Flexible Services and Manufacturing Journal
https://doi.org/10.1007/s10696-024-09537-x

A machine learning framework for explainable knowledge mining and production, maintenance, and quality control optimization in flexible circular manufacturing systems

Georgios K. Koulinas[1] · Panagiotis D. Paraschos[1] · Dimitrios E. Koulouriotis[2]

Accepted: 29 February 2024

Abstract

In the present study, we employed multiple decision tree algorithms to categorize cases and reflect the most efficient policies constructed by a reinforcement learning algorithm. These approaches treated a complex production, maintenance, and quality control optimization problem within a degrading manufacturing and remanufacturing system. The decision trees' nodes represent the independent variables, while the trees' leaves represent the set of function values. The reinforcement learning method revealed all optimization parameters and best policies, which were employed as the training sample for the tree algorithms. After constructing every decision tree, each resulting decision rule was used to solve the optimization problem, and its performance was assessed. Additionally, we performed a sensitivity analysis to determine if the pruning level impacts the objective function value and, generally, the effectiveness of the proposed approach.

Keywords Decision trees · Reinforcement learning · Manufacturing/remanufacturing systems · Data mining

1 Introduction

Modern manufacturing companies endeavor to achieve long-term sustainability, tackling challenges, e.g., customer retention (Galletta et al. 2018), imposed by the market itself. In this effort, they constantly introduce new merchandise in an attempt to sustain, or increase their market share (Palsodkar et al. 2023). For the merchandise, they integrate supply chains that facilitate the material flow between integrated

✉ Georgios K. Koulinas
gkoulina@pme.duth.gr

[1] Department of Production and Management Engineering, Democritus University of Thrace, Xanthi, Greece

[2] School of Mechanical Engineering, National Technical University of Athen, Heroon Polytechniou 9, Zografou, Athens, Greece, 15780 Athens, Greece

Published online: 03 April 2024

facilities and machines. In this respect, the supply chain management integrates a diversity of operations, including raw material procurement, product storage and delivery to customers. These operations could be well-supported by the digitization of the manufacturing operations and processes encouraged by current manufacturing concepts, such as Industry 4.0 (Dohale et al. 2023). However, despite this digitization, they are still ad-hoc controlled through operation strategies. By definition, these strategies, e.g., maintenance strategies (O'Donovan et al. 2015), intend to enrich the productivity and maintainability of manufacturing environments, yielding a better financial performance for them. They are formulated by strategists according to datasets captured from the integrated processes and operations exploiting intelligent systems, including cyber-physical ones (Hofmann and Rüsch 2017). Due to their large size, their assessment and process is rather complex to be performed with typical data processing methods and tools (Zhong et al. 2017). To this end, the aforementioned systems incorporate machine learning methodologies to extract manufacturing knowledge (Ahuett-Garza and Kurfess 2018).

In addition, machine learning approaches employed for scheduling operations in manufacturing systems. In the study of Tan et al. (2019) introduced a multi-agent model and a planning and scheduling algorithm, which enables manufacturing robots to work in coordination. Chen et al. (2019a, b) proposed a model that scales well for semi-supervised learning to forecast energy consumption with a few labelled data by modeling and compensating the unlabeled data. Recently, Agarwal et al. (2020) showed how deep learning can be used to find the input variables that maximize process profit or other non-linear process objectives such as quality.

Recent reviews can be found in Kang et al. (2020) that have grouped and analyzed the published research on the use of machine learning in production lines, and in the study of Dogan and Birant (2021) which provides a detailed review of literature in order to give an overview of how machine learning methods may be used to create intelligent manufacturing processes. Additionally, this study identifies many critical research issues that remain unresolved in the current literature on the same subject.

Machine learning based algorithms have been recognized as effective for knowledge mining and making predictions while treating complex production management problems and performing manufacturing systems optimization. In the recent study of Paraschos et al. (2021), a novel two-agent reinforcement learning-based approach proposed that includes parametric production and maintenance operations to increase the efficiency of the system. Also, Paraschos et al. (2022) introduced a novel framework to obtain joint policies for the authorization of production, recycling, maintenance and remanufacturing activities within deteriorating circular manufacturing systems. This framework uses a reinforcement learning algorithm and ad-hoc production control policies.

Algorithms based on machine learning, and specifically decision tree algorithms, proved to be efficient for knowledge mining and predictive modeling (Wu et al. 2008). In such kind of problems, a decision tree method uses parameters connected with the problem's input variables to construct a model, and then, estimate the values of the dependent variable. A decision tree is created by breaking a dataset into structural components and describing the links between the features and the dependent variable in a decision tree. Decision trees are a relatively prevalent classification approach due to their potential to handle a huge range of parameters, both numerical and nominal, while generating a fairly clear result. Despite the potential merits, decision tree algorithms have been criticized as they are often found to be inefficient when providing predictions, owing to their nature.

Recently, decision trees used by academics for treating a wide range of problems in supply chains and manufacturing. More specifically, in the study of Soeprapto Putri et al. (2018) proposed and implemented a decision tree model for defect classification, which provides a logical basis for the classifications, can be used for root cause analysis of defects, and supports finding similar defects. In Antosz et al. (2020), rough sets theory and decision trees were employed to decrease supply chain service level failures during the implementation of the lean maintenance concept to increase operational efficiency. Additionally, Lyu et al. (2020) used a decision tree to model a process resulting in defective items and extract rules for recognizing defective batches and their corresponding production process features, while (Mahato and Narayan 2020) were employed a gradient-boosted decision trees to model and train supply chains to prevent service level failures. Also, Zangaro et al. (2020), used a Classification and Regression Tree (CART)-based decision tree to decide which manufacturing line to use for the delivery of components for assembly to help humans optimize their costs. In the recent study of Koulinas et al. (2021), decision trees algorithms used to illustrate efficient policies depicted from a hybrid reinforcement learning algorithm while optimizing a degrading manufacturing/remanufacturing system.

The present paper extends the work of Koulinas et al. (2020), by investigating a two-stage production system and applying tree methods for rule extraction. The study aims to improve the system's performance and validate the usefulness of the suggested algorithmic technique. Given that explainability in machine learning algorithms refers to the ability to understand and interpret the output or decision made by the algorithm (Linardatos et al. 2021), we performed additional research to assess the influence of tree algorithmic factors on their efficacy, and make the model more transparent and trustworthy. Studying production systems is vital in a world where digitization and automation are quickly altering conventional industrial processes. Researchers may get significant insights into these systems' decision-making processes and suggest areas for improvement by using tree algorithms for rule extraction. The current study stresses the importance of production optimization, specifically emphasizing a two-stage production system. The study intends to enhance the system's performance, improve its efficiency, and maximize profitability by employing tree algorithms. The research examines tree algorithms' performance and characteristics to choose the most efficient strategy. This analysis offers a more thorough knowledge of the optimization process and might offer suggestions for further study. Overall, the suggested method demonstrates the value of algorithmic innovation in the manufacturing industry and offers a potential prospect for improving the performance of production systems.

The rest of the paper is organized as follows: Sect. 2 provides a literature review of the research area addressed, Sect. 3 describes the proposed framework, consisting of the input dataset and the decision trees algorithms descriptions, Sect. 4 contains the experimental analysis, and in the Sect. 5 the conclusions are illustrated.

2 Literature review

The academic literature on manufacturing systems has explored a range of approaches, with certain systems specializing in single-stage production (Rajasekharan and Peters 2000) and others including multiple stages to complete and store goods (Kim and Sarkar 2017). Some systems are even equipped to remanufacture used or returned items (Khakbaz and Tirkolaee 2022). Periodic (Rivera-Gómez et al. 2013) or aperiodic (Chen et al. 2019a, b) inspections are often conducted to monitor degradation in manufacturing systems, which can also impact product quality. Methods such as control charts (Salmasnia et al. 2022) or sampling plans (Duffuaa et al. 2009) are used to detect any such degradation in product quality.

In the context of manufacturing/remanufacturing systems, relevant research has focused on developing control policies that enhance the profitability of systems that are experiencing degradation. Assid et al. (2021) developed optimal control policies for long-term total cost minimization with a hybrid manufacturing-remanufacturing system, considering three production decisions.

In a similar way, Koulinas et al. (2021) used decision trees to define the most efficient policies, derived by a reinforcement learning algorithm, for a complex manufacturing problem of a multi-stage manufacturing/remanufacturing system. Assid et al. (2023) provided an optimal control theory-based solution of the production planning and control problem within a hybrid manufacturing-remanufacturing system. Also, Rasay et al. (2022) defined an integrated problem of optimal maintenance planning and statistical process control of two stages of a two-stage dependent manufacturing system where, due to machinery deterioration and/or equipment failure, either stage can fall into the out of control state. Also, a genetic algorithm (GA) applied to find the optimal values of the decision variables minimizing the long-run expected average cost per unit time.

In the study of Liu and Papier (2022) proposed a framework for product substitution as part of repair and refurbishment using Bayesian estimation for optimal two-way substitution between new and remanufactured products. Additionally, Zhang et al. (2021) developed a joint production-maintenance decision model to determine the amount of system component deterioration and component maintenance to reduce component idleness and shorten the time required to complete production. In the study of Dehayem Nodem et al. (2011) presented a method to find the optimal control policy for a manufacturing system subject to random machine failure and repair.

Furthermore, various studies have attempted to integrate mechanisms and techniques for the planning and scheduling of activities like production and remanufacturing, to enhance the quality of remanufactured products. Gan et al. (2022) formulated an optimal model to solve production scheduling and maintenance planning problems of a multi-component system with economic dependence. He et al. (2020) proposed a framework for remanufacturing-ontology and knowledge management, along with a reuse methodology that leverages Case Based Reasoning to reuse the most similar past solution. Liu et al. (2021) considered a novel mixed-integer linear programming model to solve process planning problems effectively using commonly available mathematical programming solvers, such as CPLEX and Gurobi. Wang et al. (2023) formalizes the remanufacturing process as a real job shop problem and studies it in terms of scheduling with simultaneous energy consumption minimization and makespan minimization. In the recent study of Paraschos et al. (2021) tried to approve frequent minimal maintenance activities by using parametric control policies and reinforcement learning techniques, to decrease the downtime of manufacturing systems. Wang and Fei (2020) suggested a novel study to deal with integrated approach in financial decision making of remanufacturing production system from manufacturing and service firms point of view. In addition, Scheller et al. (2021) developed a comprehensive modelling framework for the optimal coordination of master production and recycling scheduling in order to achieve positive contribution margins, while (Arabsheybani and Arshadi Khasmeh 2021) proposed a multi-objective mathematical model considering simultaneous optimization of resiliency and uncertainty in multi-period and multi-product supply chain.

Existing studies on optimization of manufacturing systems tend to focus largely on the use of ad hoc policies or control charts for scheduling activities and quality inspections, with little consideration for the current state of the products and systems being manufactured. These methodologies are out of sync with the principles of smart circular manufacturing that aims to eliminate waste and enhance product quality through material reuse.

In response to this challenge, our research proposes an integrated design and operation management process that incorporates continuous quality inspections and maintenance, recycling, and remanufacturing activities. By employing reinforcement learning alongside ad hoc policies, such as Kanban, we develop joint policies for production, maintenance, recycling, and remanufacturing. The objective of this unique approach is to boost the adaptability and robustness of circular manufacturing systems while at the same time maintaining cost-effectiveness.

3 The proposed framework

This research focuses on applying and performing analysis of popular and verified efficient decision tree algorithms to construct tree structures that show the best-discovered rules using a reinforcement learning algorithm. It is worth noting that the leaves of the trees represent groupings of independent variables, while the nodes represent parameters. The suggested framework is shown in Fig. 1 below.

Initially, the original dataset was generated by a reinforcement learning/Base Stock-based algorithm. Then, we performed four different decision tree algorithms to construct decision models, trying to generalize the findings from the dataset. The trees algorithms used were the REPTree, the HoeffdingTree, the J48 (C4.5), and the RandomTree algorithms. Next, we extracted the results from each algorithm and decision rule to define decision policies. The accuracy of each of the trees is monitored using standard output metrics. The result from every tree, the decision rule, is used to define control policies that could be customizable to numerous different

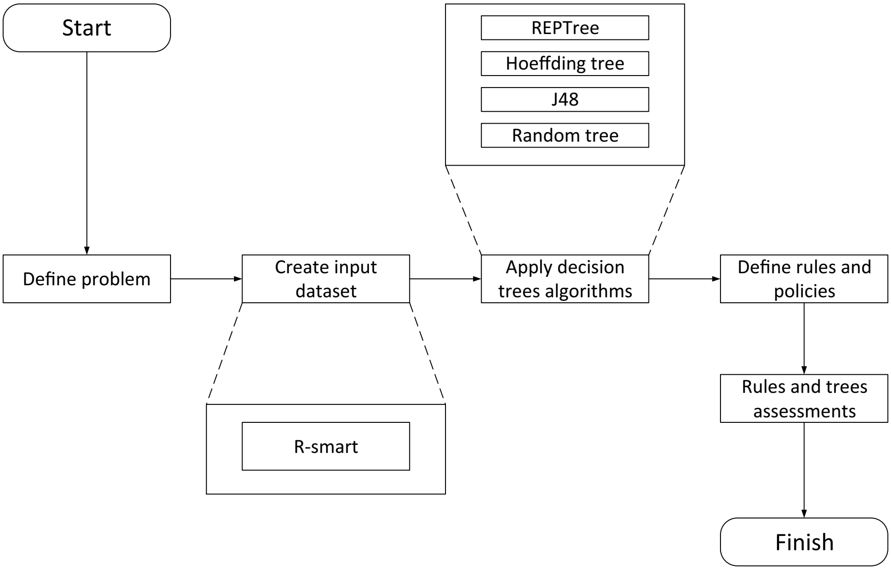

Fig. 1 The flowchart of the proposed approach

production system configurations. The efficiency of each rule regarding the total system profitability is documented by applying each proposed policy and comparing the resulting average profits. In addition, we performed a sensitivity analysis of the J48 algorithm to test if the size of the tree impacts the algorithm's accuracy and if it has an impact on the total system profitability.

The J48 (C4.5) is a classic decision tree construction method introduced by Quinlan (1992). The ID3 (Iterative Dichotomizer 3) algorithm (Quinlan 1986) underlies its functionality since it constructs a decision tree from training data collection using information entropy. The attribute with the most significant information gain, the most remarkable change in information entropy, is used as the choice criterion. The above is because the attribute with the most significant information gain can distinguish the set's instances more clearly. The tree nodes represent the set's different characteristics. At each tree node, the algorithm chooses the data characteristic that most effectively divides the collection into subgroups. The tree branches represent the various attribute values, while the final nodes represent the dependent variable's categorization. The WEKA program uses the J48 classifier to build a C4.5 pruned or unpruned tree.

The REPTree is a rapid decision tree method based on C4.5 that builds several trees and then chooses the best one to be the representative. Each decision tree is constructed using gain/variance information and pruned using reduced-error pruning with back fitting. As with C4.5, missing values are handled by fragmenting the relevant instances.

The Random Tree method generates a tree using randomly selected characteristics and does not conduct pruning. Additionally, it enables the estimation of class probabilities (Frank et al. 2016).

The Hoeffding Tree algorithm implements the Hoeffding tree technique (Hulten and Spencer 2001) and is capable of learning from large amounts of data. This method uses the fact that a limited amount of data allows for the selection of highly excellent characteristics to divide. To be more exact, the Hoeffding bound quantifies the amount of information needed to calculate fitness measurements for a characteristic. Since this method makes use of the Hoeffding constraint, it has the benefit of guaranteeing high efficiency (Hulten and Spencer 2001).

3.1 The multi-stage system description

The manufacturing system involves two stages to process one type of product. Each stage contains a manufacturing facility and a storage facility. The first stage manufacturing facility manufactures work-in-progress parts, which are later stored in the first storage facility and remain there until they are transferred into the second stage of the production process. In the second stage, the manufacturing facility completes the final goods, stockpiled in the second storage facility. The maximum capacity of the first and second inventories can be defined as Q_{WIP}^{max} and Q_{P}^{max}, respectively.

The behavior of the multi-stage system is formulated by a variety of random and dynamic events defined by a discrete event simulation technique (Xanthopoulos et al. 2016). In this respect, a series of degradation failures can affect the system's operability. Let n stages describe the system condition. With each failure, the two-stage system moves from stage n to stage $n+1$. These failures degrade the system's condition and the completed items' quality. In addition, maintenance activities are initiated to prevent significant deterioration. These activities restore the system to its previous condition before deterioration. If the system significantly deteriorates, it is considered that the system is malfunctioning and cannot continue to operate normally. Hence, the system must be repaired. The repair activities restore the system to stage 0, which is in very good condition.

Given the degradation in product quality, the system can produce high-quality, low-quality, and defective items. Regarding available sellable products, the customers can acquire only high-quality and low-quality items. On the other hand, defective items can be either remanufactured or recycled to generate revenues. The remanufacturing activities generate ready-to-be-sold products utilizing the stored faulty items. Conversely, recycling removes faulty items and creates ample space for storing high-quality and low-quality products.

Furthermore, with a probability P_r, the dissatisfied customers can return their acquired products to the system. The returned products are stockpiled in the second storage facility and can be refurbished. The refurbishing activities turn the returned items into sellable products, which are considerably cheap and of low quality. Customers acquire these items when the inventory does not contain high-quality, low-quality, and remanufactured products.

3.2 Description of the dataset

A reinforcement learning/Base Stock-based control mechanism is implemented. It assumes a similar functionality to the reinforcement learning framework introduced in the works of Paraschos et al. (2022), Xanthopoulos et al. (2017). In this regard, it exploits an interacting agent observing the manufacturing environment and searching for an optimal operation strategy that yields improved competence, reliability, and output. Analytically, the decision-making process followed by the mechanism is described as follows. Let d_e denote the timepoints at which the mechanism makes decision. At d_e, it receives information regarding the state of the studied system. The current state of the system is:

$$S^t \leftarrow \left\{f, m_{st}, \pi_\alpha, \pi_\epsilon\right\} \tag{1}$$

where f stands for system degradation, m_{st} represents the status of the multi-stage system, π_α and π_e denote the high-quality and faulty items, respectively.

Given the observed state of system, the mechanism decides upon an action. Specifically, the mechanism can initiate a maintenance activity, refurbish returned items, recycle defective items, or manufacture good under the Base Stock policy. Formally, the permitted set of action Δ is:

$$\Delta\left\{f, m_{st}, \pi_\alpha, \pi_\epsilon\right\} \leftarrow (do\ nothing, Base\ Stock),\quad f \in [0, 2] \tag{2}$$

$$\Delta\left\{f, m_{st}, \pi_\alpha, \pi_\epsilon\right\} \leftarrow (do\ nothing),\quad m_{st} = 0 \tag{3}$$

$$\Delta\left\{f, m_{st}, \pi_\alpha, \pi_\epsilon\right\} \leftarrow (do\ nothing, maintain),\quad f \geq 2\ \ and\ \ \pi_\alpha = Q_P^{max} \tag{4}$$

$$\Delta\left\{f, m_{st}, \pi_\alpha, \pi_\epsilon\right\} \leftarrow (do\ nothing, refurbish, recycle),\quad f \geq 2\ \ and\ \ \pi_\epsilon = Q_P^{max} \tag{5}$$

Following the above expressions, the mechanism behaves as follows. In case of system downtime $\left(m_{st} = 0\right)$, the mechanism remains idle as the system does not operate. When the condition of the system is relatively good ($0 \leq f < 2$), the mechanism can authorize a Base Stock-based production activity or opt to remain idle. Under the Base-Stock policy, the system starts to produce new parts when it receives a new order. In this regard, the information regarding the customer demand is transmitted to the integrated manufacturing facilities to initiate the manufacturing process. This approach facilitates the production of high quality and ready-to-be-sold products. However, in the case of considerable degradation ($f \geq 2$), the mechanism can opt for system maintenance to improve the condition of the multi-stage production system and the quality of manufactured items. In addition, the system can recycle faulty products, or refurbish returned products to make ample space for the storage of sellable products, and hence fulfill pending customer orders.

After selecting an action at d_e, a relative reward is received by the mechanism in the subsequent $d_e + 1$. The goal of the mechanism is to find the optimum policy

p^{op} that maximizes the attained average reward $\overline{A_r}$. Formally, the attained average reward is:

$$\overline{A_r} \leftarrow \lim_{l\to\infty}\frac{1}{l}E\left(\sum_{n=1}^{l}A_{d_e+1}\right) \tag{6}$$

where $A = K_{tp} + K_{lp} + K_{rec} + K_{ref} + X_r - X_{m_n} - X_p - X_{stor} - X_{los} - X_{rem} - X_{ref} - F_{ret}$.

In the above expression, K_{tp}, K_{lp}, K_{rec}, and K_{ref} refer to the high-quality, low-quality, recycled and refurbished items, X_r and X_{m_n} represent costs associated to repair and maintenance at n, X_p and X_{stor} denote the production and storage costs, X_{los} and X_{rem} refer to lost sales and remanufacturing costs, respectively. Finally, X_{ref} and F_{ret} denote the refurbishing cost and returned product fee.

For the purposes of this study, a reinforcement learning algorithm, named R-smart (Gosavi 2004), is implemented to seek out optimal control policies for the subsequent decision-tree constructions. It falls into the category of the average cost reinforcement learning algorithms, which endeavor to maximize the defined average reward at every d_e. To achieve that goal, it calculates two discrete values, that is, action value ($q_a(S, \delta)$) and average action reward ($\overline{A_{op}}$). The action values, also called q-values, are stored in a table, and are calculated at each d_e using (7). On the other hand, the average action reward is estimated with (8) at the next decision making timepoint ($d_e + 1$), when the action with the highest value is selected. Mathematically, R-smart is defined as follows:

$$q_a(S,\delta) \leftarrow q_a(S,\delta) + \beta\left[A' - \overline{A_r} + q_a\left(S',\delta'\right) - q_a(S,\delta)\right] \tag{7}$$

$$\overline{A_{op}} \leftarrow (1-\gamma)\overline{A} + \beta A \tag{8}$$

Regarding the mathematical notation in expressions (7) and (8), A' and S' refer to the reward and the system state attained by the mechanism at $d_e + 1$, δ and δ' denote the actions selected by the mechanism at d_e and $d_e + 1$, respectively, β is a real-valued contant. Finally, the action-state space is efficiently investigated using the e-greedy algorithm. The algorithm is characterized by a variable π_ε, receiving values between 0 and 1. Let us consider a probability π_r for opting an action. The algorithm can select either a random action ($\pi_r = \pi_\varepsilon$), or the one with the highest value ($\pi_r = 1 - \pi_\varepsilon$).

4 Experimental analysis

4.1 Configuration

This section describes the experimental evaluation carried out for evaluating the decision tree-based decision-making framework. First, we generated the training dataset employing the reinforcement learning/Base Stock-based control mechanism presented in Sect. 2. The dataset contained optimal control policies for optimizing manufacturing, restoring and quality control activities. The generation of decision rules was performed with the utilization of decision trees trained based on the resulted dataset. For this purpose, the Waikato Environment for Information Analysis (WEKA^{TM}) software was used for classification methods application, and construct decision trees in this research. WEKA^{TM} is an open-source predictive analysis platform comprised of a collection of machine learning algorithms and data processing techniques. Data mining activities such as data pre-processing, clustering, sorting, feature collecting, regression, and visualization may be accomplished with the WEKA^{TM} tool suite (Frank et al. 2016).

The performance of the constructed decision trees was evaluated in the context of the two-stage manufacturing system, presented in Sect. 2. To this end, 25 simulation experiments were carried out. Each experiment was replicated 15 times and lasted up to the point where the examined two-stage manufacturing system completed 5.5 million items. During the experiments, the operation of the manufacturing system was simulated under real world-like fluctuating conditions pertaining to the customer demand, production, failures, repairs, maintenance, and remanufacturing.

Table 1 presents the first 10 simulation experiments. In the table, λ_π refers to the average rate of orders arrivals, λ_m denotes the average rate of manufacturing activities, μ_{α_n} and μ_{e_n} represent the average rates of failure activities and system maintenance at stage n, μ_ρ refers to the average rate of repair activities, λ_r denotes the average rate of remanufacturing activities. It is worth noting that the mentioned average rates are exponentially distributed. Moreover, it is sensible that $\mu_{\alpha_0} > \mu_{\alpha_1} > \cdots > \mu_{\alpha_n}$, and $\mu_{e_0} < \mu_{e_1} < \cdots < \mu_{e_n}$.

4.2 Tree performance indices

Numerous output indicators may be utilized to evaluate the tree algorithms' trustworthiness. Initially, the TP (True Positive) index is calculated by counting the sum of positive instances identified correctly as positive classes. In contrast, the FP (False Positive) index is calculated by adding the number of negative classes classified incorrectly as positive. Additionally, the TPR (True Positive Rate) may be calculated by dividing the TP by the total of positive classes and the FPR (False Positive Rate) by the number of negative classes. Additionally, since the F-measure is the harmonic mean of precision and recall, it is a combined precision and accuracy indicator. Precision is calculated by dividing TP by the sum of TP and FP. In contrast, accuracy is calculated by multiplying TP by the TN (True Negative—the sum of correctly predicted negative occurrences) and dividing by the total number of cases categorized.

4.3 Results

The current study has a total of 326432 cases processed using each decision tree method. Additionally, the effect of pruning on the accuracy of the best algorithm is examined. The comparison findings for the eight instances are shown in Table 2 (4

Table 1 A summary of the experiments

	λ_π	λ_m	μ_{α_n} $0 \leq n \leq 10$	μ_ρ	μ_{e_n} $0 \leq n \leq 10$	λ_r
1	2.35	1.5	{1.95, 2.49, 3.12, 3.28, 3.82, 4.36, 4.61, 4.93, 5.24, 5.48, 6.04, 6.22}	34.36	{30.65, 29.52, 28.39, 27.26, 26.13, 25, 23.87, 22.74, 21.61, 20.48, 19.35, 18.22}	6.4
2	2.35	2.16	{5.26, 5.77, 6.28, 6.73, 6.92, 7.36, 7.48, 7.6, 7.72, 7.84, 7.96, 8.08}	34.36	{30.65, 29.52, 28.39, 27.26, 26.13, 25, 23.87, 22.74, 21.61, 20.48, 19.35, 18.22}	6.4
3	2.35	3.62	{5.26, 5.77, 6.28, 6.73, 6.92, 7.36, 7.48, 7.6, 7.72, 7.84, 7.96, 8.08}	26.27	{18.02, 17.52, 17.02, 16.52, 16.02, 15.52, 15.26, 14.11, 13.96, 12.81, 11.66, 10.51}	6.4
4	2.35	4.22	{8.13, 8.39, 8.65, 8.91, 9.13, 9.39, 9.65, 9.91, 10.17, 10.43, 10.69, 10.95}	26.27	{18.02, 17.52, 17.02, 16.52, 16.02, 15.52, 15.26, 14.11, 13.96, 12.81, 11.66, 10.51}	6.4
5	2.35	5.14	{8.13, 8.39, 8.65, 8.91, 9.13, 9.39, 9.65, 9.91, 10.17, 10.43, 10.69, 10.95}	34.36	{10.46, 9.74, 9.02, 8.3, 7.58, 6.86, 6.14, 5.42, 4.7, 3.98, 3.26, 2.54}	6.4
6	3.63	1.5	{1.95, 2.49, 3.12, 3.28, 3.82, 4.36, 4.61, 4.93, 5.24, 5.48, 6.04, 6.22}	17.83	{10.46, 9.74, 9.02, 8.3, 7.58, 6.86, 6.14, 5.42, 4.7, 3.98, 3.26, 2.54}	5.17
7	3.63	2.16	{5.26, 5.77, 6.28, 6.73, 6.92, 7.36, 7.48, 7.6, 7.72, 7.84, 7.96, 8.08}	17.83	{18.02, 17.52, 17.02, 16.52, 16.02, 15.52, 15.26, 14.11, 13.96, 12.81, 11.66, 10.51}	5.17
8	3.63	3.62	{5.26, 5.77, 6.28, 6.73, 6.92, 7.36, 7.48, 7.6, 7.72, 7.84, 7.96, 8.08}	6.24	{30.65, 29.52, 28.39, 27.26, 26.13, 25, 23.87, 22.74, 21.61, 20.48, 19.35, 18.22}	5.17
9	3.63	4.22	{8.13, 8.39, 8.65, 8.91, 9.13, 9.39, 9.65, 9.91, 10.17, 10.43, 10.69, 10.95}	6.24	{18.02, 17.52, 17.02, 16.52, 16.02, 15.52, 15.26, 14.11, 13.96, 12.81, 11.66, 10.51}	5.17
10	3.63	5.14	{8.13, 8.39, 8.65, 8.91, 9.13, 9.39, 9.65, 9.91, 10.17, 10.43, 10.69, 10.95}	17.83	{30.65, 29.52, 28.39, 27.26, 26.13, 25, 23.87, 22.74, 21.61, 20.48, 19.35, 18.22}	5.17

Table 2 Comparative results for the decision trees algorithms

	J48 (0.35)	J48 (0.25)	J48 (0.2)	J48 (0.1)	J48 (0.05)	HoeffdingTree	REPTree	RandomTree
Size of tree	77	75	67	63	53		37	349
Number of leaves	39	38	34	32	27			
Correctly Classified Instances	326,321	326,324	326,325	3,263,216	326,314	326,216	326,311	326,305
Incorrectly Classified Instances	111	108	107	116	118	216	121	127
Relative absolute error	0.0715%	0.0705%	0.072%	0.076%	0.0797%	0.2355%	0.0833%	0.0655%
Total Number of Instances	326,432	326,432	326,432	326,432	326,432	326,432	326,432	326,432

algorithms). Notably, the C4.5 (J48) was applied with confidence factors of 0.35, 0.25, 0.2, 0.1, and 0.05, while lower values result in greater pruning. Initially, the correctly categorized cases and those erroneously classified are shown, with the J48 (0.2) being the most effective, accurately classifying 326,325 occurrences. The pruned J48 (0.25) method was more accurate than the HoeffdingTree, RERTree, and RandomTree algorithms, which wrongly categorized 216, 121, and 127 cases, respectively.

As for the processing time needed for each algorithm to build the model, it took 5.35 s for the J48 (0.35), 3.73 s for the J48 (0.35), 3.81 s for the J48 (0.2), 4.44 s for the J48 (0.1), and 4.32 s for the J48 (0.05). As for the rest of the algorithms, the HoeffdingTree needed 4.83 s to build the model, while the REPTree consumed 3.04 s and the RandomTree about 2.67 s. As expected, a larger size of trees (namely less pruned) need more processing time, but generally, the time needed for model construction was reasonable.

Regarding the effect of pruning on the J48 algorithm's accuracy, it was noted that having more trimmed trees, i.e., trees with a smaller diameter, does not ensure that the algorithm would be more accurate. More precisely, the J48 (0.2) tree is erroneously categorized in just 107 occurrences, while the more pruned J48 (0.05) tree is incorrectly classified in 118 occasions, and the less pruned J48 (0.35) tree is incorrectly classified in 111 instances.

Additionally, Table 3 includes information on the performance of each tree method in terms of TP rate, FP rate, and F-measure. More precisely, the J48 method obtains the highest TP rate for all classes except "refurbish," for which the RandomTree approach beats the other techniques. Regarding the J48 method variations, it should be noted that less pruned trees are more likely to result in higher TP rate values. There are many equalities, presumably due to the high number of instances and the decimal precision chosen. Regarding the FP rate, the HoeffdingTree algorithm has the highest value (for the class "recycle"). In contrast, the other values are equivalent to zero due to the minimal number of incorrectly classified instances compared to the total number of examples used. Regarding the F-Measure index, the J48 algorithm outperforms all other algorithms for all classes except "refurbish," for which the REPTree algorithm outperforms all others, with the J48 coming second and the RandomTree and HoeffdingTree following.

In this research, each rule extracted by every decision tree algorithm was employed to address the optimization problem that the R-smart algorithm was initially confronted with, validating its efficacy in managing the production system and optimizing its profitability. Each rule was applied to the optimization problem, resulting in the profitability shown in Table 4. It is worth noting that the average profit values for the processed cases are shown. These results demonstrate that the J48 algorithm outperforms the other methods regarding average profit.

In addition, regarding the J48 algorithm, it is found that less pruning assists in producing a better-performing decision rule since the J48 (0.35) rule achieved better average profitability instead of the J48 (0.05), J48 (0.1), J48 (0.2), and the J48 (0.25). Moreover, it is worth noting that the J48 (0.35) algorithm achieved the best profitability for 10 of the 25 scenarios, the J48 (0.25) for 3 of 25, the J48 (0.2) for 5 of 25, the J48 (0.1) for 6 of 25, and the J48 (0.05) for 6 of 25. As for the rest algorithms,

Table 3 Results for the algorithms of the present analysis

Class	TP Rate	FP Rate	F-Measure	Algorithm
Do nothing	1	0	1	J48 (0.35)
	1	0	1	J48 (0.25)
	1	0	1	J48 (0.2)
	1	0	1	J48 (0.1)
	1	0	1	J48 (0.05)
	0.999	0	0.999	HoeffdingTree
	1	0	1	REPTree
	1	0	1	RandomTree
Produce	1	0	1	J48 (0.35)
	1	0	1	J48 (0.25)
	1	0	1	J48 (0.2)
	1	0	1	J48 (0.1)
	1	0	1	J48 (0.05)
	0.999	0	0.999	HoeffdingTree
	1	0	1	REPTree
	1	0	1	RandomTree
Refubrish	0.966	0	0.975	J48 (0.35)
	0.966	0	0.978	J48 (0.25)
	0.966	0	0.979	J48 (0.2)
	0.963	0	0.977	J48 (0.1)
	0.963	0	0.979	J48 (0.05)
	0.905	0	0.949	HoeffdingTree
	0.966	0	0.983	REPTree
	0.968	0	0.963	RandomTree
Maintain	0.994	0	0.991	J48 (0.35)
	0.994	0	0.991	J48 (0.25)
	0.994	0	0.991	J48 (0.2)
	0.994	0	0.991	J48 (0.1)
	0.995	0	0.990	J48 (0.05)
	0.974	0	0.987	HoeffdingTree
	0.994	0	0.988	REPTree
	0.989	0	0.989	RandomTree
Recycle	1	0	1	J48 (0.35)
	1	0	1	J48 (0.25)
	1	0	1	J48 (0.2)
	1	0	1	J48 (0.1)
	1	0	1	J48 (0.05)
	1	0.001	1	HoeffdingTree
	1	0	1	REPTree
	1	0	1	RandomTree

Table 3 (continued)

Class	TP Rate	FP Rate	F-Measure	Algorithm
Weighted Avg	1	0	1	J48 (0.35)
	1	0	1	J48 (0.25)
	1	0	1	J48 (0.2)
	1	0	1	J48 (0.1)
	1	0	1	J48 (0.05)
	0.999	0	0.999	HoeffdingTree
	1	0	1	REPTree
	1	0	1	RandomTree

the HoeffdingTree did not find the highest profits for any of the scenarios, the REPTree found 4 of 25, and the RandomTree found 3 of 25. Note that when applying the R-learning algorithm (without any decision tree constructed and no decision rule), the average profitability is 4.10E+08 k€. The best profitability is found for 10 of the 25 scenarios. This finding illustrates the value of the current innovative approach since the rule extracted by the data generated with R-smart achieves better results than the R-smart algorithm alone.

The rule derived using the most efficient method (in terms of profitability), the J48 (0.35), is described in Table 5. This rule states that if the system's "machine" level is equal to or less than zero, the action executed is "do nothing." Additionally, suppose the "machine" parameter is positive, and the "prod a" value is equal to or less than 99. In that case, the "def prod" value is equal to or less than 99, the "def prod" value is equal to or less than 6, the "prod a" value is equal to or less than 9, and the "machine" parameter is equal to or less than 1, the action performed is "produce". Similarly, the rule contains "instructions" for carrying out actions in each instance, which may result in a higher objective function value for the issue.

5 Conclusions

In this study, we used a variety of decision tree algorithms to classify instances and represent the highest performance policies generated by a reinforcement learning algorithm while optimizing a complex production, maintenance, and quality control problem within a degrading manufacturing and remanufacturing system. More specifically, we applied decision tree algorithms for the first time to mine knowledge that can optimize production, quality, and maintenance in the context of flexible manufacturing systems. The results we achieved were highly encouraging. Additionally, we conducted further research to exposit the influence of algorithmic factors on the decision tree's efficacy, ensuring that decision-makers could understand the procedures involved and trust the model utilized.

Table 4 Results of profitability for the trees algorithms used (10^3 €)—(R-smart)

Scenario	REPTREE	Hoeffding Tree	J48 (0.05)	J48 (0.1)	J48 (0.2)	J48 (0.25)	J48 (0.35)	Random Tree	R-smart
1	2.47E+08	5.22E+08	2.47E+08	5.22E+08	5.22E+08	2.48E+08	5.23E+08	2.47E+08	2.51E+08
2	5.22E+08	2.47E+08	2.48E+08	2.47E+08	2.48E+08	2.48E+08	5.23E+08	2.48E+08	7.35E+07
3	2.48E+08	2.48E+08	5.22E+08	5.22E+08	5.22E+08	2.48E+08	5.22E+08	5.22E+08	4.55E+07
4	2.48E+08	2.48E+08	5.22E+08	2.47E+08	5.22E+08	2.47E+08	5.22E+08	2.47E+08	8.71E+08
5	2.48E+08	2.47E+08	5.22E+08	5.22E+08	2.48E+08	2.47E+08	5.22E+08	2.47E+08	2.71E+08
6	2.47E+08	2.47E+08	5.23E+08	2.47E+08	5.22E+08	2.48E+08	5.23E+08	2.48E+08	2.93E+08
7	2.47E+08	5.22E+08	2.48E+08	5.23E+08	2.48E+08	2.47E+08	5.23E+08	2.47E+08	1.82E+08
8	2.48E+08	2.48E+08	2.48E+08	5.22E+08	2.48E+08	5.23E+08	2.47E+08	2.48E+08	7.62E+08
9	5.23E+08	2.48E+08	5.22E+08	5.22E+08	5.22E+08	5.23E+08	5.22E+08	2.47E+08	6.85E+08
10	2.48E+08	2.48E+08	2.48E+08	2.47E+08	5.22E+08	5.23E+08	5.22E+08	2.47E+08	1.23E+08
11	2.47E+08	2.48E+08	2.47E+08	2.48E+08	5.23E+08	2.47E+08	2.48E+08	5.23E+08	5.37E+08
12	5.23E+08	2.47E+08	2.48E+08	2.47E+08	2.48E+08	5.23E+08	2.48E+08	2.48E+08	4.37E+08
13	2.47E+08	2.48E+08	5.23E+08	5.22E+08	2.47E+08	2.48E+08	5.22E+08	5.22E+08	6.79E+07
14	5.22E+08	5.22E+08	5.23E+08	5.22E+08	5.22E+08	5.23E+08	2.48E+08	5.22E+08	3.99E+07
15	2.47E+08	2.47E+08	5.22E+08	5.22E+08	5.22E+08	2.48E+08	2.47E+08	2.48E+08	5.02E+08
16	2.48E+08	2.48E+08	2.47E+08	2.47E+08	2.47E+08	2.47E+08	5.22E+08	2.48E+08	1.25E+08
17	2.48E+08	2.47E+08	2.47E+08	2.47E+08	5.23E+08	5.23E+08	5.22E+08	5.23E+08	7.56E+08
18	2.48E+08	2.48E+08	5.22E+08	5.22E+08	5.22E+08	5.23E+08	2.48E+08	5.22E+08	8.88E+08
19	5.22E+08	2.47E+08	2.47E+08	5.22E+08	2.48E+08	2.48E+08	5.22E+08	2.48E+08	2.75E+08
20	5.22E+08	2.47E+08	2.48E+08	5.22E+08	2.48E+08	5.23E+08	5.23E+08	5.22E+08	6.22E+08
21	2.48E+08	2.47E+08	2.47E+08	5.22E+08	5.23E+08	2.48E+08	5.23E+08	2.47E+08	8.40E+08
22	2.48E+08	2.47E+08	2.48E+08	2.48E+08	2.48E+08	2.48E+08	5.23E+08	2.48E+08	4.73E+08
23	5.22E+08	2.48E+08	2.47E+08	5.22E+08	5.22E+08	2.47E+08	5.22E+08	5.22E+08	4.40E+08
24	2.47E+08	2.48E+08	5.22E+08	2.48E+08	5.22E+08	2.47E+08	2.48E+08	5.22E+08	3.41E+08

Table 4 (continued)

Scenario	REPTREE	Hoeffding Tree	J48 (0.05)	J48 (0.1)	J48 (0.2)	J48 (0.25)	J48 (0.35)	Random Tree	R-smart
25	5.22E+08	2.48E+08	5.22E+08	5.22E+08	5.22E+08	2.48E+08	5.22E+08	2.48E+08	3.58E+08
Avg. Profit	3.35E+08	2.81E+08	3.69E+08	4.12E+08	4.13E+08	3.36E+08	4.46E+08	3.47E+08	4.10E+08

Table 5 The rule extracted by the J48 (0.35) algorithm

```
J48 pruned tree
------------------
 machine <= 0: do nothing (81608.0)
 machine > 0
|   deterioration <= 1
|   |   prod_a <= 99
|   |   |   def_prod <= 99
|   |   |   |   def_prod <= 6
|   |   |   |   |   prod_a <= 9
|   |   |   |   |   |   machine <= 1: produce (140.0)
|   |   |   |   |   |   machine > 1
|   |   |   |   |   |   |   machine <= 2
|   |   |   |   |   |   |   |   def_prod <= 2
|   |   |   |   |   |   |   |   |   deterioration <= 0: produce (30.0/13.0)
|   |   |   |   |   |   |   |   |   deterioration > 0
|   |   |   |   |   |   |   |   |   |   prod_a <= 5: do nothing (18.0/1.0)
|   |   |   |   |   |   |   |   |   |   prod_a > 5
|   |   |   |   |   |   |   |   |   |   |   def_prod <= 1: do nothing (8.0/1.0)
|   |   |   |   |   |   |   |   |   |   |   def_prod > 1: produce (4.0)
|   |   |   |   |   |   |   |   def_prod > 2
|   |   |   |   |   |   |   |   |   prod_a <= 2
|   |   |   |   |   |   |   |   |   |   def_prod <= 4
|   |   |   |   |   |   |   |   |   |   |   prod_a <= 1
|   |   |   |   |   |   |   |   |   |   |   |   deterioration <= 0: produce (4.0/1.0)
|   |   |   |   |   |   |   |   |   |   |   |   deterioration > 0: do nothing (4.0/1.0)
|   |   |   |   |   |   |   |   |   |   |   prod_a > 1: do nothing (4.0)
|   |   |   |   |   |   |   |   |   |   def_prod > 4
|   |   |   |   |   |   |   |   |   |   |   prod_a <= 0: do nothing (4.0/1.0)
|   |   |   |   |   |   |   |   |   |   |   prod_a > 0: produce (8.0)
|   |   |   |   |   |   |   |   |   prod_a > 2: produce (56.0/6.0)
|   |   |   |   |   |   |   machine > 2: produce (140.0)
|   |   |   |   |   prod_a > 9: produce (3780.0)
|   |   |   |   def_prod > 6: produce (55800.0)
|   |   |   def_prod > 99: refubrish (600.0)
|   |   prod_a > 99: maintain (606.0)
|   deterioration > 1
|   |   prod_a <= 99
|   |   |   def_prod <= 4
|   |   |   |   prod_a <= 8
|   |   |   |   |   machine <= 1: recycle (270.0)
|   |   |   |   |   machine > 1
|   |   |   |   |   |   machine <= 2
|   |   |   |   |   |   |   deterioration <= 5
|   |   |   |   |   |   |   |   def_prod <= 1
|   |   |   |   |   |   |   |   |   prod_a <= 3: maintain (32.0/8.0)
|   |   |   |   |   |   |   |   |   prod_a > 3
|   |   |   |   |   |   |   |   |   |   deterioration <= 4
|   |   |   |   |   |   |   |   |   |   |   def_prod <= 0: maintain (15.0/6.0)
|   |   |   |   |   |   |   |   |   |   |   def_prod > 0
|   |   |   |   |   |   |   |   |   |   |   |   prod_a <= 5: maintain (6.0/2.0)
|   |   |   |   |   |   |   |   |   |   |   |   prod_a > 5
|   |   |   |   |   |   |   |   |   |   |   |   |   deterioration <= 3: refubrish
(6.0/3.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   deterioration > 3: recycle (3.0/1.0)
|   |   |   |   |   |   |   |   |   |   deterioration > 4: recycle (10.0/1.0)
|   |   |   |   |   |   |   |   def_prod > 1
|   |   |   |   |   |   |   |   |   prod_a <= 2
|   |   |   |   |   |   |   |   |   |   prod_a <= 0: maintain (12.0/2.0)
|   |   |   |   |   |   |   |   |   |   prod_a > 0
|   |   |   |   |   |   |   |   |   |   |   def_prod <= 3
|   |   |   |   |   |   |   |   |   |   |   |   def_prod <= 2: refubrish (8.0/4.0)
|   |   |   |   |   |   |   |   |   |   |   |   def_prod > 2
|   |   |   |   |   |   |   |   |   |   |   |   |   deterioration <= 2: maintain (2.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   deterioration > 2: recycle (6.0/2.0)
|   |   |   |   |   |   |   |   |   |   |   def_prod > 3: recycle (8.0/3.0)
|   |   |   |   |   |   |   |   |   prod_a > 2: recycle (72.0/7.0)
|   |   |   |   |   |   |   deterioration > 5
|   |   |   |   |   |   |   |   prod_a <= 0
|   |   |   |   |   |   |   |   |   def_prod <= 2: produce (6.0/2.0)
|   |   |   |   |   |   |   |   |   def_prod > 2: recycle (4.0)
|   |   |   |   |   |   |   |   prod_a > 0: recycle (80.0/2.0)
|   |   |   |   |   |   machine > 2: recycle (270.0)
|   |   |   |   prod_a > 8: recycle (8190.0/1.0)
|   |   |   def_prod > 4: recycle (172800.0/2.0)
|   |   prod_a > 99
|   |   |   def_prod <= 98: maintain (1782.0)
|   |   |   def_prod > 98
|   |   |   |   def_prod <= 99: maintain (18.0)
|   |   |   |   def_prod > 99: recycle (18.0)
```

Among the performed algorithms, the C4.5 (J48) was found to be the most accurate, especially its highly pruned version (J48 (0.01)), with the Hoeffding Tree, Random Tree, and REPTree, following. Additionally, the extracted rules were employed to address the initial optimization problem that the reinforcement learning algorithm addressed to demonstrate their effectiveness at controlling the production system and maximizing its profitability. Again, the J48 algorithm proved the most efficient compared to the other three decision tree algorithms. As for the J48 algorithm, we performed a sensitivity analysis to determine if the pruning level impacts the objective function value. Regarding profitability, we discovered that utilizing the J48 algorithm to build more extensive and complex trees can generate rules that result in higher profits. However, it is worth noting that this approach may compromise accuracy. Note that this finding is only for the J48 algorithm and cannot be generalized since the more giant trees created with the RandomTree algorithm have lower profitability.

Additionally, we retrieved the decision rule from each associated tree and put it in the optimization issue to validate its effectiveness. The verification findings established that the J48 method is the most efficient. At the same time, the pruning level affects the rule's efficiency, as extremely large or small tree sizes result in poor rule performance regarding system profitability.

In terms of management implications, the comparison of findings has a significant impact on real-world industries and the ever-evolving challenges they face. By thoroughly analyzing and categorizing the control policies generated by the production mechanism, we have developed a framework that produces effective decision rules for the manufacturing and quality control of items. Owing to the universal nature of these principles, they can be implemented in new manufacturing challenges and systems without incurring significant costs. This article reveals how our framework can be applied to real-world situations, thereby enabling organizations to improve their manufacturing processes and yield better quality items.

The retrieved information has significant implications for decision-making processes in complex and dynamic real-world industrial environments. Specifically, managers can leverage the suggested technology to analyze vast quantities of data collected from production systems and identify broad patterns, requirements, and trends dictated by customers and the manufacturing sector. This information can be utilized to adapt the integrated production process and supply chains while minimizing emergent operational expenses like missed sales. The proposed method is an intelligent and robust tool for decision-making in the Industry 4.0 era, which has been transformed by technologies such as the Internet of Things (IoT). Future work could include the application of additional machine learning principles such as neural networks to compare algorithmic accuracy and system profitability. Additionally, more complex production systems could be explored.

Author contributions GKK: Conceptualization, Methodology, Software, Validation, Investigation, Formal Analysis, Data Curation, Visualization, Writing—Original Draft, Writing—Review & Editing. PDP: Conceptualization, Methodology, Validation, Investigation, Data Curation, Visualization, Writing—Review & Editing. DEK: Conceptualization, Methodology, Validation, Investigation, Writing—Review

& Editing, Supervision, Project Administration, Funding acquisition. All authors have read and agreed to the published version of the manuscript.

Funding Open access funding provided by HEAL-Link Greece. The authors have no relevant financial or non-financial interests to disclose.

Data availability The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Conflict of interest No potential competing interest is reported by the authors.

References

Agarwal P, Melih Tamer M, Sahraei H, Budman H (2020) Deep learning for classification of profit-based operating regions in industrial processes. Ind Eng Chem Res 59(6):2378–2395. https://doi.org/10.1021/acs.iecr.9b04737

Ahuett-Garza H, Kurfess T (2018) A brief discussion on the trends of habilitating technologies for industry 4.0 and smart manufacturing. Manuf Lett 15:60–63. https://doi.org/10.1016/j.mfglet.2018.02.011

Antosz K, Paśko Ł, Gola A (2020) The use of artificial intelligence methods to assess the effectiveness of lean maintenance concept implementation in manufacturing enterprises. Appl Sci 10(21):1–24. https://doi.org/10.3390/app10217922

Arabsheybani A, Khasmeh AA (2021) 19 Robust and resilient supply chain network design considering risks in food industry: flavour industry in Iran. Int J Manag Sci Eng Manag 16(3):197–208. https://doi.org/10.1080/17509653.2021.1907811

Assid M, Gharbi A, Hajji A (2021) Production planning and control of unreliable hybrid manufacturing-remanufacturing systems with quality-based categorization of returns. J Clean Prod 312:127800. https://doi.org/10.1016/j.jclepro.2021.127800

Assid M, Gharbi A, Hajji A (2023) 10 control policies of changeable manufacturing-remanufacturing systems using two failure-prone production facilities. Int J Adv Manuf Technol 125(1):279–297. https://doi.org/10.1007/s00170-022-10664-8

Chen C, Liu Y, Kumar M, Qin J, Ren Y (2019a) Energy consumption modelling using deep learning embedded semi-supervised learning. Comput Ind Eng 135:757–765. https://doi.org/10.1016/j.cie.2019.06.052

Chen Z, He Y, Zhao Y, Han X, Liu F, Zhou Di, Wang W (2019b) 5 Mission reliability-oriented selective maintenance optimization for intelligent multistate manufacturing systems with uncertain maintenance quality. IEEE Access 7:109804–109816. https://doi.org/10.1109/ACCESS.2019.2933580

Dehayem Nodem FI, Kenné JP, Gharbi A (2011) 13B simultaneous control of production, repair/replacement and preventive maintenance of deteriorating manufacturing systems. Int J Prod Econ 134(1):271–282. https://doi.org/10.1016/j.ijpe.2011.07.011

Dogan A, Birant D (2021) Machine learning and data mining in manufacturing. Expert Syst Appl. https://doi.org/10.1016/j.eswa.2020.114060

Dohale V, Verma P, Gunasekaran A, Akarte M (2023) Manufacturing strategy 4.0: a framework to usher towards industry 4.0 implementation for digital transformation. Ind Manag Data Syst 123(1):10–40. https://doi.org/10.1108/IMDS-12-2021-0790

Duffuaa SO, Al-Turki UM, Kolus AA (2009) 7 Process-targeting model for a product with two dependent quality characteristics using acceptance sampling plans. Int J Prod Res 47(14):4031–4046. https://doi.org/10.1080/00207540701644243

Frank E, Hall M, Witten I (2016) The WEKA Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques." Fourth Edi. Morgan Kaufmann

Galletta A, Carnevale L, Celesti A, Fazio M, Villari M (2018) A cloud-based system for improving retention marketing loyalty programs in industry 4.0: a study on big data storage implications. IEEE Access 6:5485–5492. https://doi.org/10.1109/ACCESS.2017.2776400

Gan J, Zhang W, Wang S, Zhang X (2022) 14 Joint decision of condition-based opportunistic maintenance and scheduling for multi-component production systems. Int J Prod Res 60(17):5155–5175. https://doi.org/10.1080/00207543.2021.1951447

Gosavi A (2004) Reinforcement learning for long-run average cost. Eur J Oper Res 155(3):654–674. https://doi.org/10.1016/S0377-2217(02)00874-3

He Y, Hao C, Wang Y, Li Y, Wang Y, Huang L, Tian X (2020) 14A an ontology-based method of knowledge modelling for remanufacturing process planning. J Clean Prod 258:120952. https://doi.org/10.1016/j.jclepro.2020.120952

Hofmann E, Rüsch M (2017) Industry 4.0 and the current status as well as future prospects on logistics. Comput Ind 89:23–34. https://doi.org/10.1016/j.compind.2017.04.002

Hulten G, Spencer L (2001) Mining time-changing data streams. In: KDD '01: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, pp 97–106

Kang Z, Catal C, Tekinerdogan B (2020) Machine learning applications in production lines: a systematic literature review. Comput Ind Eng 149:106773. https://doi.org/10.1016/j.cie.2020.106773

Khakbaz A, Tirkolaee EB (2022) 3 A sustainable hybrid manufacturing/remanufacturing system with two-way substitution and WEEE directive under different market conditions. Optimization 71(11):3083–3106. https://doi.org/10.1080/02331934.2021.1935937

Kim M-S, Sarkar B (2017) 2 multi-stage cleaner production process with quality improvement and lead time dependent ordering cost. J Clean Prod 144:572–590. https://doi.org/10.1016/j.jclepro.2016.11.052

Koulinas G, Paraschos P, Koulouriotis D (2020) A decision trees-based knowledge mining approach for controlling a complex production system. Procedia Manuf 51:1439–1445. https://doi.org/10.1016/j.promfg.2020.10.200

Koulinas G, Paraschos P, Koulouriotis D (2021) A machine learning-based framework for data mining and optimization of a production system. Procedia Manuf 55:431–438. https://doi.org/10.1016/j.promfg.2021.10.059

Linardatos P, Papastefanopoulos V, Kotsiantis S (2021) Explainable AI: a review of machine learning interpretability methods. Entropy 23(1)

Liu B, Papier F (2022) 12 Remanufacturing of multi-component systems with product substitution. Eur J Oper Res 301(3):896–911. https://doi.org/10.1016/j.ejor.2021.11.029

Liu Q, Li X, Gao L (2021) 14B A Novel MILP model based on the topology of a network graph for process planning in an intelligent manufacturing system. Engineering 7(6):807–817. https://doi.org/10.1016/j.eng.2021.04.011

Lyu J, Liang CW, Chen PS (2020) A data-driven approach for identifying possible manufacturing processes and production parameters that cause product defects: a thin-film filter company case study. IEEE Access 8:49395–49411. https://doi.org/10.1109/ACCESS.2020.2974535

Mahato PK, Narayan A (2020) Robust supply chains with gradient boosted trees. In: 2020 IEEE Symposium Series on Computational Intelligence, SSCI 2020. The University of British Columbia, Department of Computer ScienceBC, Canada, pp 2633–39

O'Donovan P, Leahy K, Bruton K, O'Sullivan DTJ (2015) An industrial big data pipeline for data-driven analytics maintenance applications in large-scale smart manufacturing facilities. J Big Data 2(1):25. https://doi.org/10.1186/s40537-015-0034-z

Palsodkar M, Yadav G, Nagare MR (2023) Integrating industry 4.0 and agile new product development practices to evaluate the penetration of sustainable development goals in manufacturing industries. J Eng Design Technol. https://doi.org/10.1108/JEDT-02-2022-0101

Paraschos PD, Koulinas GK, Koulouriotis DE (2021) 16 parametric and reinforcement learning control for degrading multi-stage systems. Procedia Manuf 55:401–408. https://doi.org/10.1016/j.promfg.2021.10.055

Paraschos PD, Xanthopoulos AS, Koulinas GK, Koulouriotis DE (2022) Machine learning integrated design and operation management for resilient circular manufacturing systems. Comput Ind Eng 167:107971. https://doi.org/10.1016/j.cie.2022.107971

Quinlan JR (1986) Induction of decision trees. Mach Learn 1(1):81–106. https://doi.org/10.1007/BF00116251

Quinlan JR (1992) C4.5 programs for machine learning. Morgan Kaufmann, San Mateo

Rajasekharan M, Peters BA (2000) 1 strategic configuration of flexible electronics assembly facilities facing stochastic requirements. Int J Prod Res 38(3):639–656. https://doi.org/10.1080/002075400189347

Rasay H, Naderkhani F, Azizi F (2022) 11 Opportunistic maintenance integrated model for a two-stage manufacturing process. Int J Adv Manuf Technol 119(11):8173–8191. https://doi.org/10.1007/s00170-021-08571-5

Rivera-Gómez H, Gharbi A, Kenné JP (2013) 4 Joint production and major maintenance planning policy of a manufacturing system with deteriorating quality. Int J Prod Econ 146(2):575–587. https://doi.org/10.1016/j.ijpe.2013.08.006

Salmasnia A, Hajihosseini Z, Maleki MR (2022) 6 an economic manufacturing quantity model with rework process for deteriorating products under maintenance-quality policy. Int J Model Simul 42(6):946–965. https://doi.org/10.1080/02286203.2021.2004341

Scheller C, Schmidt K, Spengler TS (2021) 17B decentralized master production and recycling scheduling of lithium-ion batteries: a techno-economic optimization model. J Bus Econ 91(2):253–282. https://doi.org/10.1007/s11573-020-00999-7

Soeprapto Putri NK, Steven Puika K, Ibrahim S, Darmawan L (2018) Defect classification using decision tree. In: Proceedings of 2018 International Conference on Information Management and Technology, ICIMTech 2018. Information Systems Department, Bina Nusantara University, School of Information Systems, Jakarta, 11480, Indonesia, pp 281–85

Tan Q, Tong Y, Wu S, Li D (2019) Modeling, planning, and scheduling of shop-floor assembly process with dynamic cyber-physical interactions: a case study for cps-based smart industrial robot production. Int J Adv Manuf Technol 105(9):3979–3989. https://doi.org/10.1007/s00170-019-03940-7

Wang W, Tian G, Yuan G, Pham DT (2023) 14C energy-time tradeoffs for remanufacturing system scheduling using an invasive weed optimization algorithm. J Intell Manuf 34(3):1065–1083. https://doi.org/10.1007/s10845-021-01837-5

Wang Y, Fei W (2020) 17A production decisions of manufacturing and remanufacturing hybrid system considering downward substitution: a comprehensive model integrating financial operations. IEEE Access 8:124869–124882. https://doi.org/10.1109/ACCESS.2020.3006858

Wu X, Vipin Kumar J, Quinlan R, Ghosh J, Yang Q, Motoda H, Mclachlan GJ, Ng A, Liu B, Yu PS, Zhou Z-H, Steinbach M, Hand DJ, Steinberg D (2008) Top 10 algorithms in data mining. Knowl Inf Syst 14(1):1–37. https://doi.org/10.1007/s10115-007-0114-2

Xanthopoulos AS, Koulouriotis DE, Gasteratos A, Ioannidis S (2016) Efficient priority rules for dynamic sequencing with sequence-dependent setups. Int J Ind Eng Comput 7(3):367–384. https://doi.org/10.5267/j.ijiec.2016.2.002

Xanthopoulos ASS, Kiatipis A, Koulouriotis DEE, Stieger S (2017) Reinforcement learning-based and parametric production-maintenance control policies for a deteriorating manufacturing system. IEEE Access 6:576–588. https://doi.org/10.1109/ACCESS.2017.2771827

Zangaro F, Minner S, Battini D (2020) A supervised machine learning approach for the optimisation of the assembly line feeding mode selection. Int J Prod Res. https://doi.org/10.1080/00207543.2020.1851793

Zhang W, Gan J, Hou Q (2021) 13A Joint decision of condition-based maintenance and production scheduling for multi-component systems. Proc Inst Mech Eng Part b J Eng Manuf 236(6–7):726–740. https://doi.org/10.1177/09544054211043759

Zhong RY, Xun Xu, Klotz E, Newman ST (2017) Intelligent manufacturing in the context of industry 4.0: a review. Engineering 3(5):616–630. https://doi.org/10.1016/J.ENG.2017.05.015

Georgios K. Koulinas received the Diploma and PhD degrees in Production and Management Engineering from the Democritus University of Thrace, Greece, and the M.sc. degree in Engineering Project Management from the Hellenic Open University, Greece. He is currently an Assistant Professor with the Department of Production and Management Engineering, Democritus University of Thrace, Greece and an Adjunct Lecturer with the Hellenic Open University, Greece. He has authored several scientific articles. His research interests include hyper-heuristic algorithms, metaheuristics, fuzzy systems, machine learning, artificial intelligence algorithms, risk assessment, project scheduling, project management, industrial engineering and management, and safety systems.

Panagiotis D. Paraschos received the Diploma and Ph.D. degrees in production & management engineering from the Democritus University of Thrace (DUTh), Greece, in 2018 and 2023, respectively. His doctoral thesis is about joint production, maintenance and quality control optimization in cyberphysical manufacturing systems by means of machine learning and deep learning. His research interests involve process planning and control, quality control, smart manufacturing, Industry 4.0, cyberphysical systems, machine learning-based optimization, manufacturing personalization and sustainability.

Dimitrios E. Koulouriotis received the diploma (combined B. Sc. And M.Sc.) degree in Electrical and Computer Engineering from the Democritus University of Thrace, Greece, the M.Sc. degree in Electronic and Computer Engineering and the Ph.D. in Production and Management Engineering from the Technical University of Crete, Greece. Currently, he is a Full Professor in the School of Mechanical Engineering, at National Technical University of Athens. He served as a Professor (2003-2023) and Chairman in the Department of Production and Management Engineering (2020-2023) and Director of the Industrial Production Laboratory, Logistics Laboratory, and Ergonomics and Safety Laboratory at Democritus University of Thrace, Greece (2015-2023). He is the author of three books and numerous articles. His research interests include intelligent systems, machine learning, knowledge management, industrial engineering and management, machine vision and signal processing, system safety and business intelligence.