO conc. (ppm)	0.15	Quantile Transformer (Normal)	0.2533	252467.00
Pre-ECO flue gas temperature A (°C)	0.20	Min-Max Scaler	0.6534	71.1030
Pre-ECO flue gas temperature B (°C)	0.15	Power Transformer	0.6915	74.9431
Flue gas flow rate (m3/h)	0.20	Quantile Transformer (Normal)	0.5446	14.9431
Steam production (MJ/h)	0.20	Quantile Transformer (Normal)	0.5217	157.6350
Generated electricity (MW/hr)	0.20	Quantile Transformer (Normal)	0.6616	8.1102
Predict efficiency (model 1)	0.20	Max-Abs Scaler	0.9960	0.000082
Predict efficiency (model 2)	0.20	Max-Abs Scaler	0.9981	0.000065

Table 1 - ML model optimal parameters and performance

8.4.3 Genetic Algorithm

The final step of the model is the application of the GA. To run the GA, the min and max constraints of each input, process control parameter and state variable were needed, to create the gene space of the solution's possible values. Therefore, the GA only generated solutions with inputs and process control parameter values within the stated ranges.

For this research, a simulation was performed for a production demand of 168 MJ/h steam output for model 1 and electricity generation of 48 MW/hr for model 2. Each generation of the GA receives a set of chromosomes, of which contains a set of genes that in this case, have different values of production inputs and process control parameters. Moreover, the fitness function was calculated for each possible solution to optimize efficiency. First, the steam/electricity output is predicted using the previously created ML model. If the predicted steam output is lower than the desired output demanded (168 MJ/h for model 1 and 48 MW/hr for model 2), the fit value of the chromosome is 0. Otherwise, the fit value is calculated by predicting efficiency based on the chromosome's genes that contain both production inputs and process control parameters, and the predicted steam output calculated by the previously mentioned ML model. Excess energy generation that surpasses the desired output penalizes the fitness function according to Equation 4.

For model 1, GA iterated for 1000 generations, while a similar simulation for model 2 required a higher number of iterations, over 30000 generations. A mutation rate of 10% was applied in both cases. Figure 23 shows the evolution of the fitness value over the generations of model 1 and model 2 respectively.

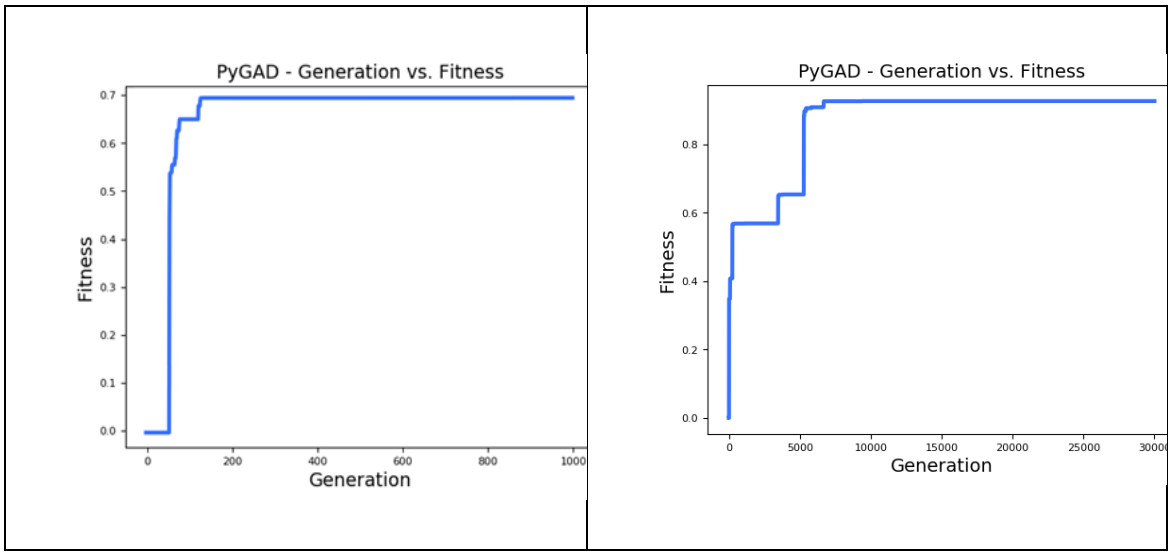

Figure 23 - Improvement of the fitness through generations of GA

As demonstrated in Figure 8, for model 1, by the end of the 1000th generation, the GA reaches an efficiency value of 70%, which is equivalent to the fitness value. Moreover, the best solution was achieved before the 200th generation indicating a quick convergence. Further generations have small or no improvement in production efficiency. On the other hand, model 2 reaches an efficiency of around 93% by the end of the 10000th generation indicating that longer time is required for convergence. Different mutation rates were used to test that the model did not converge to a local optimum. With different mutation rates, such as 10-20-30-25%, the algorithm approximately converged to the same efficiency value, reinforcing that the global optimum was potentially achieved (Hassanat et al., 2019b).

After convergence, the GA selects the best chromosome containing a set of optimal genes. Table 2 shows the results for each input and process control parameter recommended by the AI solution for optimal efficiency for models 1 and 2.

		Recommendation for optimal efficiency Model 1 for 168 MJ/h	Recommendation for optimal efficiency Model 2 for 48 MW/hr
Input	Wood infeed A (t/h)	32.4	28.30
	Wood infeed B (t/h)	39.6	28.56
	Oil infeed (kg/h)	0.0	0.0
Process control	Burner primary air pressure (bar)	0.0	0.00084
	Burner secondary pressure (bar)	0.3	0.28
	Burner tertiary air pressure (bar)	0.0	0.00091
	Fluid bed primary air pressure (bar)	12.2	10.73

Table 2 - Optimal recommendations for an output of 168 MJ/h and 48 MW/hr (model 1 and 2)

According to the proposed hybrid AI, to achieve optimal efficiency and produce a steam/electricity output of 168 MJ/h and 48 MW/hr for models 1 and 2 respectively, the inputs parameters and process control parameters should be set as shown in table 2. The GA also predicts the values of state variables, that is shown in Table 3 as an example for model 2.

State variable	Value for the optimal solution (model 1)	Value for the optimal solution (model 2)	Desired Min. value	Desired Max. value
Steam pressure (bar)	96.8	97.2	95	102
Steam temperature (°C)	532.1	534.3	532	535
Furnace temperature A (°C)	824.1	799.8	None	900
Furnace temperature B (°C)	824.4	787.9	None	900
Pre-ECO flue gas temperature A (°C)	432.2	414.2	None	None
Pre-ECO flue gas temperature B (°C)	434.4	430.7	None	None
Flue gas temperature A (°C)	150.2	181.3	120	None
Flue gas temperature B (°C)	137.9	123.7	120	None
Flue gas NOx conc. (ppm)	198.2	184.2	None	200
Flue gas CO conc. (ppm)	1083.2	348.6	None	5000
Flue gas flow rate (m3/h)	66.5	62.9	None	None

Table 3 - State variable values for the optimal solution

8.5 Discussion

For the previous example, model 1 achieved an efficiency of 70%, while model 2 achieved an efficiency of 93% exceeding the average that was achieved in the past, using a traditional, loop control-based method. Therefore, it indicates that the proposed hybrid AI method is superior to the traditional, loop control-based method. The past average efficiency for model 1when the steam output ranged from 167 to 169 MJ/h was 45.4% and for model 2 when the electricity output ranged from 47 to 49 MW/hr the efficiency achieved was 37%, reinforcing that the hybrid AI solution can increase efficiency (by 25% for model 1 and 58% for model 2) and reduce waste, if implemented. Furthermore, the predicted state variables for the recommended input and process control settings are all within the desired ranges, satisfying the production requirements (Table 3).

Once the DEA efficiency estimation and the ML training phases are completed, optimal input and process control settings can be calculated for multiple desired outputs. Therefore, GA phase can be decoupled from the DEA and ML model phases; and the GA can ben run multiple times to search for optimal solutions. This decoupling considerably reduces the calculation time, as the most time-consuming calculations are performed for the DEA and ML phases.

Given the scalability of the proposed AI framework, feature selection can be applied to ensure only the most relevant sensors are used. Additionally, the framework can leverage GPUs and cloud-based virtual machines with high computational power and memory to support large-scale operations. Although this may incur some costs, these can be offset by the resulting increases in production efficiency. With the rapid advancement and democratization of cloud technologies provided by major companies such as Microsoft, Amazon, Alibaba, Google, and others, access to powerful computing resources is becoming increasingly affordable. This trend supports the feasibility of managing an exponential increase in sensor data.

Moreover, the ML training component can be decoupled from the GA optimization process. Since the GA consumes significantly fewer computational resources, this decoupling provides flexibility for users to control how frequently they retrain their ML-based digital twin with new datasets.

Once the DEA efficiency estimation and ML training phases are completed, optimal input and process control settings can be computed for various desired outputs. The GA phase, being independent of DEA and ML, can then be executed multiple times to search for optimal solutions. This separation significantly reduces overall computation time, as the most resource-intensive tasks are completed in the DEA and ML stages.

Computation time was measured and is presented in Table 4. GA runtime increased notably as the number of generations rose from 1,000 to 30,000. To further reduce computation time during production, caching the results for common output ranges can help. Creating a lookup table of optimal settings for frequently requested outputs would eliminate the need to recalculate identical scenarios repeatedly.

It is also important to note that computation time may vary depending on the hardware used. For this research, all model testing was conducted on a standard commercial-grade notebook. Computation times of PID controller were not available for comparison.

Model	DEA Calculation Time	ML Model Calculation Time	GA Calculation Time
Model 1	2906.49s	1965.89s	248.86s
Model 2	2681.01s	2925.68s	3285.34s

Table 4 - Calculation time of the AI solution for model 1 and 2

Figure 24 shows the efficiency that could be achieved with the hybrid AI solution in comparison what was achieved using traditional loop control-based method, for different desired steam / electricity outputs for models 1 and 2. The dotted line shows the estimated efficiency that could be achieved using the hybrid AI solution that corresponds to the DEA efficiency frontier, while the dots represent the efficiency archived without it. The difference between the black and the grey dots is, that the grey ones represent observations, where constraints were violated for any of the state variables. Subsequently they were excluded from the solution space.

The area between the dots and the dotted line indicates how much efficiency increase could be achieved using the hybrid AI solution. It can be observed that the hybrid AI solution searches for the most efficient settings, that was achieved in the past and giving

recommendations that leads to the most efficient DMUs located on the efficiency frontier. The area above the dotted line represents the waste of efficiency regardless of the usage of the hybrid AI solution. As seen in Figure 24, the efficiency improvements are significant using the hybrid AI solution.

Moreover, it can be observed that there were multiple occasions when the production system operated in non-ideal conditions violating the state variables constraints, the AI solution not just increase efficiency but guarantees that the maximum efficiency is achieved without compromising state variables constraint. Investigating the area above the dotted line and trying different approaches to reduce it could lead to further improvements in production performance. Another interesting pattern visible on Figure 24 is that with the use of the hybrid AI solution, the correlation between efficiency and steam output diminishes, therefore low steam outputs can be as efficient or even more than running production at full capacity, producing high steam outputs. This gives more flexibility to the production system, allowing management to decide how much to produce without being concerned with getting penalized with higher waste.

This research therefore demonstrated the benefits of the hybrid AI system, being able to optimize input and control parameters settings, leading to more efficient production, and consequently reducing production costs and resource usage compared to the tradition loop control-based model. Considering that thermoelectric generated energy has a relatively high environmental impact, efficient usage of inputs can reduce CO_2 emissions, leading to a cleaner production system, benefiting individuals and the society.

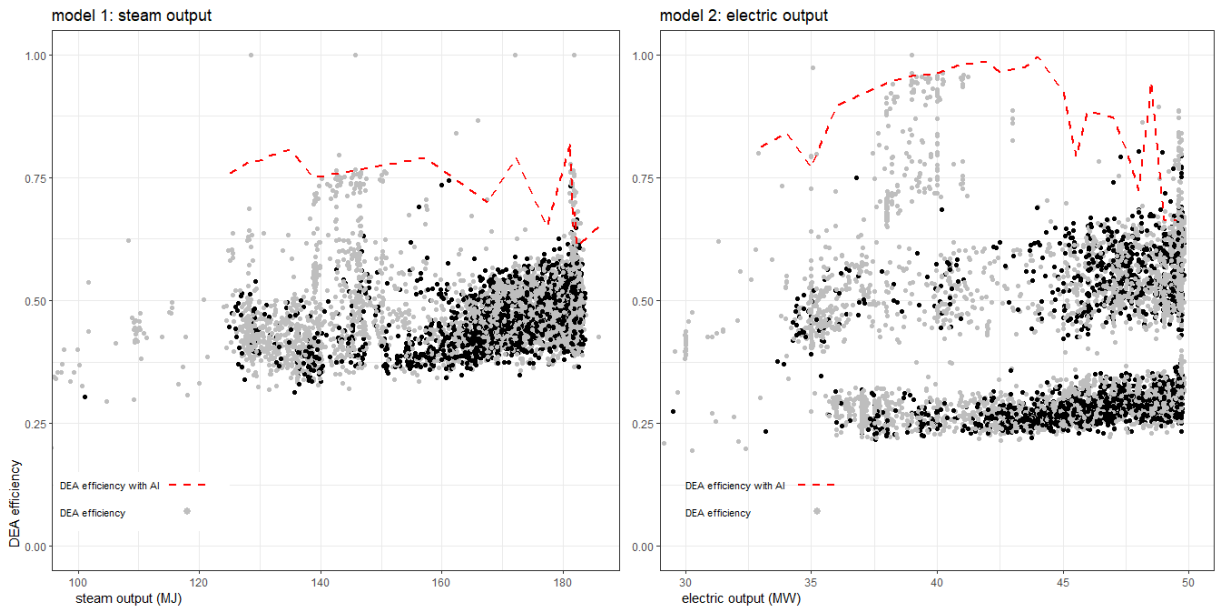

Figure 24 - Efficiencies with and without the AI solution for model 1 and 2 (grey points are observations violating state variable constraints)

9-Conclusion

This research presented a new integrated framework that combines GA, ML, DEA, and DT to enhance the intelligence and flexibility of production systems. Unlike other studies that used these techniques separately or in small combinations, this work proposed a comprehensive framework where all four methods work together in a balanced and connected manner. This combination allows the system not only to predict key indicators accurately but also to make intelligent suggestions based on current and past data.

A key part of the study was the detailed explanation of the AI framework's model. This clear description helped explain how DT technology, DEA, ML, and GA work together to optimize production efficiency. The mathematical details provided a deeper understanding of the model's workings, including the specific algorithms and equations used to optimize decision-making processes. This made it easier for researchers to reproduce the research and offered valuable insights for practitioners interested in the technical side of the framework. Moreover, a real-world example summary was provided, demonstrating the applicability of the framework.

To validate the effectiveness of the proposed solution, the framework was tested with real data from a thermal power plant. This type of environment is complex and represents many of the challenges faced by large industries. The results showed significant improvements, with efficiency gains ranging from 25% to 58%, compared to the traditional PID method. These gains were measured in terms of key production metrics such as throughput, energy consumption, and emissions. For example, the optimization process using GA, in conjunction with ML and DT, significantly reduced energy waste, improved operational efficiency, and lowered emissions, aligning with sustainability goals. These

improvements are important for reducing costs, increasing productivity, and supporting environmental goals such as reducing energy waste and emissions.

The traditional method used as the baseline in this research was the PID feedback control. The proposed AI framework was directly compared to this method, which served as the benchmark for measuring performance improvements. The results demonstrated that the AI-based solution outperformed PID, offering considerable improvements in efficiency and flexibility, which are critical aspects in production environments.

One significant advantage of the framework is its ease of adjustment. Companies can decide how frequently the models should be updated, helping to balance computational resources with the need for accurate and up-to-date results. This is especially useful in industries where the situation changes quickly. However, as datasets grow or models need to be retrained more frequently, the system may face limitations if the computing infrastructure is insufficient. To address this, future versions of the system could utilize cloud computing or edge computing, which would allow for faster and more distributed processing.

A limitation of the current model is that it primarily learns from past data, which was generated under human supervision and based on standard operational decisions. This means the model may not capture better ways of operating that have never been attempted before. To overcome this, it would be important to integrate DOE, which creates new and more varied scenarios. This would allow the model to learn from a broader range of situations, helping it find better and more creative solutions.

Future work could focus on exploring how the framework behaves with other optimization methods beyond GA. Techniques such as Reinforcement Learning, Bayesian

Optimization, or fuzzy logic could potentially yield even better results, especially in more complex or unpredictable situations. Comparing different optimization methods could help determine which one performs best depending on the type of production system.

Implementing the framework in real production environments is another key direction. This would allow for testing its applicability, feasibility, and effectiveness in practical settings. It would also show how the solution adapts to different production types, further confirming its reliability and usefulness in various contexts. Moreover, such studies can further investigate computational resource consumption.

Next steps should also involve applying the framework in other industries. The methodology could be beneficial in sectors such as oil and gas, food and beverages, automotive, pharmaceuticals, and renewable energy, where reliable and efficient systems are crucial. It would also be interesting to compare this framework with other modern solutions used in the industry, especially those different from the PID feedback loop, which was used as a baseline in this research. These comparisons could help identify which solution offers the best results under different objectives and constraints. Additionally, incorporating techniques that address uncertainties, and unexpected changes would make the system more robust and secure.

In summary, this study advances the creation of smarter and more adaptable production systems. The combination of GA, ML, DEA, and DT provides a solid foundation for making better decisions based on data. It helps operators take action based on clear evidence and also supports the improvement of performance and sustainability. The framework's flexible and scalable design means it can be applied even in environments with stringent requirements or rapidly changing conditions. As more industries undergo digital

transformation, tools like this will be key to creating smarter, more efficient, and future-ready production systems.

10-References

Agrawal, A., Fischer, M., & Singh, V. (2022). Digital twin: From concept to practice. *Journal of Management in Engineering*, *38*(3), 06022001.

Aheleroff, S., Xu, X., Zhong, R. Y., & Lu, Y. (2021). Digital twin as a service (DTaaS) in industry 4.0: An architecture reference model. *Advanced Engineering Informatics*, *47*, 101225.

Ahmed, S., Hasan, M. Z., MacLennan, M., Dorin, F., Ahmed, M. W., Hasan, M. M., Hasan, S. M., Islam, M. T., & Khan, J. A. M. (2019). Measuring the efficiency of health systems in Asia: a data envelopment analysis. *BMJ Open*, *9*(3), e022155. https://doi.org/10.1136/bmjopen-2018-022155

Al Bataineh, A., Kaur, D., & Jalali, S. M. J. (2022). Multi-Layer Perceptron Training Optimization Using Nature Inspired Computing. *IEEE Access*, *10*. https://doi.org/10.1109/ACCESS.2022.3164669

Azizi, A., & Azizi, A. (2019). Hybrid artificial intelligence optimization technique. *Applications of Artificial Intelligence Techniques in Industry 4.0*, 27–47.

Badnjević, A., Pokvić, L. G., Hasičić, M., Bandić, L., Mašetić, Z., Kovačević, Ž., Kevrić, J., & Pecchia, L. (2019). Evidence-based clinical engineering: machine learning algorithms for prediction of defibrillator performance. *Biomedical Signal Processing and Control*, *54*, 101629. https://doi.org/10.1016/j.bspc.2019.101629

Bahreini, M., Zarei, J., Razavi-Far, R., & Saif, M. (2021). Robust and reliable output feedback control for uncertain networked control systems against actuator faults. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *52*(4), 2555–2564.

Bai, Y., Chen, M., Zhou, P., Zhao, T., Lee, J., Kakade, S., Wang, H., & Xiong, C. (2021). How important is the train-validation split in meta-learning? *International Conference on Machine Learning*, 543–553.

Banker, R. D., Charnes, A., & Cooper, W. W. (1984). Some models for estimating technical and scale inefficiencies in data envelopment analysis. *Management Science*, *30*(9), 1078–1092. https://doi.org/10.1287/mnsc.30.9.1078

Batty, M. (2018). Digital twins. *Environment and Planning B: Urban Analytics and City Science*, *45*(5), 817–820. https://doi.org/10.1177/2399808318796416

Berente, N., Gu, B., Recker, J., & Santhanam, R. (2021). SPECIAL ISSUE: MANAGING AI MANAGING ARTIFICIAL INTELLIGENCE 1. *MIS Quarterly*, *45*(3).

Bharadiya, J. P. (2023). The role of machine learning in transforming business intelligence. *International Journal of Computing and Artificial Intelligence*, *4*(1), 16–24.

Borase, R. P., Maghade, D. K., Sondkar, S. Y., & Pawar, S. N. (2021). A review of PID control, tuning methods and applications. *International Journal of Dynamics and Control*, *9*, 818–827.

Botín-Sanabria, D. M., Mihaita, A.-S., Peimbert-García, R. E., Ramírez-Moreno, M. A., Ramírez-Mendoza, R. A., & Lozoya-Santos, J. de J. (2022). Digital twin technology challenges and applications: A comprehensive review. *Remote Sensing*, *14*(6), 1335.

Brownlee, J. (2020). *Data preparation for machine learning: data cleaning, feature selection, and data transforms in Python*. Machine Learning Mastery.

Budach, L., Feuerpfeil, M., Ihde, N., Nathansen, A., Noack, N., Patzlaff, H., Naumann, F., & Harmouch, H. (2022a). The effects of data quality on machine learning performance. *ArXiv Preprint ArXiv:2207.14529*.

Budach, L., Feuerpfeil, M., Ihde, N., Nathansen, A., Noack, N., Patzlaff, H., Naumann, F., & Harmouch, H. (2022b). The effects of data quality on machine learning performance. *ArXiv Preprint ArXiv:2207.14529*.

Cavalcanti, J. H., Kovács, T., & Kő, A. (2022). Production System Efficiency Optimization Using Sensor Data, Machine Learning-based Simulation and Genetic Algorithms. *Procedia CIRP*, *107*, 528–533.

Cavalcanti, J. H., Kovacs, T., Ko, A., & Pocsarovszky, K. (2024). Production system efficiency optimization through application of a hybrid artificial intelligence solution. *International Journal of Computer Integrated Manufacturing*, *37*(6), 790–807.

Charnes, A., Cooper, W. W., & Rhodes, E. (1978). Measuring the efficiency of decision making units. *European Journal of Operational Research*, *2*(6), 429–444. https://doi.org/10.1016/0377-2217(78)90138-8

Cheng, Y., Peng, J., Zhou, Z., Gu, X., & Liu, W. (2017). A hybrid DEA-adaboost model in supplier selection for fuzzy variable and multiple objectives. *IFAC-PapersOnLine*, *50*(1), 12255–12260. https://doi.org/10.1016/j.ifacol.2017.08.2038

Chia, K. S. (2018). Ziegler-nichols based proportional-integral-derivative controller for a line tracking robot. *Indonesian Journal of Electrical Engineering and Computer Science*, *9*(1), 221–226. https://doi.org/10.11591/ijeecs.v9.i1.pp221-226

Chicco, D., Warrens, M. J., & Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ Computer Science*, *7*, e623. https://doi.org/10.7717/peerj-cs.623

Chinnaiyan, B., Balasubaramanian, S., Jeyabalu, M., & Warrier, G. S. (2025). AI Applications–Computer Vision and Natural Language Processing. *Model Optimization Methods for Efficient and Edge AI: Federated Learning Architectures, Frameworks and Applications*, 25–41.

Chuang, K. V, & Keiser, M. J. (2018). Comment on "Predicting reaction performance in C–N cross-coupling using machine learning." *Science*, *362*(6416), eaat8603.

Dave, P. Y. (2020). The history of lean manufacturing by the view of Toyota-Ford. *International Journal of Scientific & Engineering Research*, *11*(8), 1598–1602.

de Hond, A. A. H., Leeuwenberg, A. M., Hooft, L., Kant, I. M. J., Nijman, S. W. J., van Os, H. J. A., Aardoom, J. J., Debray, T. P. A., Schuit, E., & van Smeden, M. (2022). Guidelines and quality criteria for artificial intelligence-based prediction models in healthcare: a scoping review. *NPJ Digital Medicine*, *5*(1), 2.

Di Noia, A., Martino, A., Montanari, P., & Rizzi, A. (2020). Supervised machine learning techniques and genetic optimization for occupational diseases risk prediction. *Soft Computing*, *24*(6), 4393–4406. https://doi.org/10.1007/s00500-019-04200-2

Diker, A., Sönmez, Y., Özyurt, F., Avcı, E., & Avcı, D. (2021). Examination of the ECG signal classification technique DEA-ELM using deep convolutional neural network features. *Multimedia Tools and Applications*, *80*, 24777–24800.

Dong, R., She, C., Hardjawana, W., Li, Y., & Vucetic, B. (2019). Deep learning for hybrid 5G services in mobile edge computing systems: Learn from a digital twin. *IEEE Transactions on Wireless Communications*, *18*(10), 4692–4707.

Dresch, A., Lacerda, D. P., Antunes Jr, J. A. V., Dresch, A., Lacerda, D. P., & Antunes, J. A. V. (2015). *Design science research*. Springer.

Elouataoui, W., El Mendili, S., & Gahi, Y. (2023). An Automated Big Data Quality Anomaly Correction Framework Using Predictive Analysis. *Data*, *8*(12), 182.

Eskandarpour, R., Khodaei, A., & Arab, A. (2017). Improving power grid resilience through predictive outage estimation. *2017 North American Power Symposium (NAPS)*, 1–5.

Fahim, M., Sharma, V., Cao, T.-V., Canberk, B., & Duong, T. Q. (2022). Machine learning-based digital twin for predictive modeling in wind turbines. *IEEE Access*, *10*, 14184–14194.

Grieves, M. (2014). Digital twin: manufacturing excellence through virtual factory replication. *White Paper*, *1*(2014), 1–7.

Guzmán, J. L., & Hägglund, T. (2024). Tuning rules for feedforward control from measurable disturbances combined with PID control: a review. *International Journal of Control*, *97*(1), 2–15.

Hafeez, G., Alimgeer, K. S., & Khan, I. (2020). Electric load forecasting based on deep learning and optimized by heuristic algorithm in smart grid. *Applied Energy*, *269*, 114915. https://doi.org/10.1016/j.apenergy.2020.114915

Hassanat, A., Almohammadi, K., Alkafaween, E., Abunawas, E., Hammouri, A., & Prasath, V. B. S. (2019a). Choosing mutation and crossover ratios for genetic algorithms—a review with a new dynamic approach. *Information*, *10*(12), 390.

Hassanat, A., Almohammadi, K., Alkafaween, E., Abunawas, E., Hammouri, A., & Prasath, V. B. S. (2019b). Choosing mutation and crossover ratios for genetic algorithms—a review with a new dynamic approach. *Information*, *10*(12), 390. https://doi.org/10.3390/info10120390

Hayat, A., Shahare, V., Sharma, A. K., & Arora, N. (2023). Introduction to industry 4.0. In *Blockchain and its Applications in Industry 4.0* (pp. 29–59). Springer.

He, Q., Wu, M., Liu, C., Jin, D., & Zhao, M. (2023). Management and real-time monitoring of interconnected energy hubs using digital twin: Machine learning based approach. *Solar Energy*, *250*, 173–181.

Hohenbichler, N. (2009). All stabilizing PID controllers for time delay systems. *Automatica*, *45*(11), 2678–2684.

Holland, J. H. (1975). Adaptation in Natural and Artificial Systems. ann arbor: University of michigan press. *Ann Arbor: The University of Michigan Press*.

Hong, H.-K., Leem, B., & Kim, S.-M. (2019). Using a Hybrid Model of DEA and Decision Tree Algorithm C5. 0 to Evaluate the Efficiency of Ports. *The Journal of the Korea Contents Association*, *19*(7), 99–109. https://doi.org/10.5392/JKCA.2019.19.07.099

Hossain, E., Roy, S., Mohammad, N., Nawar, N., & Dipta, D. R. (2021). Metrics and enhancement strategies for grid resilience and reliability during natural disasters. *Applied Energy*, *290*, 116709.

Hussain, A., Muhammad, Y. S., & Sajid, M. N. (2017). Performance evaluation of best–worst selection criteria for genetic algorithm. *Math Comput Sci*, *2*(6), 89–97.

Ibrahim, K. S. M. H., Huang, Y. F., Ahmed, A. N., Koo, C. H., & El-Shafie, A. (2022). A review of the hybrid artificial intelligence and optimization modelling of hydrological streamflow forecasting. *Alexandria Engineering Journal*, *61*(1), 279–303.

Jamwal, A., Agrawal, R., Sharma, M., & Giallanza, A. (2021). Industry 4.0 technologies for manufacturing sustainability: A systematic review and future research directions. *Applied Sciences*, *11*(12), 5725.

Jankovic, A., Chaudhary, G., & Goia, F. (2021). Designing the design of experiments (DOE)–An investigation on the influence of different factorial designs on the characterization of complex systems. *Energy and Buildings*, *250*, 111298.

Jeng, J.-C., & Lee, M.-W. (2023). Multi-loop PID controllers design with reduced loop interactions based on a frequency-domain direct synthesis method. *Journal of the Franklin Institute*, *360*(4), 2476–2506.

Jiang, Y., Yin, S., Li, K., Luo, H., & Kaynak, O. (2021). Industrial applications of digital twins. *Philosophical Transactions of the Royal Society A*, *379*(2207), 20200360.

Judge, M. A., Khan, A., Manzoor, A., & Khattak, H. A. (2022). Overview of smart grid implementation: Frameworks, impact, performance and challenges. *Journal of Energy Storage*, *49*, 104056.

Karin Kirk. (2022, October 24). *Energy loss is single-biggest component of today's electricity system*.

Katoch, S., Chauhan, S. S., & Kumar, V. (2021). A review on genetic algorithm: past, present, and future. *Multimedia Tools and Applications*, *80*, 8091–8126.

Kebonye, N. M. (2021). Exploring the novel support points-based split method on a soil dataset. *Measurement*, *186*, 110131.

Kherkhar, A., Chiba, Y., Tlemçani, A., & Mamur, H. (2022). Thermal investigation of a thermoelectric cooler based on Arduino and PID control approach. *Case Studies in Thermal Engineering*, *36*, 102249.

Kheyri, S., Lotfi, F. H., Najafi, S. E., & Parchkolaei, B. R. (2023). Presenting a predictive benchmark model of after-sales service agencies for vehicles based on the data envelopment analysis approach. *International Journal of Services and Operations Management*, *46*(1), 1–34.

Ko, T., Lee, J. H., Cho, H., Cho, S., Lee, W., & Lee, M. (2017). Machine learning-based anomaly detection via integration of manufacturing, inspection and after-sales service data. *Industrial Management & Data Systems*. https://doi.org/10.1108/IMDS-06-2016-0195

Koç, T., & Bayhan, N. (2024). Control of a thermoelectric cooling module by metaheuristic optimization algorithms. *Journal of Aeronautics and Space Technologies*, *17*(1), 89–106.

Kostov, B., & Hristov, V. (2023). Optimizing cycle time of Industrial robot for loading molding machine: A comprehensive analysis and optimization approach. *2023 5th International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, 1–5.

Król, J., & Ocłoń, P. (2018). Economic analysis of heat and electricity production in combined heat and power plant equipped with steam and water boilers and natural gas engines. *Energy Conversion and Management*, *176*, 11–29.

Laref, R., Losson, E., Sava, A., & Siadat, M. (2019). On the optimization of the support vector machine regression hyperparameters setting for gas sensors array applications. *Chemometrics and Intelligent Laboratory Systems*, *184*, 22–27. https://doi.org/10.1016/j.chemolab.2018.11.011

Lawal, A. I., Oniyide, G. O., Kwon, S., Onifade, M., Köken, E., & Ogunsola, N. O. (2021). Prediction of mechanical properties of coal from non-destructive properties: a comparative application of MARS, ANN, and GA. *Natural Resources Research*, *30*(6), 4547–4563. https://doi.org/10.1007/s11053-021-09955-w

Le Thi, H. A., Le, H. M., & Dinh, T. P. (2019). *Optimization of complex systems: theory, models, algorithms and applications* (Vol. 991). Springer.

Liashchynskyi, P., & Liashchynskyi, P. (2019). Grid search, random search, genetic algorithm: a big comparison for NAS. *ArXiv Preprint ArXiv:1912.06059*.

Liu, C., Le Roux, L., Körner, C., Tabaste, O., Lacan, F., & Bigot, S. (2022). Digital twin-enabled collaborative data management for metal additive manufacturing systems. *Journal of Manufacturing Systems, 62*, 857–874.

Liu, H., Wang, C., Ju, P., & Li, H. (2022). A sequentially preventive model enhancing power system resilience against extreme-weather-triggered failures. *Renewable and Sustainable Energy Reviews, 156*, 111945.

Liu, L., Huang, Y., & Zhan, X. (2019). The evolution of collective strategies in SMEs' innovation: a tripartite game analysis and application. *Complexity, 2019*. https://doi.org/10.1155/2019/9326489

Liu, Y., Dillon, T., Yu, W., Rahayu, W., & Mostafa, F. (2020). Missing Value Imputation for Industrial IoT Sensor Data with Large Gaps. *IEEE Internet of Things Journal*, *7*(8). https://doi.org/10.1109/JIOT.2020.2970467

Liu, Y., Zhang, J.-M., Min, Y.-T., Yu, Y., Lin, C., & Hu, Z.-Z. (2023). A digital twin-based framework for simulation and monitoring analysis of floating wind turbine structures. *Ocean Engineering, 283*, 115009.

Liu, Y.-Y., & Barabási, A.-L. (2016). Control principles of complex systems. *Reviews of Modern Physics*, *88*(3), 035006. https://doi.org/10.1103/RevModPhys.88.035006

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

Martišauskas, L., Augutis, J., Krikštolaitis, R., Urbonas, R., Šarūnienė, I., & Kopustinskas, V. (2022). A framework to assess the resilience of energy systems based on quantitative indicators. *Energies*, *15*(11), 4040.

Merei, G., Berger, C., & Sauer, D. U. (2013). Optimization of an off-grid hybrid PV–Wind–Diesel system with different battery technologies using genetic algorithm. *Solar Energy*, *97*, 460–473. https://doi.org/10.1016/j.solener.2013.08.016

Min, Q., Lu, Y., Liu, Z., Su, C., & Wang, B. (2019). Machine learning based digital twin framework for production optimization in petrochemical industry. *International Journal of Information Management*, *49*, 502–519.

Mokhtari, S., Navidi, W., & Mooney, M. (2020). White-box regression (elastic net) modeling of earth pressure balance shield machine advance rate. *Automation in Construction*, *115*, 103208. https://doi.org/10.1016/j.autcon.2020.103208

Montoya-Rincon, J. P., Mejia-Manrique, S. A., Azad, S., Ghandehari, M., Harmsen, E. W., Khanbilvardi, R., & Gonzalez-Cruz, J. E. (2023). A socio-technical approach for the assessment of critical infrastructure system vulnerability in extreme weather events. *Nature Energy*, *8*(9), 1002–1012.

Mosavi, A., Salimi, M., Faizollahzadeh Ardabili, S., Rabczuk, T., Shamshirband, S., & Varkonyi-Koczy, A. R. (2019). State of the art of machine learning models in energy

systems, a systematic review. *Energies*, *12*(7), 1301. https://doi.org/10.3390/en12071301

Moshayedi, A. J., Roy, A. S., & Liao, L. (2019). PID Tuning Method on AGV (automated guided vehicle) Industrial Robot. *Journal of Simulation and Analysis of Novel Technologies in Mechanical Engineering*, *12*(4), 53–66.

Moura Júnior, R. M. F. de. (2021). *Práticas colaborativas gamificadas para Prevenir Lesões por Pressão.*

Mustafa, F. S., Khan, R. U., & Mustafa, T. (2021). Technical efficiency comparison of container ports in Asian and Middle East region using DEA. *The Asian Journal of Shipping and Logistics*, *37*(1), 12–19.

Nagunwa, T. (2024). Comparative Analysis of Nature-Inspired Metaheuristic Techniques for Optimizing Phishing Website Detection. *Analytics*, *3*(3), 344–367.

Nasir, M. T., Afaneh, D., & Abdallah, S. (2022). Design Modifications for a Thermoelectric Distiller with Feedback Control. *Energies*, *15*(24), 9612.

Nayyar, A., Gadhavi, L., & Zaman, N. (2021). Machine learning in healthcare: review, opportunities and challenges. *Machine Learning and the Internet of Medical Things in Healthcare*, 23–45.

Nguyen, T.-H., Do, T.-T., Nguyen, D.-N., Lu, D.-N., & Nguyen, H.-N. (2020). A Hybrid Method Based on Genetic Algorithm and Ant Colony System for Traffic Routing Optimization. *VNU Journal of Science: Computer Science and Communication Engineering*, *36*(1). https://doi.org/10.25073/2588-1086/vnucsce.236

Nnamoko, N., & Korkontzelos, I. (2020). Efficient treatment of outliers and class imbalance for diabetes prediction. *Artificial Intelligence in Medicine*, *104*, 101815.

Orong, M. Y., Sison, A. M., & Medina, R. P. (2018). A new crossover mechanism for genetic algorithm with rank-based selection method. *Proceedings of 2018 5th International Conference on Business and Industrial Research: Smart Technology for Next Generation of Information, Engineering, Business and Social Science, ICBIR 2018*. https://doi.org/10.1109/ICBIR.2018.8391171

Pagliosa, M., Tortorella, G., & Ferreira, J. C. E. (2021). Industry 4.0 and Lean Manufacturing: A systematic literature review and future research directions. *Journal of Manufacturing Technology Management*, *32*(3), 543–569.

Pakuhinezhad, O., & Atrian, A. (2024). Unraveling the Tapestry of Artificial Intelligence: From Myth to Reality, Ethics to Economics. *Ethics to Economics (April 7, 2024)*.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., & Dubourg, V. (2011). Scikit-learn: Machine learning in Python. *The Journal of Machine Learning Research*, *12*, 2825–2830.

Perez, H., & Tah, J. H. M. (2020). Improving the accuracy of convolutional neural networks by identifying and removing outlier images in datasets using t-SNE. *Mathematics*, *8*(5), 662.

Priya Varshini, A. G., & Anitha Kumari, K. (2020). Predictive analytics approaches for software effort estimation: A review. *Indian J. Sci. Technol*, *13*, 2094–2103.

Quiroz-Flores, J. C., & Vega-Alvites, M. L. (2022). Review lean manufacturing model of production management under the preventive maintenance approach to improve efficiency in plastics industry smes: a case study. *South African Journal of Industrial Engineering*, *33*(2), 143–156.

Rácz, A., Bajusz, D., & Héberger, K. (2021). Effect of dataset size and train/test split ratios in QSAR/QSPR multiclass classification. *Molecules*, *26*(4), 1111.

Rai, P. R., Nanjundan, P., & George, J. P. (n.d.). Enhancing Industrial Operations through AI-Driven Decision-Making in the Era of Industry 4.0. In *AI-Driven IoT Systems for Industry 4.0* (pp. 42–55). CRC Press.

Raith, A., Rouse, P., & Seiford, L. M. (2019). Benchmarking Using Data Envelopment Analysis: Application to Stores of a Post and Banking Business. In *Multiple Criteria Decision Making and Aiding* (pp. 1–39). Springer. https://doi.org/10.1007/978-3-319-99304-1_1

Rajakumaran, S. (2023). *Fundamentals Of AI And Deep Learning*. Academic Guru Publishing House.

Rane, N. (2023). Integrating leading-edge artificial intelligence (AI), internet of things (IOT), and big data technologies for smart and sustainable architecture, engineering and construction (AEC) industry: Challenges and future directions. *Engineering and Construction (AEC) Industry: Challenges and Future Directions (September 24, 2023)*.

Raza, M. Q., & Khosravi, A. (2015). A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings. *Renewable and Sustainable Energy Reviews*, *50*, 1352–1372.

Razzaghi, M. (2009). Optimization of time delay systems by hybrid functions. *Optimization and Engineering*, *10*, 363–376.

Redell, N. (2019). Shapley decomposition of R-squared in machine learning models. *ArXiv Preprint ArXiv:1908.09718*.

Román-Ramírez, L. A., & Marco, J. (2022). Design of experiments applied to lithium-ion batteries: A literature review. *Applied Energy*, *320*, 119305.

Russell, S. J., & Norvig, P. (2016). *Artificial intelligence: a modern approach*. Pearson.

Salehi, V., Veitch, B., & Musharraf, M. (2020). Measuring and improving adaptive capacity in resilient systems by means of an integrated DEA-Machine learning approach. *Applied Ergonomics*, *82*, 102975. https://doi.org/10.1016/j.apergo.2019.102975

Sanfilippo, F., Hua, T. M., & Bos, S. (2020). A comparison between a two feedback control loop and a reinforcement learning algorithm for compliant low-cost series elastic actuators. *Proceeding of the 53rd Hawaii International Conference on System Sciences (HICSS 2020)*.

Schwaber, K. (1997). Scrum development process. *Business Object Design and Implementation: OOPSLA'95 Workshop Proceedings 16 October 1995, Austin, Texas*, 117–134.

Segovia, M., & Garcia-Alfaro, J. (2022). Design, modeling and implementation of digital twins. *Sensors*, *22*(14), 5396.

Senthilnathan, S. (2019). Usefulness of correlation analysis. *Available at SSRN 3416918*. https://doi.org/10.2139/ssrn.3416918

Shajahan, M. S. M., Jamal, D. N., Mathew, J., Akbar, A. A. A., Sivakumar, A., & Hameed, M. S. S. (2022). Improvement in efficiency of thermal power plant using optimization and robust controller. *Case Studies in Thermal Engineering*, *33*, 101891.

Shaw, M., Rights, J. D., Sterba, S. S., & Flake, J. K. (2023). r2mlm: An R package calculating R-squared measures for multilevel models. *Behavior Research Methods*, *55*(4), 1942–1964.

Shekar, B. H., & Dagnew, G. (2019). Grid search-based hyperparameter tuning and classification of microarray cancer data. *2019 Second International Conference on Advanced Computational and Communication Paradigms (ICACCP)*, 1–8. https://doi.org/10.1109/ICACCP.2019.8882943

Sheuly, S. S., Ahmed, M. U., & Begum, S. (2022). Machine-learning-based digital twin in manufacturing: A bibliometric analysis and evolutionary overview. *Applied Sciences*, *12*(13), 6512.

Silva, G. J., Datta, A., & Bhattacharyya, S. P. (2007). *PID controllers for time-delay systems*. Springer Science & Business Media.

Singh, D., & Singh, B. (2022). Feature wise normalization: An effective way of normalizing data. *Pattern Recognition*, *122*, 108307. https://doi.org/10.1016/j.patcog.2021.108307

Sivanandam, S. N., Deepa, S. N., Sivanandam, S. N., & Deepa, S. N. (2008). Genetic algorithm optimization problems. *Introduction to Genetic Algorithms*, 165–209.

Somefun, O. A., Akingbade, K., & Dahunsi, F. (2021). The dilemma of PID tuning. *Annual Reviews in Control*, *52*, 65–74.

Sommerville, I. (2020). *Engineering software products* (Vol. 355). Pearson London.

Soori, M., Dastres, R., Arezoo, B., & Jough, F. K. G. (2024). Intelligent robotic systems in Industry 4.0: A review. *Journal of Advanced Manufacturing Science and Technology*, 2024007.

Taherinezhad, A., & Alinezhad, A. (2023). Nations performance evaluation during SARS-CoV-2 outbreak handling via data envelopment analysis and machine learning methods. *International Journal of Systems Science: Operations & Logistics*, *10*(1), 2022243.

Tao, F., Xiao, B., Qi, Q., Cheng, J., & Ji, P. (2022). Digital twin modeling. *Journal of Manufacturing Systems*, *64*, 372–389.

Touzani, S., Granderson, J., & Fernandes, S. (2018). Gradient boosting machine for modeling the energy consumption of commercial buildings. *Energy and Buildings*, *158*, 1533–1543. https://doi.org/10.1016/j.enbuild.2017.11.039

Umunnakwe, A., Huang, H., Oikonomou, K., & Davis, K. R. (2021). Quantitative analysis of power systems resilience: Standardization, categorizations, and challenges. *Renewable and Sustainable Energy Reviews*, *149*, 111252.

Vabalas, A., Gowen, E., Poliakoff, E., & Casson, A. J. (2019). Machine learning algorithm validation with a limited sample size. *PloS One*, *14*(11), e0224365.

VanDerHorn, E., & Mahadevan, S. (2021). Digital Twin: Generalization, characterization and implementation. *Decision Support Systems*, *145*, 113524.

Vie, A., Kleinnijenhuis, A. M., & Farmer, D. J. (2020). Qualities, challenges and future of genetic algorithms: a literature review. *ArXiv Preprint ArXiv:2011.05277*.

vom Brocke, J., Hevner, A., & Maedche, A. (2020). Introduction to design science research. *Design Science Research. Cases*, 1–13.

Wan, Z., Xu, Y., & Šavija, B. (2021). On the use of machine learning models for prediction of compressive strength of concrete: influence of dimensionality reduction on the model performance. *Materials*, *14*(4), 713.

Wang, C., Ju, P., Wu, F., Pan, X., & Wang, Z. (2022). A systematic review on power system resilience from the perspective of generation, network, and load. *Renewable and Sustainable Energy Reviews*, *167*, 112567.

Wang, J., Wang, X., Ma, C., & Kou, L. (2021). A survey on the development status and application prospects of knowledge graph in smart grids. *IET Generation, Transmission & Distribution*, *15*(3), 383–407.

Wang, X., Jiang, W., & Luo, Z. (2016). Combination of convolutional and recurrent neural network for sentiment analysis of short texts. *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 2428–2437.

Wang, X., & Li, R. (2022). Physical Exercise Effect Evaluation Based on Data Envelopment Analysis. *International Journal of Education and Humanities*, *5*(1), 59–63.

Wang, Y., Zhang, S., Wu, S., Zhou, Y., Du, J., & Li, H. (2022). A Digital Twin Model of Smart Factory Production System. *International Conference on Image, Vision and Intelligent Systems*, 960–973.

Wang, Z., Ding, T., Jia, W., Huang, C., Mu, C., Qu, M., Shahidehpour, M., Yang, Y., Blaabjerg, F., & Li, L. (2022). Multi-stage stochastic programming for resilient integrated electricity and natural gas distribution systems against typhoon natural disaster attacks. *Renewable and Sustainable Energy Reviews*, *159*, 111784.

Weise, T. (2009). Global optimization algorithms-theory and application. *Self-Published Thomas Weise*, *361*, 153.

Wen, L., Zhou, K., Yang, S., & Lu, X. (2019). Optimal load dispatch of community microgrid with deep learning based solar power and load forecasting. *Energy*, *171*, 1053–1065. https://doi.org/10.1016/j.energy.2019.01.075

Xie, J., Alvarez-Fernandez, I., & Sun, W. (2020). A review of machine learning applications in power system resilience. *2020 IEEE Power & Energy Society General Meeting (PESGM)*, 1–5.

Xu, K., & Pérez-Arancibia, N. O. (2020). Electronics-free logic circuits for localized feedback control of multi-actuator soft robots. *IEEE Robotics and Automation Letters*, *5*(3), 3990–3997.

Xu, X., Wu, Y., Zuo, L., & Chen, S. (2019). Multimaterial topology optimization of thermoelectric generators. *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, *59186*, V02AT03A064. https://doi.org/10.1115/DETC2019-97934

Yan, Z., Zhou, W., Wang, Y., & Chen, X. (2022). Comprehensive Analysis of Grain Production Based on Three-Stage Super-SBM DEA and Machine Learning in Hexi Corridor, China. *Sustainability*, *14*(14), 8881.

Yang, W., Xiang, W., Yang, Y., & Cheng, P. (2022). Optimizing federated learning with deep reinforcement learning for digital twin empowered industrial IoT. *IEEE Transactions on Industrial Informatics*, *19*(2), 1884–1893.

Zhang, D., & Gao, X. (2022). A digital twin dosing system for iron reverse flotation. *Journal of Manufacturing Systems*, *63*, 238–249.

Zhang, Q., Wang, Z., Ma, S., & Arif, A. (2021). Stochastic pre-event preparation for enhancing resilience of distribution systems. *Renewable and Sustainable Energy Reviews*, *152*, 111636.

Zhang, Z., Zhao, Y., Canes, A., Steinberg, D., & Lyashevska, O. (2019). Predictive analytics with gradient boosting in clinical medicine. *Annals of Translational Medicine*, *7*(7).

Zhou, J., Li, L., Vajdi, A., Zhou, X., & Wu, Z. (2021). Temperature-constrained reliability optimization of industrial cyber-physical systems using machine learning and feedback control. *IEEE Transactions on Automation Science and Engineering*, *20*(1), 20–31.

Zhou, Y., Zhou, N., Gong, L., & Jiang, M. (2020). Prediction of photovoltaic power output based on similar day analysis, genetic algorithm and extreme learning machine. *Energy*, *204*, 117894. https://doi.org/10.1016/j.energy.2020.117894

Zhu, N., Zhu, C., & Emrouznejad, A. (2020). A combined machine learning algorithms and DEA method for measuring and predicting the efficiency of Chinese manufacturing listed companies. *Journal of Management Science and Engineering*. https://doi.org/10.1016/j.jmse.2020.10.001

Ziegler, J. G., & Nichols, N. B. (1942). Optimum settings for automatic controllers. *Transactions of the American Society of Mechanical Engineers*, *64*(8), 759–765.