Università Politecnica delle Marche
Scuola di Dottorato di Ricerca in Scienze dell'Ingegneria
Corso di Dottorato in Ingegneria Industriale

Data-driven approaches to maintenance policy definition: general framework and applications

Ph.D. Dissertation of:
Sara Antomarioni

Supervisors:
Prof. Maurizio Bevilacqua

Prof. Filippo Emanuele Ciarapica

Ph.D. Course coordinator:
Prof. Giovanni Di Nicola

XIX edition - new series

Università Politecnica delle Marche
Dipartimento di Ingegneria Industriale e Scienze Matematiche
Via Brecce Bianche — 60131 - Ancona, Italy

Acknowledgments

I wish to express my recognition to Professor Maurizio Bevilacqua and Professor Filippo Ciarapica that mentored me during the development of this research by giving me precious advice and actively cooperating. I also wish to thank them and Univerità Politecnica delle Marche for financing my Doctoral project.

I wish to thank Professor Gülgün Alpan and Professor Jan Ola Strandhagen for accepting to review this thesis, as well as to the whole examination board .

I wish to acknowledge Universidad de Sevilla and, specifically, Professor Adolfo Crespo Marquez and his research group for hosting me for four months supporting my research during my stay in Spain.

I am also grateful to my colleagues for helping me improve as an aspiring researcher.

The most profound acknowledgment goes to Simone, my big family, and my lifelong friends for their constant support and extreme patience during the whole Ph.D. journey.

Contents

List of Figures

List of Tables

Abstract

High levels of process reliability are required to comply with the increased competitiveness characterizing the current industrial scenario. This aspect is particularly relevant for complex plants since many components are potentially more subject to failure occurrence. In this context, this thesis aims to propose a general framework to support the maintenance management process. Four different applications are presented following the proposed framework and based on an Italian medium-sized oil refinery case study. In the first application, the proposed framework is adopted to derive the Association Rules describing components breakdowns after a stoppage of the oil refinery plant. The components that are most likely to break within a given time interval after a plant stoppage are identified to propose the best maintenance strategy. The second application regards a predictive optimization-based maintenance policy, based on the definition of Association Rules describing relationships among components' breakages. An integer linear programming model is formulated to select the optimal set of components to repair to improve the plant's reliability. In the third application, a bi-objective Component Repairing Problem is developed in order to reduce the impact on both the time to recover from a stoppage and the overall maintenance costs. The bi-objective Component Repairing Problem is solved through the AUGMEnted ε-CONstraint approach and through a bi-objective Large Neighbourhood Search meta-heuristic. In the fourth application, the Association Rule Mining and Social Network Analysis are contextually adopted to identify the hidden interactions between components that lead to a domino effect between failures. The conjunction of these two methodologies is useful because the Association Rule Mining helps identify the interaction among events and the Social Network Analysis support comprehension of such interactions.

Following the proposed general framework, Association Rule Mining and Social Network Analysis are also applied to pursue a second objective, that is extending the analysis of the production processes in terms of failures and related effects, through the results of the Failure Modes Effects and Criticalities Analysis which are used as the input of data-driven analysis. Two case studies are shown in this context: the first one regards an offshore and onshore plant for oil and gas extraction and storing; the second one regards a hydro-electrical power plant.

Chapter 1.

Introduction

1.1. Project background

Competition on a global scale, fast-changing customer needs, and shorter product life cycles require a high efficiency level in all industrial environments [1]. Indeed, production efficiency depends also on the reliability of the company's plants, for which the implementation of an effective maintenance policy becomes crucial. In fact, any interruption of the production flow may negatively affect the whole system and, therefore, profits [2]. Furthermore, the more complex the system, the higher the number of components potentially subject to breakage is.

In industrial sectors characterized by high operational risk, the occurrence of component failures is particularly critical since it may affect not only the plant's operations but also the integrity of the environment and people's safety [3]. Several actors are interested in ensuring that the integrity of production plants is maintained due to potential adverse consequences related to public health, safety, and heavy financial liabilities. In fact, in case of system failure, it represents a critical issue not only for the managers of the plant itself but also for governmental entities, consumers and stakeholders in general [4]. Process industries should also be particularly considered since researches show that, for example, in refineries, the maintenance and operations departments comprise about 30% of the total personnel and maintenance costs are the most impacting voice on the operational budget [5]. Being able of predicting future failures can provide valid support in this sense since the analysis of data related to the actual production processes ensures the possibility of having an accurate prevision [6] and, thus, this information can be used to deploy appropriate maintenance strategies.

As reported by the European Committee for Standardization [7], three main branches of approaches should be taken into account in defining the most appropriate maintenance policy for an industrial environment:

- corrective maintenance: an intervention is carried out after a breakage occurrence in order to restore the normal system functioning;
- preventive maintenance: maintenance is carried out at predefined intervals or conditions;
- predictive maintenance: maintenance is carried out according to significant characteristics like the breakage forecast, suggested by estimations of the degradation state, or the component breakage probability.

Several different strategies are still applied, but a growing focus is put on predictive maintenance. Predictive maintenance can be considered as the attitude to "use the actual operating conditions of plant equipment and systems to optimize total plant operations" [8]. In fact, predictive maintenance policies offer the opportunity of anticipating component breakages through the analysis of historical data and the implementation of proper algorithms. Specifically, the increasing data availability enabled by Industry 4.0 infrastructures facilitates the development of data-driven algorithms for asset availability and reliability maximization that allow continuous process control [9]. The main benefits harbored by the implementation of the predictive maintenance strategy involve both the increasing of reliability levels, reducing failures and breakdowns up to 70%, and the lowering of the total maintenance costs up to 40% [10].

Starting from these key points, the present thesis aims at developing a comprehensive strategy to support the implementation of data-driven maintenance policies based on the implementation of data analytics techniques that help in the Knowledge Discovery in Databases (KDD) process, i.e., in automatically analyzing data to bring out valid, novel, potentially useful and ultimately understandable patterns [11]. Since the KDD field is characterized by a number of possible techniques, a thorough investigation on the characteristics of the data available and of the different variables to analyze represents the first step for the selection of the most appropriate one [12]. The maintenance policies proposed in this thesis all rely on mining the Association Rules (ARs), a very popular and widely applicable methodology belonging to the KDD field. Two main reasons explain the rationale behind this selection, as recommended by Crespo et al. [13]: on the one hand, the Association Rule Mining (ARM) represents a powerful methodology to discover non-trivial, relevant relationships that are not immediately identifiable in a large amount of data; on the other hand, ARM is user friendly, since the results are provided intuitively so that their interpretation is rapid and intuitive. Moreover, there is no need to formulate research hypotheses.

Additionally, other approaches can be implemented after the ARM, such as decision support systems based on threshold definition, optimization approaches, or network analysis, in order to guide the decision-making process.

As the central methodology of this thesis, a four-layer framework is proposed to describe the conceptual maintenance management from the early stages of data collection until the implementation and performance monitoring of the data analytics-based maintenance strategies. Following the same framework, two different lines of research are deployed: the first one aims to define data-driven maintenance policies considering data collected from the production processes; the second one, instead, is devoted to the extension of the failure analysis traditionally carried out by the companies (e.g., the failure modes effects and criticality analysis). In this way, the data collected and pretreated are further analyzed to extract useful knowledge from them.

The remainder of this introductory section is dedicated to describing the fundamental aspects of the two research lines (Section 1.2 and 1.3) and providing a general outline of the rest of the thesis (Section 1.4).

1.2. Defining data-driven maintenance policies

The monitoring of a production process is connected to several variables that, if appropriately traced, generate a large amount of data. Defining the best maintenance policy represents a critical issue for all kinds of production plants. The process industry is significantly affected by this aspect, as all the activities are sequentially connected. Many variables like temperature, flow rates, level, and chemical characteristics of raw materials have to be measured [14] and monitored. Indeed, in the Industry 4.0 era, sensors are used by companies to gather a huge amount of data related to production, maintenance events, and component failures. KDD techniques can significantly support the automatic extraction of valid, useful, and unknown relations from a large amount of data [11], and they are suitable to manage such data. Consequently, they can successfully support the maintenance policy definition, capitalizing on the data collected along the processes.

In literature, models for data-driven maintenance already exist. Some of these models are used to implement Condition Based Maintenance solutions (e.g., [15,16]),others are used to implement models or simulations for predictive maintenance (e.g., [17,18]). The research focus is mainly on predicting the occurrence of component failures to reduce unexpected events and the consequent stoppage of the production processes. Although the existing research is valuable, specific frameworks supporting the maintenance management process to analyze and predict relationships between component failures and avoid them are not satisfactorily deployed. Indeed, less attention has been paid to developing a framework for the decision-making process to capitalize on the implementation of data-driven techniques, achieve satisfying levels of reliability, and avoid wasting resources using the available amount of data produced and collected during the production processes.

For these reasons, the framework proposed in this thesis aims to address this gap by introducing an innovative decision-making tool in this critical activity capitalizing on the vast amount of data available in the production environment. Four different approaches to the maintenance policy definition are deployed in the following of this dissertation, basing on the case study of an Italian medium-sized oil refinery:

1. In the first application, KDD is applied to derive AR describing components breakdowns after a stoppage of the oil refinery plant (e.g., the mass flow across the plant is interrupted). Considering the stoppages of the plant, the purpose of this application is to answer the following research questions: what are the components likely to break within a given time interval after a plant stoppage? Considering the probability level of breakdowns, is a predictive maintenance intervention preferable to a corrective one? [19]
2. The second application focuses on developing a predictive optimization-based maintenance policy, under the assumption that a component fails, based on the

definition of ARs describing relationships among component breakages. Furthermore, an integer linear programming model is formulated aiming to select the set of components to repair in order to improve the overall robustness to breakages of the plant, respecting both the given total repair time available and budget. An experimental campaign carried out on a real case study of an oil refinery and a detailed sensitivity analysis on some parameters of the mathematical model are used to evaluate the performances of the proposed approach [18].

3. The goal of the third application is to define a data-driven approach, based on the available data related to past failures, to predict the components that will break in a time interval after a stoppage occurs in order to increase the reliability of the whole plant. A bi-objective Mixed Integer Linear Programming (MILP) model for the bi-objective Component Repairing Problem (b-CRP) is formulated in order to reduce the impact on both the time to recover from a stoppage and the overall maintenance costs (e.g., the maintainer hourly cost and the component repair costs). The b-CRP problem is solved through the AUGMEnted ε-CONstraint (AUGMECON) approach and a bi-objective Large Neighborhood Search (b-LNS) meta-heuristic for efficiently addressing medium and large-sized instances, carrying out an experimental campaign on real-lifelike case studies inspired by an oil refinery plant. Data-driven analysis on the effectiveness of the b-LNS moves is also proposed [20].
4. Instead, the fourth application is based on the development of a data-driven approach simultaneously adopting the Association Rule Mining (ARM) and Social Network Analysis (SNA). The objective is to identify the hidden interactions between components that lead to a domino effect between failures. The implementation of the SNA is rather limited in the maintenance-related field, while the joint application of SNA and ARM is entirely lacking. The conjunction of these two methodologies is useful because the ARM will be used to identify the interaction among events and the SNA to define the nature of such interactions. In comparison to previous works, this framework allows researchers to identify

communities of nodes in order to analyze local and global paths and define the most influential entities.

1.3. Data-driven extension of failure analysis

Process plants are nowadays required to meet a high number of regulations, norms, as well as customer expectations for aspects like safety and environmental protection [21], especially companies dealing with high operational risks. In some cases, like in the oil and gas field, such plants are characterized by low investments in asset renewals and, thus, mainly composed of aging infrastructures and components [22]. As a consequence, there is a growing focus on defining ad-hoc maintenance frameworks able to ensure the execution of the operations safely [23]. The techniques traditionally applied for monitoring the operations and determining the maintenance approaches for improving the asset reliability can be accompanied by new methodologies more oriented to data analytics [13,24]. KDD can be used for extending the analysis of failure modes and effects since it represents a prominent enhancement opportunity of the maintenance policy improvement and risk reduction through the application of advanced techniques, expanding the knowledge achievable through the traditional approaches.

In this context, the second line of research pursued in this thesis aims at proposing a framework for extending the analysis of the production processes in terms of failures and related effects through the well-known and widely applied Failure Mode, Effect and Criticality Analysis (FMECA). The results of the FMECA are then used as the input of data-driven analysis. The FMECA represents a useful tool for identifying the potential failure modes of a system or process, their impact, and the deriving consequences on the global performance.

Many techniques and solutions have been proposed for strategic data-driven FMECA (e.g., [25–27]). Some of these models are used to automate the identification of failure modes (e.g., [28,29]), others are used to improve the risk assessment process (e.g., [23,30]). At the same time, some authors integrate FMECA and remaining useful life prognosis (e.g., [31,32]). Less attention has been paid to the introduction of data-driven frameworks for supporting the failure analysis. Although the existing research is valuable, a framework based on data-driven approaches that structurally analyze and predict the relationships among failure modes and effects, supporting the definition of the maintenance policy, is not present in current literature.

This gap is addressed by introducing an innovative procedure to support the failure analysis for enlarging the current body of knowledge, facilitating the visualization, and, thus, the understanding of previously unknown paths. In particular, the development of the framework relies on the simultaneous adoption of two data-driven techniques for analyzing the output data of the FMECA: the ARM and SNA. As previously mentioned, the ARM is applied to define the relationships among failure modes, their effects on the system, and the adopted maintenance tasks. The rationale behind the adoption of ARM is based on their intuitiveness, allowing their interpretation even for non-domain experts, and their applicability to different fields (market analysis, operations analysis, and control). Moreover, ARM does not require the formulation of hypotheses, allowing an unbiased analysis of the whole dataset. In this way, relationships that are not identifiable through the FMECA itself can emerge. In parallel, the SNA enables company managers to jointly explain the ARs through a graphical representation of the results: the network provides a more understandable data format for the decision-makers. The SNA supports the definition of the nature of the interactions represented by the ARs, identifying communities of nodes to analyze local and global patterns and locate influential entities.

Two case studies are proposed to present this approach:

1. The first one considers an offshore and onshore plant for oil and gas extraction and storing; the complete process from the data collection to the FMECA and its extension is described in detail, providing an extensive description of the procedure followed and the results obtained.
2. The second one regards a hydro-electrical power plant; the implementation of the proposed approach is presented in detail, while the stages upwards are treated in a less extensive way [33].

1.4. Outline

In the next chapters, the thesis is deployed as follows: Chapter 2 contains the literature review; Chapter 3 describes the general framework, while the applications of the two lines of research previously mentioned are presented in Chapter 4 and Chapters 5. Chaper 6 and 7 are respectively dedicated to the discussion and conclusions of the dissertation.

The thesis is based on the following papers authored by the candidate:

1. Antomarioni, S., Bevilacqua, M., Potena, D., & Diamantini, C. (2019). Defining a data-driven maintenance policy: an application to an oil refinery plant. International Journal of Quality & Reliability Management, 36(1), 77-97.
2. Antomarioni, S., Pisacane, O., Potena, D.,Bevilacqua, M., Ciarapica, F. E., & Diamantini, C. (2019). A predictive association rule-based maintenance policy to minimize the probability of breakages: application to an oil refinery. The International Journal of Advanced Manufacturing Technology, 105(9), 3661-3675.
3. Pisacane, O., Potena, D., Antomarioni, S., Bevilacqua, M., Emanuele Ciarapica, F., & Diamantini, C. (2020). Data-driven predictive maintenance policy based on multi-objective optimization approaches for the component repairing problem. Engineering Optimization, (in press).
4. Antomarioni, S., Bevilacqua, M., Ciarapica, F. E. (2021) Data-driven approach to predict domino effect between component failures: an asset maintenance framework and a case study on a process industry. Under review on "International Journal of Quality and Reliability Management"
5. Association rules and social network analysis for supporting the failure mode effects and criticality analysis. Under review on "Journal of Intelligent Manufacturing"
6. Antomarioni, S., Bellinello, M. M., Bevilacqua, M., Ciarapica, F. E., da Silva, R. F., & de Souza, G. F. M. (2020). A Data-Driven Approach to Extend Failure Analysis: A Framework Development and a Case Study on a Hydroelectric Power Plant. Energies, 13(23), 6400.

Chapter 2.

Literature review

In the following subsections, the literary contributions considered as state of the art for the development of the dissertation are proposed. The chapter is articulated as follows: section 2.1 describes, in general, the data-driven applications addressing maintenance or reliability issues; section 2.2 focuses the attention on the Association Rule Mining applications to the production management in general and to the maintenance aspects in particular. In section 2.3 and 2.4, respectively, the application of mono-objective and multi-objective optimization is reviewed. Section 2.5, instead, collects the main contributions concerning data-driven failure modes effects and criticalities analysis.

2.1. Data-driven applications for maintenance and reliability

KDD is an interdisciplinary field aiming at extracting important information and knowledge from a large amount of data [34]. Data Mining (DM) approaches represent a branch of the KDD techniques that can support companies throughout transforming data into value. The focus is on searching for hidden information, patterns, and tendencies in a large amount of data [35]. Therefore, DM acts as a facilitator to discover information from which extracting value form data is currently available [14].

Data are produced during almost all the organizational processes, from the design to the production scheduling, control, and maintenance [36]. According to Braha [37], DM techniques should be integrated into organizational processes, considering its objectives and potentialities together with the goals and weaknesses of the manufacturing environments. For instance, the design process involves the setting and selection of parameters, actions, and components [38] in order to make previsions, such as cost estimating: prior data are often applied to the cost estimation problem during the designing phases, and several algorithms have been developed and are currently applied to this end [38]. Even quality analysis, such as searching for causes for deteriorating product quality, can be performed through KDD techniques [39,40]. Furthermore, the application of DM techniques can provide proper support to process analysis: Maki and Teranishi [41] propose an automated DM system to detect anomalies in a manufacturing process in order to induce engineers to pinpoint and easily prevent their causes. Indeed, problematic issues can be found and resolved through DM technology as presented by Gardner and Bieker [42] in an application in the semiconductor industry.

Moreover, the application developed by Batanov et al. [43] communicates to the user maintenance policy suggestions, together with machine diagnosis and maintenance scheduling for the analyzed devices. Instead, Romanowski and Nagi [44] manage to identify subsystems responsible for low equipment availability and, based on these results, provided a preventive maintenance schedule. Bumblauskas et al. [45] define a smart maintenance decision support system integrating optimization algorithms and analytic decision models in order to provide useful suggestions on maintenance execution, while Manco et al. [46] perform an outlier-based fault prediction through the study of non-normal signals provided by sensors and validated their approach through an experimental case study. The review of the principal condition monitoring techniques applied to fault detection of offshore wind turbine presented by Kabir et al. [47] is a further example of DM applications to maintenance policy. Instead, Antony and Nasira [48] propose a cluster analysis to perform a predictive analysis on board of train vehicles. In addition, Sammouri et al. [49] present a methodology aiming to predict rare failures mining temporal data provided by sensors installed on commercial trains. Even Jin et al. [50] dedicate their work to the railway field, developing a procedure for the predictive maintenance of railway point machines. Ming [51] creates a maintenance management system for urban transit rails. Indeed, through the application of artificial neural networks and decision trees, the author was able to supervise the equipment and to mine valuable information.

In a paper by Sipos et al. [52], a data-driven framework based on multiple-instance learning is applied to predict equipment faults: in particular, through the mining of equipment event logs, they extract the operational information useful for the predictive activity. Wang [53] implements a fault diagnostic and prognostic system, collecting data through DM techniques and exploiting artificial neural networks to train and validate the model.

Furthermore, Gröger et al. [1] present an innovative DM methodology aiming at the optimization of the whole manufacturing process, describing both conceptual and practical cases. Bevilacqua et al. [24] develop an analytic model to carry out IoT-based energy management, aiming to empower the decision-making process by integrating data provided by different smart devices.

Despite the wide application of DM techniques to manufacturing processes, this theme is not widely developed concerning oil and gas refinery plants, even if it would represent a successful discriminant in this field [54]. Indeed, according to Köksal et al. [55], only 4% of the DM applications to the manufacturing industry regard coke or petroleum refineries. In refinery, DM could be applied to analyze the influence of some variables on product quality, create rules to manage the manufacturing process, and predict price changes or requirements of different kinds of oil [56]. Zhong and Wang [56] propose a theoretical combination of computer integrated management systems and tools. Wang and Gao [57] develop an indicator to support the maintenance decision-making process, exploiting the Internet of Things and apply it to an oil transfer station. Friedemann et al. [58] describe some applications of prognostic and diagnostic systems in Energy and Rail fields, highlighting the need for adapting the operating plans to the information extracted. Moreover, they apply a similar management system to condition monitoring of subsea facilities in oil production processes to improve the availability and reliability of the infrastructure. Li et al. [59] propose a prototype system based on a rough set to deal with incomplete data in fault diagnosis and applied it to centrifugal pumps of a refinery plant. In order to predict the deposition of scales in oil wells and avoid the unavailability of the equipment, it is found that training a support vector machine can represent a valuable solution. Indeed, an appropriate maintenance strategy can be defined according to the scaling values predicted by the model [60]. Hu et al. [61] propose a methodology for oil pump fault detection, integrating multifractal theory and Mahalanobis–Taguchi system: in particular, they aim at predicting failures through vibration signals monitoring. Moreover, they highlight the necessity of storing real-time signals in an integrated diagnosis database in order to enable the application of advanced fault detection techniques.

2.1.1 Association Rule-based applications for maintenance and reliability

ARM is a valuable research area of KDD and can be successfully used for effectively representing relationships among data [62]. According to Wang [63], it can be considered for performing predictive data analysis since it relates a specific variable to others included in the same dataset. As stated by Buddhakulsomsiri et al. [64], in fact, extracting the ARs allows deducing attribute-value information contained in a dataset but not immediately identifiable due to the amount of data. Intuitiveness is one of the AR strengths, together with its applicability to several fields. In fact, various applications exist, ranging from customers' buying habits [65] to product design specifications, as well as production process control [66]. For instance, ARM can be applied to improve the design process and define the most appropriate geometric dimensions [35]. In addition, Chen [67] use ARM to solve the problem of cell-formation, according to group technology requirements, evidencing the ability of this method in determining quality solutions. In particular, this method is applied to the binary version of the cell-formation problems, and it result in being a satisfying procedure both for large and small-scale problems.

The application of ARM to the manufacturing field can be related to the purpose of enhancing overall performance. In order to pursue this aim, ARs can be applied to frequent patterns extracted from industrial processes, as they represent a useful methodology in disclosing industrial failures [68,69].

Djatna and Alitu [70] capitalize on the extraction of ARs to develop a total productive maintenance strategy, obtaining an increase in the effectiveness of maintenance response and efficiency considering time and costs. Bastos et al. [71] study a decentralized predictive maintenance system aiming to forecast the possibility of a breakdown to increase the reliability of the system. Furthermore, a manufacturing defect detective model is proposed by Chen et al. [72]: AR mining is applied in order to analyze the existing correlations between the combination of machines and defects and, integrating this procedure with a root cause machine identifier, the root cause could be detected. Wang et al. [73] study the generation of associative rules for manufacturing process planning to improve the performances previously obtained through a fuzzy decision technique and entropy-based analysis method: indeed, they combine the variable precision rough set and fuzzy clustering. Another relevant application of the rule mining algorithm to manufacturing processes is the one proposed by Agard and Kusiak [74]. Indeed, their algorithm aims at selecting subassemblies through the analysis of orders previously received from the customers. This application could ensure an improvement in performance in terms of delivery times to the contractor. Moreover, ARM can be applied to fault detection in assembly operations [75], achieving progress in terms of quality of the assembly process thanks to the monitoring or avoiding critical sequences [66]. An applicative example to a drill production process is also present in literature: results show that the approach based on AR mining is useful in providing important information on faults and related causes [68].

The development of an AR-based conceptual model is also found to be valuable in detecting the impact of human practices on risky situations in a refinery plant [3]. A further application of AR in the refinery field is analyzed by Li et al. [76]: indeed, they show that basing AR on fuzzy systemic clustering could allow extracting useful advice on the production process, even if some of the rules extracted provide unnecessary or superfluous information.

2.1.2 Data-driven maintenance policy definition: a purpose-based classification

Different data-driven methods for predictive maintenance are analyzed in this literature review. Most of these methods implement condition-based maintenance solutions, while others implement modeled or simulated predictive maintenance (statistically predictive). Both these approaches aim to define critical assets for which a physical plant owner should allocate maintenance resources. The condition-based methods focus on time and/or condition monitoring data (often provided with sensors) and statistical trending. In contrast, the latter is focused on a prediction or simulation based on an expected potential for failure [45].

Another important classification that can be made on data-driven models is related to the objectives that these models seek to achieve. For example, some models aim at predicting the remaining useful life of an asset (e.g., [77,78]), even considering the lifespan of a part and the lifetime maintenance cost [79], and others again attempt to predict failures or their causes (e.g., [9,80,81]).

According to this classification, Table 1 aims to describe the main literature contributions proposed in the field of the maintenance policy definition regarding data-driven predictive maintenance techniques and the objectives of the papers. Six main objectives can be identified in analyzing such contributions: fault prediction, fault detection and diagnosis, optimal maintenance schedule definition, equipment reliability and availability, normal behavior modeling, and, lastly, Remaining Useful Life (RUL) estimation. Regarding the data-driven techniques, instead, seventeen of them are taken into account. As presented in Table 1, Neural Networks are widely applied in all the fields described by the six objectives. Indeed, for their versatility in modeling all kinds of processes, they can be used for modeling several classes of problems. For instance, Lopes Gerum et al. [17] apply a Recurrent Neural Network to study rail and geometry defects to schedule maintenance interventions. The Artificial Neural Network deployed by Bangalore and Tjernberg [82], instead, serves as a fault detector, as well as the radial basis function neural network employed in Gharoun et al. [83]. Among the other techniques, the Support Vector Machine (SVM) and Markov models result in being widely used in several applications. For instance, Baptista et al. [84] apply SVM to predict the RUL in the aeronautic field, while Medjaher et al. [85] pursue the same objective through Gaussian hidden Markov models in studying bearings' useful life. Chen et al. [86], instead, use Hidden Markov models for RUL estimation as well as to schedule maintenance interventions.

According to this literature review in the asset maintenance field, the research focus is mainly on predicting the occurrence of component failures in order to reduce unexpected events and the consequent stoppage of the production processes. Thus, awareness in the decision-making process is mandatory for achieving satisfying reliability levels and avoiding the waste of resources [87].

Social Network Analysis aims to investigate the features of social structures relying on the network and graph theory [88]. Hence, SNA application mainly belongs to the sociological field [89]. It has been applied, for instance, to study the information and knowledge flows in the construction project teams, to improve collaboration [90], and to identify how, in the construction industry, safety communications flow within the local workers and ethnic minorities [91]. In the same sector, the fatalities are analyzed through the SNA to identify common root causes [92]. The most effective application of SNA for the current analysis can be found in Kim et al. [79], where the synchronous replacement of components driven by the life-cycle cost analysis is proposed.

The combination of the ARs and SNA was first proposed to study environmental risk management as a framework for the control and improvement of the company's environmental performance analyzed [93] and for monitoring the human factor risk management [3]. Integrating these techniques can provide a valuable methodology for having a deeper understanding of the relations among events through graphic representation. Indeed, as shown in other application fields (e.g., human factor risk management and environmental risk management), they result in being successful.

Table 1 Summary of literature contributions classified by their objectives and data-driven techniques applied.

Data-driven techniques	**Objectives**					
	Fault prediction	***Fault detection and diagnosis***	***Optimal maintenance schedule***	***Equipment reliability and availability***	***Normal behavior modeling***	***RUL estimation***
Support Vector Machine	[94], [80], [95], [96]	[97], [98], [99]			[13]	[84], [77]
K-nearest neighbors	[80]	[100]				
Regression	[80], [96]	[101]				
Neural Networks	[80], [102], [82], [83]	[97], [99], [101]	[16], [103], [17]	[16], [104], [103]	[102], [13]	[105], [78], [106]

Random Forest	[80]		[17]		[13]	
Instance-based learning			[107]			
Naive Bayesian classifier			[107]			
Decision trees	[44]		[107], [44]	[44]		
Logical Analysis of Data			[15]			
Adaptive neuro-fuzzy inference	[83]	[98]				
Association Rules	**This thesis**	[75], [71], [72], [68]		[70]	[13]	
Case-based reasoning	[108]					
Anomaly detection algorithm	[95]	[95]				
Markov models	[45], [17]	[109]	[110], [86]	[45]		[85], [86]
Clustering	[96]	[109]				
Bayesian Networks		[81], [87]	[111]			
Social Network Analysis	**This thesis**		[79], **This thesis**			

2.2 Mathematical programming applications in maintenance activities

Defining the best maintenance policy can become complex and, at the same time, significant for improving the performance of any company. In this sense, the application of mathematical optimization-based approaches is particularly interesting since it can contribute to both a cost reduction and a utilization level increment [112]. Moreover, optimization methods can be applied to solve problems where a large amount of data is used and/or require real-time decisions [113]. For example, Alkamis and Yellen [114] formulate an integer linear programming (ILP) model for preventive maintenance in oil refineries aiming to maximize its utilization level, although they do not use AR mining. Pistikopoulos et al. [115] propose a mixed-integer linear programming (MILP) model for simultaneous design, production, and maintenance planning. Similarly, Goel et al. [116] develop a MILP model integrating design, production, and maintenance plan, focused on improving the operational availability at the design stage by selecting more reliable equipment. As noted by Alrabghi and Tiwari [117], scientific contributions exist in the literature in which simulation-based optimization approaches have been successfully applied to maintenance policies in several application fields. For example, Allaoui and Artiba [118] use a simulation-optimization approach for a flow shop scheduling problem subject to maintenance constraints, due dates, and system availability. Moreover, simulation-optimization techniques can also be applied to deal with both inventory control and maintenance planning [119,120]. The combination of preventive maintenance and statistical process control can also be addressed through simulation-optimization approaches, as in Cassady et al. [121], as well as maintenance scheduling and production control [122]. Tagaras [123] formulates a combined model for process control and maintenance activities under a Markovian distribution deterioration hypothesis. Moreover, the analytical hierarchy process is combined with the goal programming for centrifugal pump maintenance in a refinery plant [124]. Lee et al. [125] propose a model to optimize the job scheduling in a multi-machine environment. The aim of the model is to define the optimal due date of each job, minimizing the total earliness and tardiness costs, and the optimal timing for maintenance activities. Kenne et al. [126] develop a near-optimal policy using numerical techniques for production planning and corrective maintenance intervention scheduling in a manufacturing system. The work of Vilarinho et al. [127] aims at finding the optimal replacement interval through the integration of the analysis of components' reliability and an optimization model for total cost (e.g., preventive replacement cost and failure replacement cost) minimization. Similarly, Mokhtari et al. [128] solve a maintenance and production scheduling problem through the formulation of a MILP model, whose objective is the minimization of the total unavailability of the system.

Irawan et al. [129] formulate an optimization model for routing and scheduling offshore wind turbine maintenance to minimize cost minimization. For this purpose, they propose a solution approach based on the optimization of a MILP model. The minimization of maintenance costs and systems interruptions are frequently analyzed together. For example, Laggoune et al. [2], instead, formulate a model for reducing the whole system down-times and maintenance costs through the preventive replacement of groups of components. Xia et al. [130] develop a maintenance procedure to reduce the total maintenance costs of the production system, scheduling the optimal time windows for periodic interventions.

In contrast, in Chalabi et al. [131], a particle swarm-based optimization approach is applied to minimize the total maintenance cost and the maximization of the process availability. Wang and Liu [132] address both the minimization of production makespan and the unavailability of the production process, formulating a multi-objective optimization model and solving it through an adaptation of the non-dominated sorting genetic algorithm II. Similarly, Hadjaissa et al. [133] concentrate on both the scheduling of the maintenance activities and the makespan minimization. They apply a genetic-based algorithm to a hybrid renewal power system. In addition, Shafiee and Sorensen [134] propose a cost-effective maintenance strategy for both reducing the interruption of systems operating conditions and limiting the maintenance costs. In the literature review presented by Ding and Kamaruddin [135], it is remarked that the focus of the studies is often on the certainty degree of maintenance policies. In particular, they distinguish among the models assuming future events certainty, those that assign a risk-level to possible future states, and the ones under uncertainty that specifically assume a probability of the occurrence of future events. For example, Xia et al. [136] formulate a condition-based predictive maintenance model for cost and availability optimization, incorporating the uncertainty related to the components' degradation. The optimization model formulated by Ilgin and Tunali [137], instead, takes into account the risk category. Indeed, they adopt a simulation-optimization approach based on the genetic algorithm, estimating the crossover probability through a factorial analysis.

2.3 Multi-objective mathematical programming in maintenance

This section describes the main literature contributions proposed in the maintenance field, focusing on multi-objective mathematical programming models and solution methods. Maintenance is usually a very time-consuming activity from the production objectives point of view since it typically requires a system stoppage. However, delaying maintenance interventions to avoid interrupting the production flow may significantly increase the failure probability [138]. Hence, owing to these contrasting criteria, several literature contributions apply multi-objective optimization techniques for designing effective maintenance policies. For example, Ruiz, García-Díaz, and Maroto [138] propose Ant Colony Optimization (ACO) and Genetic Algorithms (GAs) for preventive machine maintenance, minimizing the completion time of the last job in the production schedule. Marseguerra et al. [139], propose a GA for a condition-based maintenance policy, determining the optimal degradation level for a preventive maintenance policy, maximizing the profit and the availability simultaneously. They describe the model predicting the evolution of the degrading system through Monte Carlo Simulation (MCS). In Kumar et al. [109], a predictive tool aimed at deciding the optimum condition-based maintenance policy is designed. Specifically, the authors propose a semi-Markov process in order both to model the steady-state availability analysis of mechanical systems and to evaluate the optimal condition-monitoring interval. This interval is then used for maximizing the system availability through a GA approach. The problem of the best maintenance inspection interval identification is tackled by Marseguerra et al. [140], considering both the maximum system reliability and the minimum variance of the model parameters, proposing a multi-objective GA. A similar approach is presented in Huang et al. [141], together with a mathematical programming model for maximizing the system reliability and minimizing its costs, simultaneously. Okasha and Frangopol [142] formulate the problem of selecting the best maintenance actions by mathematical programming, maximizing the system reliability while minimizing its redundancy and the life-cycle cost. A non-dominated sorting GA (NSGA-II) is also designed.

Mathematical programming is used in Min et al. [143] for defining the best preventive maintenance policy, both minimizing maintenance costs and maximizing the reliability of a high-speed railway power system. A Chaos Self-adaptive Evolutionary Algorithm (CSEA) is also proposed. Considering the same objectives, Raad et al. [144] define a maintenance policy for water distribution systems, proposing A Multi-ALgorithm Genetically Adaptive Multi-objective (AMALGAM) search, outperforming existing approaches like NSGA-II. Similarly, Berrichi et al. [145] design an ACO algorithm for both maximizing the production system availability and minimizing the makespan. Carlos et al. [146] address a maintenance problem in a nuclear power plant, solved by a Particle Swarm Optimization (PSO) algorithm. In the same application context, Gjorgiev et al. [147] compare four different versions of GA (weight-based classical GA, weight-based steady-state GA, weighted-sum GA, and NSGA-II). Tian et al. [148] use Physical Programming (PP) for simultaneously maximizing the reliability and minimizing maintenance costs in condition-based maintenance. In Loganathan and Gandhi [149], a PSO algorithm under reliability constraints for minimizing maintenance cost is designed. Moghaddam and Usher [150] formulate a multi-objective optimization model to determine the optimal preventive maintenance and replacement schedules in a multi-component system. Both a generational GA and a Simulated Annealing (SA) algorithm are also designed. A preventive maintenance optimization-based approach is also proposed by Moghaddam [151] with the aim of determining the optimal maintenance schedules in production systems. The formulated model is then solved through a procedure that combines MCS and Goal Programming (GP). Ebrahimipour et al. [152] design an exact approach, based on the Weighted-Sum (WS) method, to schedule preventive maintenance achieving minimum cost and maximum reliability. An interactive fuzzy multi-objective linear model for the minimization of the maintenance costs and of the scheduling tardiness is formulated in Seif, et al. [153]. The selection of the most appropriate maintenance policy can also be addressed through the Analytic Hierarchy Process (AHP) since it requires several criteria to be evaluated simultaneously. For example, Bertolini and Bevilacqua [124] apply AHP for assessing the maintenance alternative policies (i.e., corrective, preventive, and predictive) considering three specific criteria, i.e., the occurrence, the severity, and the detectability. A GP model is then formulated for selecting the best maintenance policy for each centrifugal pump under budget and human resources constraints. However, they do not focus on the

CRP. AHP can also be combined with GP [154]: firstly, AHP is applied to prioritize the possible maintenance policies, comparing them in terms of cost and risk; then, selecting the best maintenance policy is performed through GP. Moghaddam [155] compares the performance of five GAs for optimizing the operational costs and the overall reliability of a Computer Numerical Control (CNC) machine. Instead, Fan and Xia [156] apply a GA to solve a multi-objective optimization problem related to an energy-efficiency building envelope retrofitting plan for maximizing the energy savings and the net present value of the investment while minimizing its payback period. A maintenance plan in building retrofit is addressed in the multi-objective model proposed by Wu et al. [157] and solved through Multi-Objective Neighborhood Field Optimization (MONFO), in which the retrofit cost, the energy-saving, and the net present value are optimized simultaneously. Recently, imperfect maintenance policies have also been proposed. For example, in Su and Liu [158], NSGA-II is applied to solve a multi-objective imperfect preventive maintenance optimization problem in the context of electromechanical products. Table 2 summarizes the main literature contributions. Specifically, for each article, it is shown whether the maintenance policy proposed is predictive (PM) and/or data-driven (DD) and/or multi-objective (MO). The third column specifies the objective(s) considered for optimization, whereas the last column reports its approaches. It is worth noting that only Antomarioni et al. [18], which proposes optimization approaches for system maintenance, actually shows predictive features[1]. In fact, they propose a predictive maintenance approach in which the CRP is mathematically formulated through Integer Linear Programming (ILP) but only for minimizing the breakage probability (single-objective), under the limiting assumption that the breakage of a specific component occurs.

[1] See Section 4.3 for the complete deployment of this application.

Table 2 Main literature contributions on optimization approaches for system maintenance

Paper	PM	DD	MO	Objectives	Approaches
[140]			✓	Reliability, Failure, Uncertainty	GA
[138]			✓	Makespan	ACO, GAs
[159]			✓	Reliability, Cost	GA
[143]			✓	Reliability, Cost	CSEA
[142]			✓	Reliability, Cost	GA
[144]			✓	Reliability, Cost	AMALGAM
[154]			✓	Cost, Risk	AHP-GP
[145]			✓	Unavailability, Makespan	ACO
[150]			✓	Reliability, Cost	GA, SA
[146]			✓	Unavailability, Cost	PSO
[160]			✓	Unavailability, Cost	LGP ε-constraint
[147]			✓	Unavailability, Ageing, Cost, Uncertainty	GAs
[161]			✓	Unavailability, Cost	ε-constraint
[162]			✓	Reliability, Cost	GA
[152]			✓	Reliability, Cost	WS
[155]			✓	Reliability, Cost	GAs
[157]			✓	Retrofit cost, Energy saving, Net present value	MONFO

[156]			✓	Energy saving, Payback period, Net present value	GA
[153]			✓	Cost, Tardiness	GA
This thesis	✓	✓		Breakages Probability	ILP
	✓	✓	✓	Reliability Max Repair Time	AUGMECON b-LNS

2.4 Data-driven failure modes effects and criticality analysis

FMECA analysis is a widely applied technique in the maintenance management field. Some applications involving its joint implementation, together with data-driven techniques, can be found in the literature. For instance, in Savino et al. [163], the fuzzy inference is used together with the FMECA to perform the criticality evaluation taking into account both safety aspects and the production performance. In Tso et al. [29], the problem of the automatic identification of the failure modes to perform an efficient FMECA is addressed through a framework based on hardware description languages and knowledge-based fault models. The automation of the FMECA has also been addressed in Grunske et al. [28] through the implementation of behavior trees that support the failure mode identification by injecting faults data. In the attempt to identify all the possible relationships among components and failure modes, in Xu et al. [164], text mining is applied to identify the potential failure modes related to a specific component. Data regarding the remaining useful life of components can be adequately analyzed through data mining techniques and used to update the FMECA data, monitor the modification of the risk of failure, and be used in the following projects [31,32]. Bayesian Networks have been successfully applied with the FMECA, as presented by L. Liu et al. [77] and Ben Said et al. [30]. Some approaches for improving the prioritization of components failure risk have also been proposed: for instance, in Chang and Cheng [165], who apply the fuzzy ordered weighting average and the DEMATEL methods are used for calculating the risk assessment, and H. C. Liu et al. [166], where the risk assessment evaluation is performed through the definition of fuzzy digraph and matrix. In Khorshidi et al. [167], the overall failure index is used as a guide for the optimal selection of the improvement actions to implement to achieve maximum system reliability. As shown in Li et al. [168] and Lv et al. [169], the domain experts' judgments on failure modes can be included in a cloud model so that they can be compared through multi-criteria decision-making approaches, while in Ma et al. [170] the quality function deployment and the FMECA are integrated to analyze improvement areas of the components, taking into account the reliability aspects and customer expectations.

2.5 Rationale for the development of the data-driven framework: research gaps identification

According to the literature review carried out in the previous sections, the maintenance management field receives broad attention from researchers and practitioners. Considering the ARM implementation in the maintenance field, their application is still limited, especially concerning the oil and gas sector.

Although integrating data mining techniques with those provided by operations research and specifically, by mathematical programming, is not a new topic (e.g., [171,172]), to the best of the author's knowledge, this work represents the first contribution in which ARM and mathematical programming are combined to each other for defining a predictive maintenance policy of an oil refinery plant. In addition, a data-driven predictive maintenance policy is proposed, and the Component Repairing Problem is modeled through multi-objective mathematical programming for simultaneously maximizing the system reliability and minimizing the maximum repair time.

The application of the ARM in combination with the SNA in analyzing failure data is novel, as well as in improving the FMECA process. However, these applications are more focused on the improvement of the FMECA itself. Instead, the proposed approach addresses the lack of a data-driven framework to support the definition of the maintenance strategies, considering the possible cascade effects related to the occurrence of hazardous events and their impact on plant reliability.

Noteworthy, the main scope of the thesis is addressing the research gap characterized by the lack of a comprehensive framework guiding the decision-maker during the whole maintenance management process. Indeed, the existing contributions mainly focus on addressing a specific issue rather than providing a complete view of the decision making process.

Chapter 3.

General framework development

In this chapter, the general framework object of the thesis is developed and described. Specifically, the insights extracted from the literature review are used in order to define a general framework guiding the decision-makers throughout the maintenance management. Some researches proposed that a general predictive maintenance framework should be organized in three steps, namely data acquisition, processing and maintenance decision making [173]. In this work, instead, the procedure is furtherly detailed, both considering the three proposed phases in the literature and extending them. Indeed, the proposed framework involves the data collection, management, breakage probability estimation, decision support model definition, implementation, and control of the approach proposed.
In Figure 1, a visual representation of the maintenance management system is proposed. In particular, as previously mentioned, four layers can be identified, each of whom dealing with a specific activity related to data management, processing, and analytics.

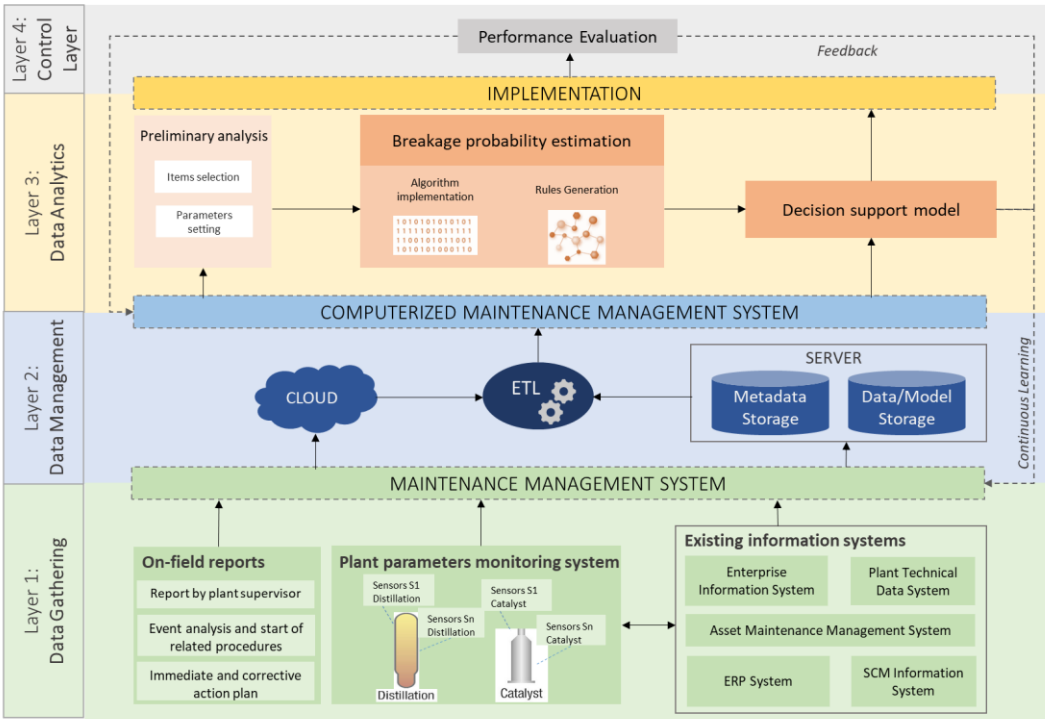

Figure 1 General framework for data-driven maintenance management

3.1. Data gathering layer

The data gathering layer represents the first layer of the model, being the aim of the framework - the development of data-driven support of the decision-making process. In this sense, the quality of the entire framework relies on appropriate data collection. In this approach, three different macro-categories of data sources are considered.

On-field reports: monitoring the operations of an industrial plant is fundamental to control it; maintenance department supervisors have to check the sub-plants in order to notice and register any possible malfunctioning affecting its performance. During each inspection, the supervisor has to record all the relevant information and create a report; in case of abnormal events, there are procedures to follow and, possibly, immediate corrective interventions to perform, annotating these details. Usually, such reports follow unstructured or semi-structured paths since free-text annotations often characterize them, making their computerization non-univocal or, at least, complex. However, these data contain useful information due to supervisors' broad knowledge of the process.

Existing information systems: data coming from the on-field reports are integrated with the information systems of the company. Information on both the normal operating conditions and on adverse events are stored in such systems, like:

- EIS: the Enterprise Information System and records administrative data, work orders type (e.g., specific replacement of a component, lubrication…) and the related costs, purchasing orders, corrective interventions, their details, and costs;
- ERP (Enterprise Resource Planning): it stores information regarding resource and inventory management;
- Plant technical data system: data regarding product and process characteristics in terms of design and functioning;
- Asset maintenance management system: data stored in this information system regards all the maintenance activities carried out in the plant (corrective, preventive, or predictive), highlighting the date, the kind of intervention, the broken component (or set of components), the team in charge of the intervention, the duration of the intervention, etc.
- Supply chain management system: it records the data from suppliers and customers– in both cases regarding their general data, order data, real-time status, and quality rate.

Plant monitoring system: the functioning of the plant is monitored by a series of sensors measuring the production process data, like flow, pressure, and density. Besides, some of the components are equipped with embedded sensors so that their state is currently monitored, generating a large amount of data to analyze. Each of them has its own IP address and communicates with a cloud-based application; hence it is fundamental that cloud resources are allocated efficiently [174]. This information integrates the systems mentioned above, giving a complete overview of the process.

Data coming from the sources mentioned above have to be integrated in order to extract information and knowledge for making informed decisions. Thus, in the second layer, the management of the collected data is performed.

3.2. Data management layer

The data management layer, that is the second layer of the framework, aims at integrating data coming from different data sources into a unique one. More specifically, the information contained in the company's server and cloud-based applications have to be merged, cleaned and transformed, in order to create a unique source to perform the analysis in the last step. In this process, all the possible problems affecting the data have to be solved to analyze only a consistent data set. For example, errors in recording the measures (e.g., misreading, repetitions) have to be removed or replaced with valid ones. In contrast, heterogeneities generated by different terminologies used in each source have to be standardized. In addition, some data could be filtered, selecting only the attributes considered relevant for the aim of the analysis. The Extraction, Transformation, and Loading (ETL) process is carried out to integrate data from the original sources to a data warehouse, the Computerized Maintenance Management System (CMMS). Specifically, the plant monitoring and supply chain management data are extracted from a Cloud Application, while the other ones come from the company's server. The use of a CMMS ensures a global view of a company's operations. It allows the collection of clean data from all the sources, integrates them, and provides an aggregation of historical and real-time conditions. This technology is particularly useful in the case of a high number of components to monitor and maintain [175]. In this way, the predictive analysis can be performed relying on a reliable, integrated data warehouse.

3.3. Data analytics layer

In the data analytics layer, the data collected and integrated into the CMMS can be studied in order to build the model for the data-driven predictive maintenance policy definition. In general, three main phases can be highlighted in this layer:

1. Preliminary analysis;

2. Breakage probability estimation;
3. Decision support model definition.

In the following paragraphs, the three steps are detailed.

3.3.1 Preliminary analysis

Considering the data collected, prepared, and stored in the CMMS, a study of the failures occurring on the asset object of the analysis is required. This step is important for the identification of both critical components and their relationships. Moreover, identifying only the information sources useful for the analysis, among the ones integrated during the data management step, is vital to be able to carry out a meaningful study. Indeed, depending on the specific objective of the implementation and the system's boundaries (e.g., extending the implementation to a portion of the plant versus the whole system), some data may not be useful. In other cases, it might be necessary to limit the study to the most significant components of the production system. Given the objective of the data-driven framework, it is important to identify the sets of components frequently failing together, i.e., within a given time interval. In this sense, the methodology selected to study such relationships is the Association Rule Mining in most case studies. Only in one case, a specific algorithm for the breakage probability estimation is introduced.

Before mining the Association Rules, two aspects have to be taken into account:

a) the components included in the study: depending on the characteristics of the asset, it is crucial to define whether all the components are relevant for the analysis or only some of them. Indeed, the study aims to create an interrelation among the critical components: the components whose replacement does not impact the working conditions of the industrial plant might be excluded from the analysis in order not to lose the focus on the critical ones.
b) the limit for the time interval: since it is stated that the aim of the framework is to identify the relations among components frequently failing together (i.e., within a given time interval), the temporal dimension has to be limited in order to provide interesting results. Indeed, considering a time interval that is too short does not provide any significant connection. Having an overly-long interval does not

provide any connection in the opposite sense, which presents false relations among failures. Even in this case, the expertise of the decision-makers is crucial.

Once the preliminary analysis is carried out and the scenario in which the framework is implemented is defined, the ARM or any other algorithm for the breakage probability estimation can be performed.

3.3.2 Breakage probability estimation

Breakage probability estimation can represent a challenging objective. This estimation is mainly based on the Association Rule Mining (ARM) in the proposed general framework. ARM aims to identify hidden and previously unknown relations in a vast amount of data, supporting the decision-makers in their processes. A formal definition of the Association Rules (ARs) and the procedure to mine them is explained in the following.

Let $K=\{k_1, k_2, \dots, k_n\}$ be a set of n binary attributes named items and $T=\{t_1, t_2, \dots, t_m\}$ be a set of m transactions. Each transaction t_i is unique and contains a subset of the items (itemsets) selected from K. In our framework, an item is a component of the analyzed asset. In contrast, a transaction is a set of components failing within a defined time interval. As defined by Agrawal et al. [176], an Association Rule (AR) is an implication $\alpha \rightarrow \beta$, such that α and β are itemsets ($\alpha, \beta \subseteq K$) having no common items ($\alpha \cap \beta = \emptyset$). In other words, given a time interval of one week, the rule $\alpha \rightarrow \beta$ is defined if and only if component β fails within one week from the failure of component α. The strength of the rule can be determined through several metrics, among which, we recall:

- supp $(\alpha, \beta)=\frac{count\{\alpha\ U\ \beta\}}{m}$; the support of the rule, which is defined as the set of transactions containing both α and β. Remarkably, this measure represents the joint probability of having α and β in a transaction ($P(\alpha, \beta)$); hence, it measures the statistical significance of the rule [176].
- conf $(\alpha \rightarrow \beta)=\frac{supp\{\alpha,\ \beta\}}{supp\{\alpha\}}$; the confidence of the rule, instead, is the set of transactions containing α, which also contain β. In this sense, the confidence can be seen as the conditional probability $P(\beta \mid \alpha)$, so it provides a measure of the rule's strength [35].

The ARM is performed according to the following roadmap:

(1) Define the frequent itemsets, namely the itemsets appearing in T more frequently than user-specified minimum support; in this work, the algorithm selected to perform the frequent itemset mining is the FP-growth [177].
(2) Considering each itemset IS defined in the previous step, all the ARs A→B are generated such that A U B = IS.
According to the aim of the study, the interest is in creating the relations among components frequently failing together.

3.3.3 Decision support model definition

Once the ARs are mined, a model must be defined to support decision-makers in capitalizing on them. As mentioned in the previous chapters, decision making can be based on different approaches. In this thesis, four different implementations are proposed:

1. The first approach adopted relies on the definition of specific thresholds by the decision-maker;
2. The second one adopts an integer linear programming optimization approach;
3. The third one introduces an integer non-linear programming with two solution approaches;
4. The fourth one relies on Social Network Analysis.

For the sake of clarity, the theoretical aspects are articulated in Chapter 4 and 5, together with the explanation of the case studies.
In addition, it is also shown how the same general framework can be applied for the extension of the failure analysis.

3.4. Control layer

The last layer of the data-driven framework involves the implementation of the proposed methodology and its implementation. During this phase, it is necessary to assess whether the implementation of the analytics approach is compliant with the expectations or assessing the sensitivity of the decision model. Furthermore, during the actual implementation of the data-driven approach, more data are produced that can be useful for future modification of the maintenance strategy. In addition, during such steps, data regarding the new implementation are collected and integrated to the organizational datasets, so that they can be furtherly processed.

Chapter 4.

Research approach applications to the case study of an oil refinery

This chapter is dedicated to the implementation of the proposed research. In the following sections, the case study involving the use of different decision support models is described. The decision support models presented in Sections 4.1, 4.2, 4.3, and 4.4 are based on the same case study, i.e., an Italian medium-sized oil refinery. The specific details useful for the development of each implementation are reported at the beginning of the related section.

4.1 Data-driven maintenance policy through a decision support model based on user-defined thresholds

4.1.1 Data gathering and management

The case study used for deploying the current application refers to an oil refinery characterized by a processing capacity of 3,900,000 tons/year of crude oil, that is about 85,000 barrel/day. The refinery's storage capacity is more than 1,500,000 m^3, and the land-based shipping system is characterized by a total capacity of more than 12,000 tons per day. A fixed sea platform is located 16 km from the coast, and it is able to accommodate tankers up to a tonnage of 400,000 tons. Furthermore, an island with a double mooring for ships up to 90,000 tons is located 4 km from the coast, while a pier for short-sea shipping is directly connected to the refinery and is equipped with three mooring points. In Figure 2, the production process executed in the refinery is schematized: noteworthy, the plant is divided into 11 sub-plants dedicated to specific functions.

The refinery receives about 340,000 tons of crude oil per month, and it goes in input in the topping sub-plant, which is the plant responsible for the primary distillation process. In general, distillates are classified into three categories, depending on their boiling point. As shown in Figure 2, three main processes branch off topping sub-plant. Light fractions, mainly characterized by petrol and liquefied petroleum gas (LPG), are processed through a unifining sub-plant; at this point, LPG can be extracted from the production process, while petrol is dispatched to isomerization or platforming processes. The medium distillate, instead, passes through hydro-desulfurization (HDS) in HDS1 or HDS2 plants. Heavy fractions and distillation residuum are subjected to thermal cracking and visbreaking.

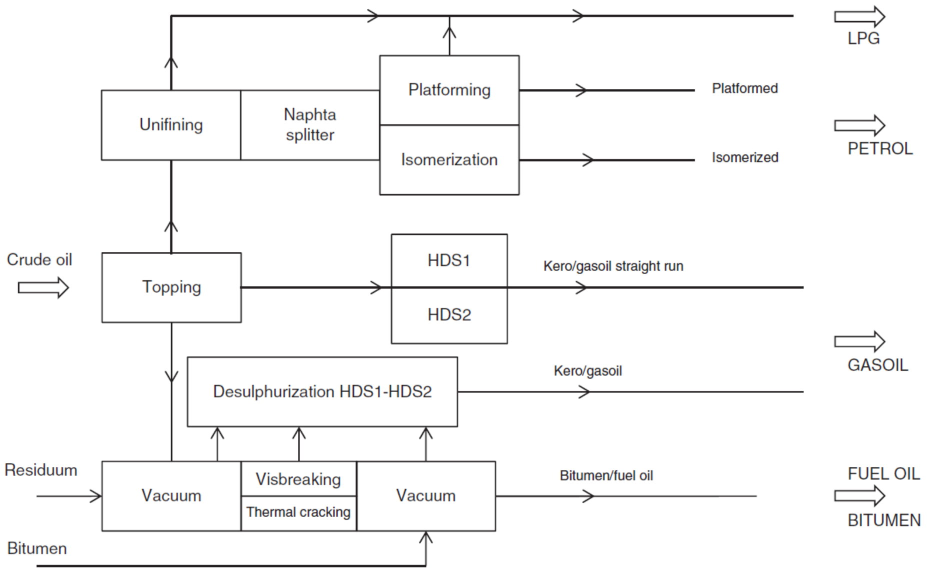

Figure 2 Schematization of the oil refinery production process

Data employed in this study belong to two different databases, containing information about production and maintenance, respectively. A view containing detailed information about the amount of product entered in the process cycle was extracted from the former database. In particular, the mass flow is provided considering hourly ranges for each of the sub-plants. The analyzed timeframe covers three years, from January 2001 to December 2003. Each instance of the view contains daily measurements. An excerpt of the view is reported in Table 3. The columns respectively contain information about the sub-plant, the code of the sensor used to evaluate the measurement (tag), the date of the measurement, the mean values of the mass flow per hour (v01, v02,…, v24), and their sum that represents the daily mass flow. The considered database has some missing values in columns containing the mean values of the hourly mass flow due to sensor measurement errors or sub-plant stoppages. In order to distinguish between the two cases (a and b), the timeframe where the data were missing was compared with the table recording the stops of the sub-plants. If a stoppage was detected, then the missing value was replaced with zero (case b). Otherwise (case a), the mass flow of the previous timeframe has been used to replace the missing value. Stoppages are grouped into three categories:

- shut down if the daily mass flow is null;
- slow down, if the daily mass flow is lower than the 25 percent of the mean daily mass flow of the same year, but not null;
- non-significant (NS) stoppage when the daily mass flow is greater than the 25 percent of the mean daily mass flow of the same year.

On the basis of this classification, a further column reporting the kind of stoppage (see Table 4) was added to the original view. The maintenance database contains data related to work orders due to malfunctioning components or breakdowns.

Table 3 Excerpt of the view extracted from the production database

sub-plant	**tag**	**date**	**v01**	**v02**	**v03**	…	**v24**	**daily mass-flow**
vacuum	FC1401	30/05/2002	0	0	0	…	0	0
vacuum	FC1401	31/05/2002	0	0	0	…	90.193	90.193
vacuum	FC1401	01/06/2002	90.193	18.039	0	…	79.740	187.972
vacuum	FC1401	02/06/2002	79.693	79.693	81.560	…	111.904	352.85

Table 4 Stoppages classification (further column added to Table 3)

stoppage
SHUT_DOWN
SLOW_DOWN
NS
-

A view with information about the malfunctioning component, the identification code of the broken section of the component, the sub-plant where the component is located, and the date of the work order was extracted. An excerpt of the view is reported in Table 5.

Table 5 Excerpt of the view extracted from maintenance database

component	item	sub-plant	date
filtro	P3304B	desulphurization	01/01/2001
tenuta	P2613B	platforming	02/01/2001
pilota	F3101	desulphurization	02/01/2001
filtro	P1846a	visbreaking	03/01/2001

First, the two views have been integrated: when a stoppage of one of the sub-plants was detected, all work orders emitted in the following six months were considered. For example, looking at Table 3, it can be noticed that a stoppage was detected in the vacuum sub-plant on June 2, 2002; hence, all the work orders emitted for malfunctioning components – belonging to any of the sub-plants – from June 2, 2002, to December 2, 2002, were taken into consideration. Table 6 reports an excerpt of the output data.

Table 6 Excerpt of the integration of the views presented in Table 3, Table 4, and Table 5

stopped sub-plant	stoppage date	kind of stoppage	work-order sub-plant	work-order date	component	item
vacuum	01/06/2002	NS	vacuum	03/06/2002	valvola	PSV1421B
vacuum	01/06/2002	NS	desulphurization	03/06/2002	allarme	ZL3328
vacuum	01/06/2002	NS	unifining	03/06/2002	allarme	PHH25165
vacuum	01/06/2002	NS	…	…	…	…
vacuum	01/06/2002	NS	naphta-splitter	29/11/2002	controllore	FC21005
vacuum	01/06/2002	NS	desulphurization	30/11/2002	-	U3200

The next subsections are devoted to present the results of the case study. In particular, vacuum and topping sub-plants are considered.

4.1.2 Data analytics

4.1.2.1 Preliminary analysis

As a preliminary analysis, the possible correlation between a stoppage in a sub-plant and work orders in the whole plant within different time slices has been evaluated. The different intervals considered were a week, a month, two months, and six months. This analysis showed no significant correlations. Hence, the analysis has been refined by considering only work orders emitted for the same sub-plant where a stoppage has been detected. In Table 7, the output of the integration of the two views is reported.

Table 7 Excerpt of the integration of the views presented in Table 3, Table 4, and Table 5, joining on the sub-plant.

sub-plant	stoppage date	stoppage	work order date	component	item
topping	07/01/2001	NS	10/01/2001	cuscinetto	P1004A
topping	07/01/2001	NS	10/01/2001	pilota	U1000
topping	07/01/2001	NS	10/01/2001	analizzatore	F1001
topping	07/01/2001	NS	11/01/2001	serbatoio	P1002B
topping	07/01/2001	NS	11/01/2001	soffiatore	F1101

The current work aims at explaining the relationships between the work orders emitted for some components after a sub-plant stoppage and within a defined time interval. In order to qualitatively evaluate the effectiveness of the proposed approach, three academic experts and six members of the maintenance department of the refinery were interviewed. They considered the approach a valuable opportunity to anticipate the occurrence of components breakdown to improve the operational performances. Moreover, the six members of the maintenance department have been involved in data preprocessing and parameter definition to adapt the methodology to the specific requirements of the oil refinery analyzed.

4.1.2.2 Association Rule Mining

The data set presented in Table 7 is employed as input for the ARs extraction: indeed, through the analysis of the AR, this work aims at identifying the most appropriate maintenance strategy for the components that had frequently required maintenance interventions after sub-plants stoppages. According to step 1 of the procedure proposed, the first parameter to be set was the timeframe: for each stopped sub-plant, members of the maintenance department required to analyze four time intervals: 1 day, 1 week, 2 weeks, and 1 month. In particular, "1 day" explains that the work order is emitted within 24 hours after the stoppage; "1 week" means that the indicated work order occurs within a week after the stoppage; "2 weeks" implies that the work order is emitted within fourteen days after the stoppage; "1 month" tells that work order is emitted within the 31st day.

Members of the maintenance department also required that the work orders were separately analyzed depending on the stoppage classification (NS, slow down and shut down) and decided to set different thresholds of support and confidence for each sub-plant.

In order to extract the ARs, all the components of the sub-plant that required maintenance within the given time interval were identified. An example is reported in Table 8, where the first column identifies the stopped sub-plant, which is the one requiring a maintenance intervention. In contrast, the second column provides the stoppage classification, followed by the date when it occurred and by the time interval. The following 349 columns represent the components constituting the various sub-plants: a value of "True" is assigned if a work order is emitted for the corresponding component during the indicated time interval. Otherwise, the assigned value is "False."

Table 8 Excerpt from the list of the components requiring maintenance interventions within a given time interval after a stoppage.

sub-plant	stoppage	Stoppage date	time interval	accoppiamento	allarme	…	valvola
topping	NS	06/03/2001	1 day	False	False	…	False
topping	NS	06/03/2001	1 month	False	False	…	True
topping	NS	06/03/2001	1 week	True	False	…	False
topping	NS	06/03/2001	2 weeks	False	False	…	True

topping	SHUT_DOWN	08/04/2001	1 day	False	False	…	False
topping	SHUT_DOWN	08/04/2001	1 month	False	False	…	False
topping	SHUT_DOWN	08/04/2001	1 week	False	False	…	False
topping	SHUT_DOWN	08/04/2001	2 weeks	False	False	…	False
topping	SLOW_DOWN	11/06/2001	1 day	False	True	…	True
topping	SLOW_DOWN	11/06/2001	1 month	False	True	…	False
topping	SLOW_DOWN	11/06/2001	1 week	True	False	…	True
topping	SLOW_DOWN	11/06/2001	2 weeks	False	True	…	True

The tool chosen to deploy the analysis is RapidMiner, a widely used DM platform that allows the design of analysis processes by composing predefined tools. According to the structure of the record, the process in RapidMiner was structured as reported in Figure 3. The first operator, Read Excel, has the function of reading the data set in Microsoft Excel format exemplified in Table 8. The second tool, filter time frame, is applied to select only data referring to a specific timeframe. Then, Exclude Attributes performed a vertical selection is performed by Exclude Attributes, with the aim of excluding attributes like Time interval, which would not provide useful information in generating the ARs due to the previous filtering. Nominal to Binominal and Numerical to Binominal tools transform the selected attributes in a Boolean format. Hence, frequent itemsets are mined through the FP-growth operator, and the ARs are extracted (Create AR).

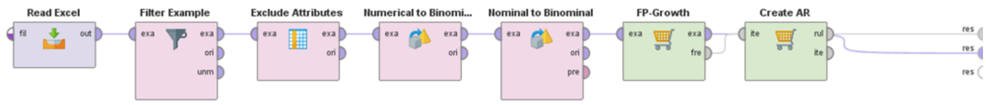

Figure 3 Representation of the process implemented in RapidMiner

Due to the complexity of cost estimation related to production losses, the methodology proposed as well as the following analysis will take into consideration only the components that can be maintained without altering the production process, or, if necessary, recurring to the by-pass systems that assure the ordinary running of the process.

4.1.2.3 Decision support model based on threshold definition

The procedure to define the most appropriate maintenance policy basing on the definition of appropriate thresholds can be summarized as follows:

(1) Define the following parameters:

   a. TF: timeframe, is the time interval, starting from the sub-plant stoppage, during which the analysis is performed;
   b. σ_{rule}: the minimum support requested to a rule to be considered;
   c. σ_{rep}: the minimum value of the support for the execution of predictive maintenance on all the items composing the rule;
   d. σ_{conf}: the minimum value of the confidence for the execution of predictive maintenance on all the items composing the rule.

(2) Mine ARs for the timeframe TF having support greater than or equal to σ_{rule}. For each component c_j of sub-plant j, an item k_j in K is defined such that it assumes a true value if the component is broken because of a stoppage, false otherwise. A transaction t in T represents the set of components broken for the given time interval TF and sub-plant j. FP-growth algorithm is applied over T to obtain the frequent item-set FI. From FI, ARs in the form r: A→B are extracted, having support (r) $\geqslant \sigma_{rule}$.

(3) For each rule r ∈ R, if support(r) $\geqslant \sigma_{rep}$, then a predictive maintenance intervention is performed for all components included in (both head and body of) the rule r.

(4) Monitor and control the sub-plant for the whole timeframe TF:

   a. If component A breaks, let R_A be the set of rules having A as the body of the rule.
   b. For each rule r ∈ R_A, If confidence(r) $\geqslant \sigma_{conf}$, then a predictive maintenance intervention is performed on all components in the head of r as well as a corrective intervention on A.

The definition of parameters (i.e., step 1) is of particular importance, as the analysis can be limited in two ways: first, according to the temporal dimension, in order to relate maintenance interventions only to stoppages in an interval relevant to maintenance policy purposes. The definition of σ_{rule} and σ_{rep} allows limiting the analysis only to critical components. Indeed, in this way, the predictive intervention is performed only for the components that present a statistically significant probability of breakdown. Since an oil refinery is composed of several components, σ_{rule} and σ_{rep} should be chosen to be low enough so that significant rules are not excluded, allowing at the same time to exclude rarely occurring rules.

Instead, the monitoring activity recommended in the last step assures control of the components not maintained before their breakage. Since the confidence of the rule r: A→B represents the probability that the component B breaks if a work order for the component A is emitted, the choice of σ_{conf} affects the decision of predictively maintain or not the component B given the breakdown of A. The definition of the parameter σ_{conf} should take into consideration a twofold aspect. Indeed, if it assumes a value too high, no predictive maintenance will be suggested. On the other hand, if it is set too low, the maintenance policy will suggest performing predictive interventions even on low breakdown probability components. To derive this kind of trade-off, a great experience, as well as the complete knowledge of the process, is necessary.

In order to provide a broader view of the procedure, it will be exemplified through an application to the Topping sub-plant. It is fundamental that the Topping sub-plant works efficiently, as it is situated at the beginning of the production process. The stoppage considered is a slow-down. According to the suggestions provided by the maintenance department members, the following parameters are set for the analysis:

1. TF: 1 month;
2. σ_{rule}= 0.10;
3. σ_{rep}=0.50;
4. $\sigma_{conf} = 0.50$;

From the dataset, 459 frequent item-sets and 678 rules were extracted (see Appendix A for an excerpt) with an execution time shorter than 1 second. 120 of the extracted rules were excluded as they had support lower than σ_{rule}. Considering the remaining 558 rules, it turned out that the support of 10 rules was higher than σ_{rep}. Hence, components of these rules (i.e., *accoppiamento, controllore, coibentazione* and *tenuta*) were immediately substituted. For the 339 unique components contained in the remaining rules, the procedure suggests a monitoring policy.

During the monitoring phase, when a work-order is emitted for a component A, maintenance attendants have to:

1. Perform a corrective intervention to replace A;
2. Check the confidence of all rules having A as the body. If the confidence is greater than the recommended threshold $\sigma_{conf} = 0.50$, then all components in the head of these rules have to be predictively replaced.

Table 9 reports an example of rules having the component *indicatore* in the body. When a work-order is emitted for component *indicatore*, components in the head of the four rules having confidence greater than or equal to 0.50 are considered for maintenance too. In Table 9, the four rules are in bold, and components to predictively repair are *presa campione, rilevatore, illuminazione,* and *amperometro*.

Table 9 Excerpt of the rules extracted for topping sub-plant

Body	Head	Support	Confidence
indicatore	**presa campione**	**0.324**	**0.667**
indicatore	**rilevatore**	**0.324**	**0.667**
indicatore	**illuminazione**	**0.297**	**0.611**
indicatore	**amperometro**	**0.243**	**0.500**
indicatore	allarme	0.216	0.444
indicatore	batteria	0.189	0.389
indicatore	tracciatura	0.162	0.333
indicatore	dreno	0.162	0.333
indicatore	trasmettitore	0.162	0.333
indicatore	strumentazione	0.162	0.333
indicatore	troppo pieno	0.162	0.333
indicatore	lubrificazione	0.162	0.333
indicatore	condensa	0.162	0.333
indicatore	refrigerante	0.135	0.278

indicatore	area	0.108	0.222
indicatore	baderna	0.108	0.222

4.1.3 Control layer

The fourth step of the general framework regards the implementation control of the proposed approach. To this end, sensitivity analyses are carried out to understand how the applicability can be adjusted depending on the outcomes obtained. For example, some rules extracted for Vacuum and Topping sub-plants will be analyzed, varying the timeframe considered for the current case study. In particular, Table 10 shows the variation of support and confidence, after a slow down of the Vacuum sub-plant, of the rule *Tenuta*→*Valvola* when the time interval considered increases: the highest probability of occurrence of both components breakdown can be observed a week after the stoppage. Moreover, if the timeframe is enlarged to two weeks and a month, the probability is lower and lower. On the contrary, confidence increases: if a work-order for component *Tenuta* is emitted on the day after the stoppage, only in 40% of cases will this cause a work-order for component *Valvola*. In all other cases, instead, a work-order for *Tenuta* will be followed by a work-order for *Valvola*.

Table 10 Support and confidence values of the rule *Tenuta* → *Valvola* for different timeframes

Tenuta* → *Valvola	**Support**	**Confidence**
1 day	0.182	0.400
1 week	0.286	1.000
2 weeks	0.125	1.000
1 month	0.118	1.000

Considering the same values of parameters σ_{rule}, σ_{conf} and σ_{conf} proposed for the Topping sub-plant, the couple of components (*Tenuta* and *Valvola*) will not be predictively maintained, as all the support results lower than σ_{rep} =0.50. In case of a work-order for component *Tenuta* after one day, no actions will be performed for *Valvola*. For the other timeframes, instead, component *Valvola* would be immediately replaced, as the confidence is higher than σ_{conf}.

Table 11 Support and confidence values of the rule *Tenuta→Valvola* for different timeframes, in case of NS or shut-down stoppages

Tenuta-->Valvola	Support	Confidence
1 day NS	-	-
1 day shut-down	-	-
1 week NS	-	-
1 week slow-down	0.286	1
1 week shut-down	0.217	0.625
2 weeks NS	-	-
2 weeks shut-down	0.309	0.680
1 month NS	-	-
1 month shut-down	0.375	0.913

Table 11 reports the performances of the rule *Tenuta→Valvola* when NS and shut-down stoppages are considered. It can be noted that, for NS stoppages, the rule is not significant; the two components never break together. Considering shut-down stoppages, support and confidence values grow when the timeframe increases: this means that the probability of occurring in a breakdown of both *Tenuta* and *Valvola* increases.

Table 12 compares the values of support and confidence for the rule *Controllore* → *Soffiatore* varying the kind of stoppage. The rule is significant for all cases related to 1-month time interval, while for a two-week timeframe, it is only meaningful for slow-down stoppages. Furtherly reducing the time interval does not lead to relevant outcomes. Considering "1 month", the category of stoppage influences both support and confidence. In particular, in case of a shut-down, the rule results in being more likely than in case of slow-down or NS stoppages.

Table 12 Support and confidence values of the rule *Controllore →Soffiatore* for different timeframes and different kind of stoppages

Controllore-->Soffiatore	Support	Confidence
1 month	0.212	0.389

1 month NS	0.192	0.385
1 month slow-down	0.176	0.250
1 month shut-down	0.232	0.448
2 weeks	-	-
2 weeks NS	-	-
2 weeks slow-down	0.125	0.333
2 weeks shut-down	-	-

Hence, the monitoring of *Controllore* and *Soffiatore* components does not have to be particularly strict for “1-day” and “1-week” intervals, as well as for “2 weeks” in case of NS and shut-down stoppages. On the contrary, it should be intensified two weeks after a slow-down and one month after all kinds of stoppages – especially for shut-downs.

4.2 Data-driven maintenance policy through a mathematical programming approach

4.2.1 Data gathering and management

The proposed approach described in this case study is applied to a real-life oil refinery, characterized by a production capacity of 85,000 barrels/day. The refinery plant is organized into sub-plants, each devoted to specific activities. In particular, the *topping* sub-plant receives crude oil in input and, then, the production process is split into three branches:

(a) The first one is dedicated to liquefied petroleum gas and petrol production. Hence, the corresponding subplants are dedicated to *unifining, naphtha splitting, isomerization*, and *platforming*.

(b) The second branch produces gas oil by means of the *hydro-desulfurization* sub-plant.

(c) Instead, the third one is composed of *thermal cracking, visbreaking*, and *hydro-desulfurization* subplants for the production of fuel oil and bitumen.

The data provided by the refinery plant’s maintenance department refer to the period from January 2001 to December 2003, and they are organized in two different databases. The former is referred to as the crude oil circulating in the sub-plant. It contains the average hourly mass-flow, the daily mass-flow (obtained by adding up the hourly measurements), and the average yearly value, calculated from the daily measurements. This database has some missing values in the columns reporting hourly mass-flow that could depend on a blockage or a measurement error. In order to replace missing values, we compare instances of the database with the list of occurred blockages, as follows:

(a) If a blockage is detected, then the missing value is replaced by 0.
(b) Otherwise, the missing value is due to a measurement error. Hence, it is replaced by the value of the hourly mass-flow measured at the previous hour.

The refinery classifies the blockages into three groups:

(1) A shut-down (ShD) is defined as an all-day blockage. Hence, the mass-flow value remains null for the whole day observed.
(2) A slow-down (SlD) blockage causes a decrease of the daily mass-flow less than 25% of the mean.
(3) All the others are classified as non-significant (NS).

In the case of a sub-plant blockage, the corresponding category is stored in the database. The other database collects information regarding the maintenance activities. In particular, it stores information about the component and the date in which the maintenance has been performed for each activity. The maintenance date is equal to or later than one of the component's breakage. In this work, we assume that it is precisely equal to the date on which the component's breakage occurs. During the monitored period, several blockages occurred: 21 NS blockages (103 h), 37 SlD blockages (122 h), and 8 days of ShD (192 h). Moreover, 767 components required maintenance activities. The two databases have been integrated adequately by joining data regarding the sub-plant blockages, and the components' breakage occurred after the blockage in a defined time interval. Table 13 shows an example of the integrated database where the first two columns report the date in which the blockage occurs (Blockage Date) and its category, respectively. The remaining columns refer to the maintenance activities performed on a given component. These data will be used as the basis for the analytics step.

Table 13 Excerpt of the integration between the two databases.

Blockage date	Blockage category	Intervention date	Component	Item
01/06/2002	NS	03/06/2002	valve	PSV1421B
01/06/2002	NS	03/06/2002	alarm	ZL3328
01/06/2002	NS	03/06/2002	indicator	PHH25165
01/06/2002	NS	…	…	…
01/06/2002	NS	29/06/2002	controller	FC21005

4.2.2 Data analytics

4.2.2.1 Preliminary analysis

According to the general framework procedure, in order to extract the ARs, data reported in Table 13 are re-arranged as presented in Table 14. In this way, they are suitable to be processed on the platform chosen for the ARM process.

Table 14 Excerpt of the input dataset for RapidMiner.

Blockage date	Blockage category	Time interval	Coupling	Alarm	…	Impeller
06/03/2001	NS	1 month	False	False	…	True
06/03/2001	NS	1 week	True	False	…	True
06/03/2001	NS	2 weeks	True	False	…	True
08/04/2001	SHUT-DOWN	1 month	True	False	…	False
08/04/2001	SHUT-DOWN	1 week	False	False	…	False
08/04/2001	SHUT-DOWN	2 weeks	False	False	…	False
11/06/2001	SLOW-DOWN	1 month	False	True	…	False
11/06/2001	SLOW-DOWN	1 week	True	False	…	True
11/06/2001	SLOW-DOWN	2 weeks	False	True	…	True

The first three columns report the date of each blockage, its category, and the considered time interval (ΔT). The following 82 columns contain a list of the components belonging to the *topping* sub-plant. If the corresponding component required a maintenance activity in the considered time interval, then the value assigned is *true*, *false* otherwise. For example, the blockage that occurred on April 8, 2001, is an SlD: a maintenance activity is performed on the component *coupling* in a 1-month time interval. The *alarm* and *impeller* are not maintained after this blockage.

Table 15 summarizes, for each sub-plant, the number of components monitored and the percentage of components broken during the period under investigation. Indeed, the numbers reported in the table confirm the need for a maintenance policy. In addition, the high percentage of component breakages implies a high cost due to the reduced production capacity of the sub-plant. And, this implicitly confirms the need to implement a predictive maintenance policy. Since the three production processes depend on the *topping* sub-plant, the maintenance policy is applied to its components. Indeed, for the proper functioning of the whole refinery plant, the flow along this sub-plant must run smoothly.

Table 15 Resume of the sub-plants, the corresponding number of components monitored in each of them, and the percentage of components requiring a maintenance intervention in the monitored period

Sub-plant	Number of components	Percentage of broken components
Topping	82	88%
Unifining	73	86%
Naphta splitting	23	52%
Isomerization	37	84%
Platforming	91	70%
Hydro-desulfurization (1)	59	75%
Thermal cracking	35	88%
Visbreaking	86	79%
Hydro-desulfurization (2)	44	63%

4.2.2.2 Association Rule Mining

Considering the information extractable from the available data, when a component breakage occurs, the ARs having the broken component as their body are extracted. To this end, the tools provided by RapidMiner (www.rapidminer.com), a widely applied data-mining platform, are used.

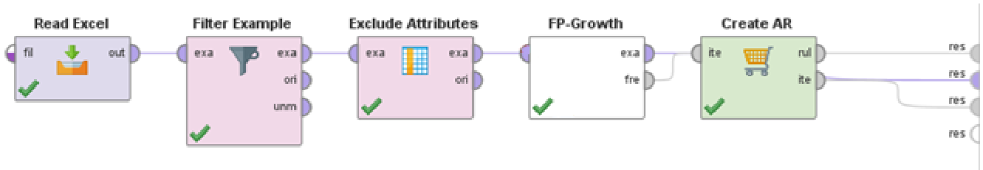

Figure 4 View of the process implemented in RapidMiner

In particular, Figure 4 describes the whole process. Firstly, the integrated dataset (as represented in Table 14) is loaded from Microsoft Excel; the operator *filter example* allows setting some filters, e.g., limiting the analysis to a specific blockage. Then, through the *exclude attributes* module, attributes that do not provide useful information are excluded from the analysis. *FP-growth* and *create AR* generate the frequent patterns and the ARs from the dataset, respectively.

4.2.2.3 Decision support model based on mathematical programming approach

Defining a mathematical model to define whether a predictive intervention is necessary can support the decision-maker. In this way, indeed, there is no need to define a-priori thresholds, thus avoiding any possible subjective bias by the process experts. In this section, we describe the solution approach proposed for defining a new predictive optimization-based maintenance policy.

The predictive maintenance policy proposed is mainly based on the integration of ARs mining and optimization techniques. Indeed, the approach developed in this section is aimed to define the optimal maintenance plan for a set of components in a plant. In fact, in large process industries, the plant production capacity is often affected by the blockages that can occur and that can be caused by several causes, e.g., scheduled interruptions, safety issues, or component breakage. Given the plant complexity, after repairing a blockage and restarting the operation, some other components may fail due to the changes of the working conditions (e.g., from a full load operation, the system switches to a blockage, then to a transient phase before switching back to full load operation). Furthermore, in industry, a production plant is often implemented through various activities performed by specific sub-plants. Without loss of generality, hereafter, the focus is on a sub-plant at a time. In fact, this solution approach is based on the idea of individuating correlations between sub-plant blockages and subsequent components' breakages. Therefore, on the basis of available historical data, we aim at discovering the correlations among components' breakages after a sub-plant blockage (using AR mining) in a given time interval. These rules can be applied for determining the components that can be predictively maintained given a component's breakage. The decision on which components have to be selected for predictive maintenance depends on both the time available for maintenance planning and their repair cost. To this purpose, a mathematical model is formulated to select the optimal set of components to predictively maintain under time and budget constraints, maximizing the overall plant reliability (i.e., minimizing the probability of future breakages).

An ILP model is formulated for defining the optimal maintenance plan for a given set of components. In particular, it aims at selecting the components with the highest breakage probability given that the breakage of a component occurs. The notation and the assumptions used throughout the paper are given in the following and summarized in Table 16.

Table 16 Nomenclature of the input data for the ILP model

Parameter	Meaning
C	Set of components
c_{ij}	Confidence of the rule i → j (breakage probability of component j given the breakage of component i)
RC_j	Repair cost of component j
T_j	Repair time of component j
T_{max}	Maximum time allowed for maintenance planning
B	Maximum budget allowed for maintenance planning

C is the set of components belonging to a plant under analysis. Each component j is characterized by a repair cost RC_j and a repair time T_j , i.e., the duration of the maintenance activity, expressed in minutes. It is worth noting that, for each component, its repair cost also takes into account every cost due to its breakage. Moreover, each component j is characterized by the confidence c_{ij} = conf (i → j) that expresses its breakage probability, given the breakage of component i, i.e., c_{ij} = P(j|i). The ILP formulation is modeled by introducing the binary decision variable x_j, equal to 1 if the component j is selected to be maintained, 0 otherwise. The maintenance planning is then optimized by solving the following ILP model:

$$\max \sum_{j \in C} c_{ij} x_j \tag{1}$$

$$\sum_{j \in C} T_j \, x_j \leq \alpha \, T_{max} \tag{2}$$

$$\sum_{j \in C} RC_j \, x_j \leq B \tag{3}$$

$$x_j \in \{0, 1\} \; \forall \, j \in C \tag{4}$$

The objective function (1), to maximize, represents the total confidence. Constraint (2) assures that the total repair time, required for all selected components, does not exceed a percentage (α) of the maximum time allowed for maintenance planning (Tmax). In such a constraint, the parameter α can be appropriately modified for scenario analysis. Constraint (3) imposes a maximum budget B that can be used for maintenance. Finally, constraints (4) provide the variable nature.

The predictive optimization-based maintenance policy consists of the steps outlined in the following, and it is briefly schematized in Figure 5.

INPUTS

- The set of components C of the plant;
- The time interval (ΔT), starting from the plant blockage, during which the analysis is performed;
- The minimum support threshold (min_sup) for ARs' extraction.

PROCEDURE

1. Find the set *R* of all ARs having a support greater than min_sup, where body and head are formed by the components broken during ΔT after past blockages.
2. Monitor the plant operations within ΔT .

(a) When a maintenance activity is required for the component $i \in C$, select all $r_{ij} \in R : i \rightarrow j$, where $j \in C$, $i = j$.

(b) Solve the ILP model described in (1)-(4) for selecting the components to be maintained on the basis of the information extracted at the previous step.

OUTPUTS

- The optimal set of components to maintain;
- The total time for maintenance planning.

It is worth noting that defining the input parameters is particularly significant in the above procedure. Indeed, the time frame has to be set so that the maintenance activities are related to plant blockages in a meaningful way. In fact, setting a too short interval could lead to the loss of relevant associations, i.e., not to consider all the component breakages related to the specific blockage. On the contrary, a time interval too long may provide misleading results. Hence, this step of the procedure has to be carried out by domain experts, able to both define the most appropriate length of the time interval and evaluate whether shortening or enlarging the time for maintenance may result particularly convenient. The number of rules also depends on min_sup. In our scenario, the min_sup threshold has to be set as low as possible in order to allow analyzing a significant number of ARs. Starting from the plant blockage, the system is monitored, and, in the case of breakage (step 2.a), the maintenance planning is defined by solving an optimization model (step 2.b). This aspect overcomes what was already proposed in section 4.2 in which a maintenance policy, based on user-defined minimum confidence, is proposed. In our approach, the solution of an ad hoc defined optimization model allows selecting the most convenient components to be maintained in a predictive way by completely removing the arbitrariness introduced by the user-defined confidence threshold.

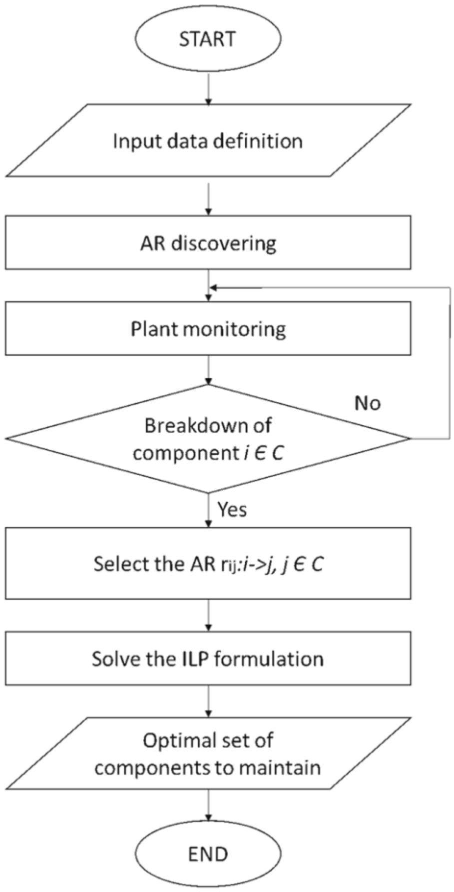

Figure 5 Flow chart schematizing the ARs-based Optimization approach

The implementation of the RapidMiner process has been run on a machine at 3.40 GHz with 16 GB of RAM. It requires 28 s to extract the full set of ARs, namely for ΔT equals to 1 week, 2 weeks, and 1 month. The ILP model has been implemented in LINGO language (www.lindo.com) and runs on the same machine. Solving the ILP model, formulated in Section 3.2, requires 0.8 s[2]. For the sake of clarity, the whole ARs extraction

[2] The time for extracting ARs and that for solving the ILP model are averaged on 5 runs.

4.2.2.4 Methodology validation

This section aims at validating the proposed methodology by considering a use case with 21 components. It is assumed that the breakage of the component $\bar{C}$ happens. The goal of the proposed methodology is to decide which components (hereafter, denoted as *Ci* $\forall i = 1, \ldots,$ 20) we have to repair in a predictive way while the plant is stopped to repair $\bar{C}$. The repair time *Ti* and the repair cost *RCi*, $\forall i = 1, \ldots,$ 20 have been randomly generated in the range [30*,* 300] and [100*,* 3000], respectively. Here, we are also assuming that ΔT equals to 1 month. An a priori breakage probability is associated with each component *Ci*, randomly generated in the range [0*,* 0.6]. Based on this probability, it is possible to determine if, in the month in which the breakage of $\bar{C}$ occurs, the component *Ci* breaks too, and a repair order is then issued. Hence, 56 months have been simulated. In particular, 36 months have been used for generating the ARs, while the remaining 20 months for testing the methodology (each denoted as *Testing Month TMi* , $\forall i = 1, \ldots,$ 20). By following the proposed methodology, after obtaining the confidence of each of the 20 rules of the type $\bar{C}$ → *Ci*, the ILP model, is then solved by setting *T*max and *B* equal to 350 and 10,000, respectively. For each testing month *TMi*, the square *confusion matrix CMi* of order 2 has been defined as follows:

$$CMi = \begin{bmatrix} RRi & RNi \\ NRi & NNi \end{bmatrix}$$

where:

1. *RRi* denotes the number of components to be repaired in *TMi* and actually selected by the ILP model;
2. *RNi* is the number of components to be repaired in*TMi* but not selected by the ILP model;
3. *NRi* represents the number of components not to be repaired in *TMi* but selected by the ILP model;
4. *NNi* counts the number of components not to be repaired in *TMi* and actually not selected by the ILP model.

Then, for each TMi , the accuracy ηi has been calculated as:

$$\eta i = \frac{NNi + RRi}{NNi + RRi + NRi + RNi}$$

Then, the average accuracy has been computed over the 20 testing months. We have run 10 simulations (varying Ti , RCi , and a priori breakage probability), obtaining a high average accuracy $\bar{\eta}$ equal to 0.836 with a variance of 0.078, proving the effectiveness of the proposed predictive methodology. It is worth noting that errors (i.e., RNi and NRi) depend on the imposed constraints on the total repair time and the total available budget. On the 10 simulation runs, the average total repair time, as well as the average total budget required, was 321 and 3646.3, respectively. Moreover, a critical issue of the proposed methodology is the availability of a large amount of data. Indeed, the quality of results depends on the extraction of valid ARs, i.e., rules whose confidence represents a reasonable estimation of the actual breakage probability, given that the breakage of the component $\bar{C}$ occurs.

4.2.2.5 Implementation of the proposed approach

This section describes the numerical results obtained by applying the solution approach detailed in the previous sections. In the following experiments, the attention is focused on the component requiring the highest number of maintenance activities, i.e., the controller. In order to compare and discuss results, four different cases are presented, considering all the blockage categories and differentiating among SlD, ShD, and NS blockages. According to the privacy policy adopted by the refinery, the details about the total budget, the repair times, and the costs of the components cannot be fully reported. However, in the following experiments, reasonably estimated values are used for them.

The first example presented regards the breakage of the controller since it resulted in the most critical component in terms of number of maintenance activities required. Indeed, from data, it turns out that the controller is the component with the highest breakage probability (87.9%). The parameters setting is performed by following the suggestion coming from the maintenance department: the set C is made up of 82 components, monitored in the topping sub-plant while the value of ΔT and min_sup are equal to 1 month and 0.005, respectively. The budget value is set to 10,000 €, while the maximum time Tmax is 350 min. Finally, α is initially set to 1. Firstly, all the ARs of interest are individuated as described in the previous section. Then, the monitoring phase starts.

When a maintenance activity is required for the component controller, all the ARs whose support is greater than min_sup and controller as body are selected. In Table 17, we report the ARs extracted for analysis. In particular, the first column shows the body of the rule, namely controller, while the second one the head of each rule. Then, in the third column, the confidence of the rules is indicated. The last two columns report the repair cost and the repair time of the component in the head of the rule. According to these rules, the components with the higher probability of breakage given the failure of the controller, i.e., confidence of the rule, are coupling, sealing device, and insulation. The solution of the optimization model, instead, highlights that when the breakage of the component controller occurs, a consequent maintenance activity should be planned for the components ammeter, drainer, lighting, liquid level, and piping (all highlighted in italics in Table 17), so that both the total repair time and the budget constraints can be respected. In this way, it can be obtained total confidence of 1.397. Indeed, the repair times estimated for the selected components are 120, 90, 10, 60, and 60, respectively. This means that 340 minutes of the 350 available are used. Moreover, the total repair cost of the selected components is 2295 €, out of the 10,000 € of the total budget. One can argue that a more straightforward way for detecting the most convenient set of components to maintain is to order them by decreasing confidence and, then, to select starting from the most likely ones, i.e., those with the highest confidence, respecting the time and budget constraints.

Table 17 Association rules having support greater than min_sup and *controller* as body.

Body	Head	Confidence	RCHead	THead
Controller	Coupling	0.690	1184	250
Controller	Sealing device	0.569	5931	600
Controller	Insulation	0.517	2300	750
Controller	*Lighting*	0.466	90	10
Controller	Tracker	0.431	235	430
Controller	Indicator	0.414	289	300
Controller	Alarm	0.362	4094	275
Controller	Area	0.328	2881	324
Controller	Sampling valve	0.310	150	300

Controller	*Ammeter*	0.259	1009	120
Controller	*Drainer*	0.224	845	90
Controller	Valve	0.224	735	300
Controller	*Liquid level*	0.224	190	60
Controller	Scanner	0.224	2100	150
Controller	*Piping*	0.224	161	60
Controller	Bearing	0.207	2500	800
Controller	Auxiliary	0.172	1010	206
Controller	Air analysis system	0.155	2103	170
Controller	Blade	0.155	1233	607
Controller	Condensation detector	0.138	207	420
Controller	Transmitting device	0.138	890	254
Controller	Lubrication	0.138	580	402
Controller	Dimmer	0.138	2930	293
Controller	Refrigerant	0.138	3290	248
Controller	Oil seal	0.138	402	300
Controller	Engine	0.138	4065	348
Controller	Electrode	0.138	5040	122
Controller	Instrumentation	0.121	1300	280
Controller	Button panel	0.121	1600	400
Controller	Pavage	0.121	2065	200
Controller	Level controller	0.121	2300	60
Controller	Battery	0.121	1280	177

In this way, the components coupling, lighting, and liquid level are selected for maintenance, with total confidence equal to 1.379. The total time required for performing this maintenance plan is 320 min, with a total repair cost of about 1500 €. Despite both a time and cost-saving, this solution provides total confidence (1.379) lower than the one detected by ILP (1.397). A more accurate perspective can be obtained if the rules are discriminated on the basis of the blockage category since it can impact the component breakages. For instance, Table 18 contains the ARs related to an SlD blockage.

Table 18 ARs extracted in the case of SID blockage having support greater than min_sup and *controller* as body

Body	Head	Confidence
Controller	Coupling	0.886
Controller	Sealing device	0.657
Controller	Insulation	0.657
Controller	Indicator	0.514
Controller	Tracker	0.457
Controller	Lighting	0.457
Controller	Sampling area	0.429
Controller	Ammeter	0.400
Controller	Alarm	0.400
Controller	Scanner	0.371
Controller	Area	0.343
Controller	Drainer	0.314
Controller	Liquid level	0.286
Controller	Auxiliary	0.286
Controller	Blade	0.257
Controller	Bearing	0.257
Controller	Air analysis system	0.229
Controller	Oil seal	0.229
Controller	Liquid level	0.224
Controller	Valve	0.200
Controller	Button panel	0.200
Controller	Dimmer	0.200
Controller	Engine	0.200
Controller	Electrode	0.200
Controller	Battery	0.200
Controller	Transmitter	0.171
Controller	Equipment	0.171
Controller	Paving	0.171
Controller	Lubrication	0.171

Controller	Level controller	0.171
Controller	Condensation detector	0.171
Controller	Refrigerant	0.143
Controller	Belt	0.143

Comparing Table 18 and Table 17, it is noteworthy that in both cases, the rules are almost all the same, but with different values of confidence. This is a reasonable result since the SlD blockages are the majority. The only exception is the component Belt, whose support is higher than the min_sup only in the case of an SlD.

When an ShD is considered (see Table 19), the number of ARs decreases, and they involve some new components, like safety valve, pressure gauge, and piston. The repairing of these components would be preferable since an ShD blockage has the highest impact on production. However, this kind of blockage is the rarest, so the related rules have a low significance.

Table 19 ARs extracted in the case of ShD blockage having support greater than min_sup and controller as body

Body	Head	Confidence
Controller	Insulation	0.714
Controller	Tracker	0.571
Controller	Lighting	0.571
Controller	Sealing device	0.429
Controller	Indicator	0.429
Controller	Alarm	0.429
Controller	Coupling	0.429
Controller	Safety valve	0.286
Controller	Valve	0.286
Controller	Blower	0.286
Controller	Refrigerant	0.286
Controller	Piston	0.286
Controller	Pressure gauge	0.286
Controller	Liquid level	0.286

Controller	Joint	0.286
Controller	Tube bundle	0.286

4.2.3 Control layer

In order to identify how a modification of the parameters may impact on the proposed implementation, a sensitivity analysis is carried out and described in the following sections.

4.2.3.1 Sensitivity analysis on α parameter

Scenario analysis to study the sensitivity of the solution varying the α parameter is carried out. In particular, a range is defined, between 0.50 and 1.50, and different cases are tested using an incremental step of 0.05. In Figure 6, the values of the objective function (1) are reported as the parameter α increases and all kinds of blockages are considered.

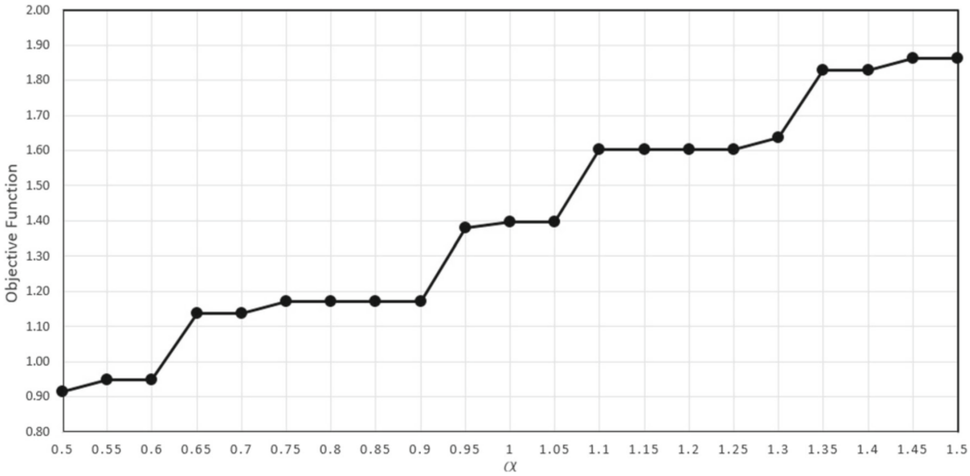

Figure 6 Values of the objective function (1) for different *α*

This figure shows the trend of the objective function with respect to the portion of the maximum repair time used. Reducing the time available for maintenance planning has obviously a significant impact on the number of components that can be maintained. Indeed, when $\alpha = 0.50$ (αTmax = 175 min), *piping*, *liquid level*, and *lighting* are selected for maintenance planning, but the total confidence decreases by about 53% (0.914). On the contrary, increasing the available time of the 50% ($\alpha = 1.50$, αTmax = 525 min) leads to total confidence of 1.862, with 25% growth. In this case, the selected components are *coupling*, *ammeter*, *lighting*, *liquid level*, and *piping*. It is worth noting that the components with high confidence, i.e., *sealing device* and *insulation* (see Table 18), have not been selected since they violate the total repair time constraint. In Table 20, for each scenario, the corresponding α, the total repair time (TRT) of the selected components are reported. Instead, the third column shows the value of the objective function (i.e., the total confidence (TC)) while the last one details the selected components. This way, the decision-maker can evaluate, on the basis of her own experience, how to properly choose the α value and how much she/he is willing to pay for increasing the total time available for maintenance.

Table 20 Optimal solution displayed for the α parameters analyzed

α	TRT	TC	Selected components
0.5	130	0.914	Lighting, liquid level, piping
0.55	190	0.948	Ammeter, lighting, liquid level
0.6	190	0.948	Ammeter, lighting, liquid level
0.65	220	1.138	Lighting, liquid level, piping, drainer
0.7	220	1.138	Lighting, liquid level, piping, drainer
0.75	250	1.172	Lighting, liquid level, piping, ammeter
0.8	250	1.172	Lighting, liquid level, piping, ammeter
0.85	250	1.172	Lighting, liquid level, piping, ammeter
0.9	250	1.172	Lighting, liquid level, piping, ammeter
0.95	320	1.379	Coupling, lighting, piping
1	340	1.397	Ammeter, drainer, lighting, liquid level, piping
1.05	340	1.397	Ammeter, drainer, lighting, liquid level, piping
1.1	380	1.603	Coupling, lighting, liquid level, piping
1.15	380	1.603	Coupling, lighting, liquid level, piping
1.2	380	1.603	Coupling, lighting, liquid level, piping
1.25	380	1.603	Coupling, lighting, liquid level, piping
1.3	440	1.638	Coupling, ammeter, lighting, piping
1.35	470	1.828	Coupling, drainer, lighting, liquid level, piping
1.4	470	1.828	Coupling, drainer, lighting, liquid level, piping
1.45	500	1.862	Coupling, ammeter, lighting, liquid level, piping

1.5	500	1.862	Coupling, ammeter, lighting, liquid level, piping

4.2.3.2 Sensitivity analysis on the budget

An additional sensitivity analysis is presented, varying the budget allocated to maintenance activities. The different values tested range from 500 € to 30,000 €, with an increment of 500 €. The greater the budget, the higher the total confidence obtained. This is due to the fact that more components can be repaired in the maximum time allowed. For example, if B is set to 500 €, the components selected for maintenance are *lighting*, *liquid level*, and *piping*. The total confidence obtained in this case is 0.914. The same solution is obtained in the case in which B is set to 1,000 €. If B ranges from 1,500 € to 2,000 €, instead, the total confidence is higher (1.379) and the components selected are *coupling*, *lighting*, and *piping*. Remarkably, allowing a budget higher than 2,500 € is not useful since the optimal solution found remains the same: *ammeter*, *drainer*, *lighting*, *liquid level*, and *piping* are the selected components, while the total confidence is 1.397. Indeed, above this value, constraint (2) becomes tighter than constraint (3), making any variation on the budget irrelevant.

4.2.3.3 Variations of the blockage category

In order to further detail the experimental campaign, in this section, the analysis is performed both distinguishing the blockage category (i.e., NS, SlD, and ShD) and varying the α parameter. Indeed, the dataset is properly filtered in order to extract only the ARs related to each blockage category and consider the corresponding confidence values to solve the model. Figure 7 shows the trends of the objective function (1) with respect to the portion of the maintenance time used. Observing the results reported in the figure, it is worth noting that in the case of an SlD blockage, the optimization model provides the highest total confidence. When ShD and NS blockages are considered, the values of the objective function are lower than the values obtained in the case of SlD blockages. Indeed, ShD blockages rarely occur, and after them, the number of components that have broken within ΔT is less than the one after SlD blockages. A further consequence of this is obtaining ARs with very quantized confidence values (see Table 19). This leads to the piecewise linear trend of the objective function in case of ShD blockages (see Figure 7). A similar trend is also reported in the case of NS blockages, but the reasons are different. It is noteworthy that the decrease of the daily mass-flow due to NS blockages is not significant, and its impact on components' breakage is limited too. Indeed, most of ARs involve only a component (e.g., the *controller*), and there are very few rules involving two components within ΔT after an NS blockage. Hence, these rules are characterized by very low confidence values. In particular, in case of ShD, when α ranges from 0.5 to 0.7, the components selected for maintenance planning are *lighting* and *liquid level* (TC = 0.857). When α varies from 0.8 to 0.9, the component *pressure gauge* is also selected, and the value of the objective function is 1.143. The components *coupling*, *lighting*, and *liquid level* are selected for any α in the range from 0.95 to 1.40 (TC = 1.286). It is worth noting that the rule with *insulation* as head has confidence by far higher (i.e., 0.714) than the ones mentioned above, but its repair time exceeds Tmax. Thus, increasing the repair time by 40% does not lead to any improvement on total confidence.

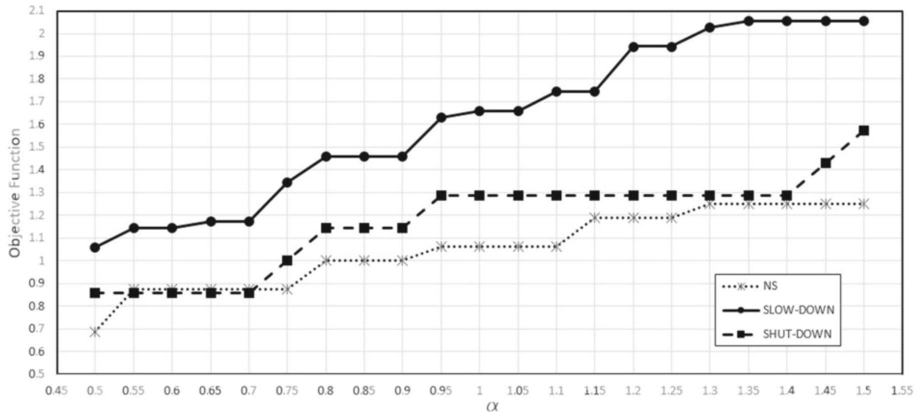

Figure 7 Comparison between objective function values (1) for different α, discriminating the blockage category

4.3 Data-driven maintenance policy based on multi-objective optimization

The implementation of the third approach proposed in this thesis is slightly different from the ones presented in the previous sections. Firstly, because two different solution approaches are compared (i.e., the lexicographic optimization and the Large Neighborhood

Search heuristics); secondly, the breakage probability estimation does not use the association rule mining: instead, a proper algorithm is developed, starting from the same hypothesis as the ARM.

4.3.1 Data gathering and management

The experimental campaign was carried out on a set of real-lifelike instances inspired by an oil refinery. To this end, a three-year time interval, from January 2001 to December 2003, was analyzed. Data came from two databases: one containing information on the process cycle, recording the hourly amount of product entering each sub-plant and, if any, the stoppage detail; the other storing information on both component breakages and the related maintenance interventions. Such a refinery is characterized by a total number of 715 monitorable components. From data, it turns out that, in the period considered, the number of component breakages occurring was 6160, and the total number of stoppages was 1164. The distributions of both repair costs and times of the components are provided in Figure 8 and Figure 9. In particular, the repair time distribution is Gaussian-like, while one can note that most of the components have a low cost.

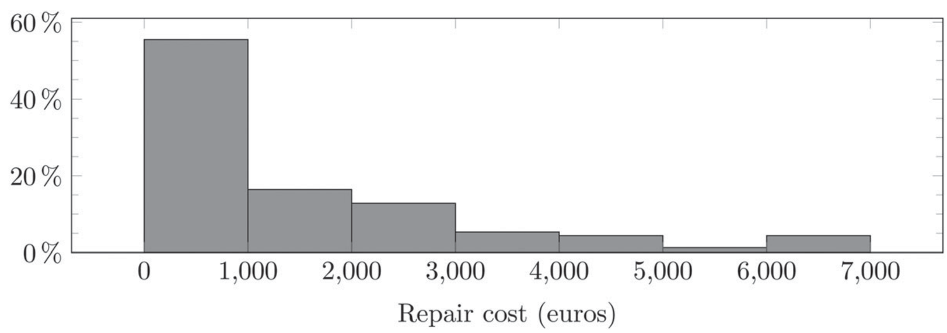

Figure 8 Distribution of component repair cost.

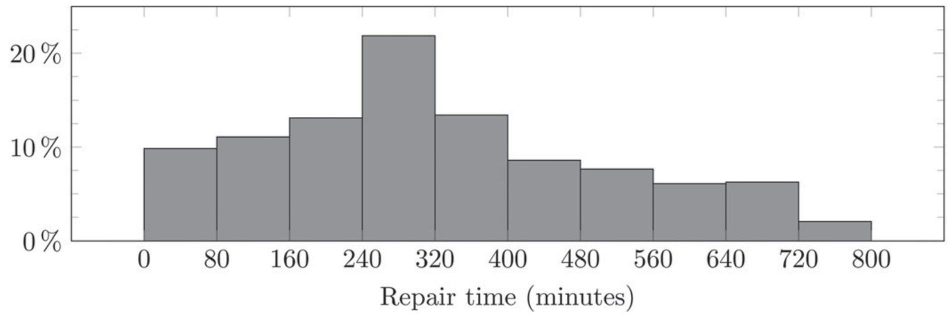

Figure 9 Distribution of component repair time.

4.3.2 Data analytics

4.3.2.1 Preliminary analysis

Since this application is based on the development of two different approaches to be compared, it is necessary to define the same notation for them.

The bi-objective Component Repair Problem (b-CRP) aims at finding the optimal set of components to repair in order simultaneously to maximize the overall system reliability and minimize the maximum repair time required, under constraints on both the total budget B and the total repair time T_{max}.

The set C contains the components from which the optimal set has to be selected. For each $c_j \in C$, a repair cost rc_{c_j}, a repair time rt_{c_j} and the number of maintainers n_{c_j} required for repairing it are given. The parameter C_{work} denotes the fixed hourly cost of employing a maintainer, whereas bp_{c_j} is the breakage probability of the j-th component in a given time interval ΔT of observation after a system stoppage. The notation adopted is also summarized in Table 21.

Table 21 Notation used

Sets	
C	Set of components
E_s	Ordered set of stoppage events
E_b	Ordered set of component breakage events
Parameters	
bp_{c_j}	Breakage probability of component c_j

rc_{c_j}	Repair cost of component c_j
rt_{c_j}	Repair time of component c_j
n_{c_j}	Numbers of operators required for repairing component c_j
ΔL_{c_j}	Lifespan of component c_j
$\overline{\Delta F}_{c_j}$	Mean time between failures of component c_j
T_{max}	Maximum time allowed for maintenance planning
B	Maximum budget allowed for maintenance planning
C_{work}	Fixed hourly cost for employing an operator
ΔT	Time interval of observation

4.3.2.2 Breakage probability estimation

The procedure followed for estimating the breakage probability bp_{c_j} of each $c_j \in C$ is detailed in Algorithm 1.

Algorithm 1 The data-driven optimization-based approach

Input:

Set of components C;

Component's life span ΔL_{c_j} and component MTBF $\overline{\Delta F}_{c_j}, \forall\, c_j \in C$;

Time interval ΔT;

Breakage probability bp_{c_j} $\forall\, c_j \in C$;

List of stoppage events $E_S = \{\langle t'_{S_1}, t''_{S_1}\rangle,\ \langle t'_{S_2}, t''_{S_2}\rangle,\ \ldots,\ \langle t'_{S_m}, t''_{S_m}\rangle\}$, where:

t'_{S_i} is the timestamp when the stoppage S_i occurs;

t''_{S_i} is the timestamp when the plant restarts after a stoppage;

such that $t'_{S_i} < t''_{S_i}, i = 1, \ldots, m$ and $t''_{S_i} < t'_{S_{i+1}}, i = 1, \ldots, m-1$;

List of component breakage events $E_b = \{\langle \gamma_1, t_{\gamma_1}\rangle, \langle \gamma_2, t_{\gamma_2}\rangle, \ldots, \langle \gamma_k, t_{\gamma_k}\rangle\}$, where

t_{γ_σ} is the timestamp which the component $\gamma_\sigma \in C$ breakage occurs at, and

$t_{\gamma_j} < t_{\gamma_{j+1}}, j = 1, \ldots, k-1$.

Output: Set of components to repair $\overline{C} \subseteq C$

```
bp_{c_j} := 0, ∀ c_j ∈ C;
i := 1;
t_start = t''_{S_i};
t_end = t'_{S_{i+1}};
for σ := 1 to k do
   if t_{γ_σ} > t_start then
      if (t_{γ_σ} < t_end ∧ t_{γ_σ} ≤ (t_start + ΔT)) then
         bp_{γ_σ} := bp_{γ_σ} + 1/m
      else
         if i < m − 1 then
```

11: $i := i + 1$;
12: $t_{start} = t''_{S_i}$;
13: $t_{end} = t'_{S_{i+1}}$;
14: **end if**
15: **end if**
16: **end if**
17: **end for**
18: **for** $j := 1$ to $|C|$ **do**
19: **if** $\Delta L_{c_j} < \overline{\Delta F}_{c_j}$ **then**
20: $bp_{c_j} = \left(1 - \frac{\overline{\Delta F}_{c_j} - \Delta L_{c_j}}{\overline{\Delta F}_{c_j}}\right) bp_{c_j}$
21: **end if**
22: **end for**
23: $\overline{C}$:=**selectBestComponents**(C, T_{max}, B, bp_{c_j}, $\forall\, c_j \in C$)

Since the phenomenon under investigation is binary (component failure/working), the algorithm implements a maximum likelihood estimation to derive the set of bp_{c_j}. This approach returns good probability estimations when working conditions do not change considerably over time, and a lot of data on past breakages is available. These two assumptions hold in the scenarios considered in this work: complex and constantly active process industries (e.g., oil refineries, steel mills, highly automated plants).

The first part of the algorithm concerns data pre-processing aimed at estimating the set of breakage probabilities bp_{c_j} from input data. To this end, the algorithm combines information regarding component breakages and system stoppages that have occurred in the past. In detail, the algorithm scrolls the list E_b of k component breakage events (step 5), computing how many times each component breaks within a time window of length ΔT starting from the restart of the system after a stoppage (steps 7–9). If a breakage event falls outside the time window, the next stoppage is used to define a new time window (steps 10–14). In order to take into account the lifespan of the component c_j, namely the time since the last replacement or repair of c_j, its breakage probability is adjusted based on the Mean Time Between Failures (MTBF) of c_j. If the component has recently been repaired, i.e., the component lifespan is less than a given percentage of its MTBF (i.e. $\overline{\Delta F}_{c_j}$), then the breakage probability will be decreased in proportion to the difference between its lifespan and $\overline{\Delta F}_{c_j}$ (step 20). If its lifespan is longer than its MTBF, then bp_{c_j} can be set to a very large value to increase the chances of the component being selected for repair, or one can force its replacement, thus setting its breakage probability to zero in order to prevent its being selected. Finally, step 23 (i.e., the routine selectBestComponents) aims at selecting the sub-set of components $\bar{C} \subseteq C$ that is more convenient to repair with regards to the minimization of two criteria, under budget and time constraints. More specifically, $\bar{C}$ represents the set of components with a total repair cost and time, neither of which exceeds its maximum availability (B and T_{max}, respectively), simultaneously minimizing the total breakage probability and the maximum repair time.

All parameters in Table 21 are:

(1) derived from data on past failure events (E_s, E_b, ΔL_{c_j}, ΔT);

(2) extracted from data on components characteristics (rc_{c_j}, rt_{c_j}, n_{c_j} , $\overline{\Delta F}_{c_j}$);

(3) constants provided by domain experts (T_{max}, B, C_{work}); or

(4) estimated by Algorithm 1 (bp_{c_j}).

It is worth noting that the value of ΔT should not be chosen too small, otherwise only a few breakages would be considered, and the maximum likelihood estimation criterion implemented in Algorithm 1 would produce inaccurate results. Similarly, ΔT should not be too wide; otherwise, the breakages would be poorly correlated with stoppages of the plant, again returning inaccurate results. For these reasons, the value of ΔT is set on the basis of past knowledge of plant stoppages. In particular, ΔT is set to the average time between two consecutive stoppages.

4.3.2.3 Decision support system based on two multi-objective optimization approaches

A bi-objective Mixed-Integer Non-Linear Programming formulation

In this Section, the bi-objective Mixed-Integer Non-Linear Programming formulation is described. It is modeled with the aim of selecting the subset $\tilde{C}$ of components from the set C in order to optimize two criteria simultaneously. The former takes into account the minimization of the total breakage probability of the non-selected components (i.e., the maximization of the total breakage probability of the selected components and thus, of the system reliability). While the latter refers to the minimization of the maximum repair time (i.e., the makespan). For this purpose, we introduce the following decision variable x_{c_j}, $\forall\ c_j \in C$, equal to 1 if the component c_j is selected for being repaired, 0 otherwise.

$$\min BP = \sum_{c_j \in C} \left(1 - x_{c_j}\right) bp_{c_j} \tag{5}$$

$$\min RT^{max} = \max_{c_j \in C} rt_{c_j} x_{c_j} \tag{6}$$

s.t.

$$C_{work} \sum_{c_j \in C} n_{c_j} x_{c_j} rt_{c_j} + \sum_{c_j \in C} rc_{c_j} x_{c_j} \ \leq B \tag{7}$$

$$\sum_{c_j \in C} rt_{c_j} x_{c_j} \ \leq T_{max} \tag{8}$$

$$x_{c_j} \in \{0; 1\} \quad \forall c_j \in C \tag{9}$$

The objective function *BP* (5) to minimize represents the total breakage probability associated with the unselected components. While, RT^{max} (6) to minimize denotes the maximum maintenance time, i.e., that the maximum time devoted to repair a selected component. This is motivated by the fact that we do not take into account also the scheduling of the maintenance activities but only the planning. Therefore, the companies may be very interested in having a maintenance plan allowing them to save time. Constraint (7) assures that the total cost used for repairing does not exceed the maximum budget allowed (*B*). It is worth noting that the total cost is due to two parts. The former considers the effective time cost of the maintainers used, knowing the time required by each component to be repaired. Instead, the latter refers to the repair cost of each component that also takes into account the material used for repairing. It is worth remarking that the costs do not include those due to the component's shortage. In fact, we consider that the predictive maintenance activities, especially in complex systems, are usually performed in the medium/long term. Therefore, a shortage of the components to be repaired is highly unlikely. These facts have been confirmed by the company from which we derived the problem and the data. Constraint (8) imposes that the total time spent for repairing does not exceed the total time available (Tmax). Finally, constraints (9) define the binary nature of the decision variables.

In order to linearize the proposed model, we introduce an additional continuous non-negative variable *y* equal to $max_{c_j \in C}\, rt_{c_j}x_{c_j}$ and, thus, the following additional constraint (10):

$$y \geq rt_{c_j}x_{c_j} \;\forall c_j \in C \tag{10}$$

For normalizing the values of the two objectives, we also divide the value of the variable *y* by $T_{max.}$ Therefore, the Mixed-Integer Linear Programming (MILP) formulation proposed for the b-CRP is in the following:

$$minBP = \sum_{c_j \in C}(1 - x_{c_j})bp_{c_j} \tag{11}$$

$$\min RT^{max} = \frac{y}{T_{max}} \tag{12}$$

s.t.

$$C_{work}\sum_{c_j \in C} n_{c_j}x_{c_j}rt_{c_j} + \sum_{c_j \in C} rc_{c_j}x_{c_j} \leq B \tag{13}$$

$$\sum_{c_j \in C} rt_{c_j}x_{c_j} \leq T_{max} \tag{14}$$

$$y \geq rt_{c_j}x_{c_j} \;\forall c_j \in C \tag{15}$$

$$x_{c_j} \in \{0; 1\} \quad \forall c_j \in C \tag{16}$$

The Augmented ε-Constraint Approach

In this Section, the AUGMEnted ε-CONstraint (AUGMECON) approach is described. It is introduced in Mavrotas [178] and Mavrotas and Florios [179] for solving the bi-objective MILP model (11)-(16). Indeed, this approach has already been applied successfully for solving several other decision problems (e.g., [180,181]).

Generally speaking, in a bi-objective optimization problem, the objective function $f(x)$, supposed to be minimized, can be expressed through a bi-dimensional vector $z = f(x) = (z_1 = f_1(x), z_2 = f_2(x))$, being the *n*-dimensional vector x a feasible solution in the feasible region $X \subseteq \mathbb{R}^n$. Therefore, the following two definitions hold:

Definition 1 *Dominance condition*:

A solution x' with (z_1', z_2') dominates a solution x'' with(z_1'', z_2'') if and only if $z_1' \leq z_1''$ and $z_2' \leq z_2''$ and at least one inequality is strictly satisfied.

Definition 2 *Pareto Efficiency*:
A solution $x \in X$ is Pareto Efficient if and only if $\nexists x' \in X$ that dominates it.
The Pareto Front contains all the Pareto Efficient solutions. The methods proposed for solving multi-objective optimization problems can be classified into three different classes: *a-priori*, *interactive,* and *a-posteriori* [182]. The *a-priori* methods (e.g., GP methods) assume to know all the preferences before starting the decision-making process, finding solutions that respect all of them. In the *interactive* approaches instead, it is assumed that all the preferences are introduced by the decision-maker during the decision-making process. Therefore, these methods require several interactions with him/her. Finally, in the *a-posteriori* approaches, all the efficient solutions are firstly generated and then analyzed according to the preferences of the decision-maker. The *Weighted Sum* and the *ε-Constraint method* are the most widely used *a-posteriori* approaches.
Regarding the ε-Constraint approach, in case of bi-objective optimization, the following optimization problem is defined (17)-(19):

$$min\ z_1 \tag{17}$$

s.t.

$$z_2 \leq \varepsilon_2 \tag{18}$$

$$x \in X \tag{19}$$

By properly varying the ε_2 parameter, right-hand side of the introduced constraint (**18**), the efficient solutions can be determined. However, one issue is related to how setting the variation range of ε_2. One way is to build a square *payoff table*, with a number of columns (rows) equal to that of the objective functions, through *lexicographic* optimization. In case of bi-objective optimization, the first row of such a 2 × 2 payoff matrix contains z_1^* and $\bar{z}_2$, respectively, denoting the optimal value of z_1 when only it is optimized and the optimal value of z_2 when only it is optimized under the constraint $z_1 = z_1^*$. In the second row, instead, it reports $\bar{z}_1$ and z_2^*, where the latter is the optimal value of z_2 when only it is optimized while the former is the optimal value of z_1 when only it is optimized under the constraint $z_2 = z_2^*$.

In order to avoid generating weakly Pareto efficient solutions, the AUGMECON approach is used where the payoff table is firstly derived through the lexicographic optimization and then, the variation range of ε_2 is determined as $[z_2^*, \bar{z}_2]$. In addition, it is required that constraint (18) has to be binding. Therefore, it is transformed into an equality by subtracting a surplus auxiliary variable *s*. Such an additional variable is also introduced in (17) with lower priority, multiplied by $\frac{eps}{\delta}$, where eps represents a user defined constant and δ is computed as $\delta = \bar{z}_2 - z_2^*$ (i.e., it is the width of the variation range). The ε_2 parameter is then varied in the range $[z_2^*, \bar{z}_2]$ by a step $\delta_{\varepsilon_2} = \frac{\delta}{\alpha}$ where α is a user input value.

It is worth remarking that, in order to avoid the *trivial* solution (i.e., the one in which no component is repaired) when only RT^{max} is minimized, the following constraint is added to the formulation (11)–(16):

$$y \geq rt^{min} \left(1 - \frac{1}{|C|} \sum_{c_j \in C} x_{c_j}\right) \tag{20}$$

where |C| counts the number of components and $rt^{min} = min_{c_j \in C} \{rt_{c_j}\}$.

A Multi-objective Large Neighborhood Search

In order to efficiently solve also medium and large-sized instances, we propose a bi-objective Large Neighborhood Search (hereafter, denoted as b-LNS). The Large

Neighborhood Search (LNS) is a meta-heuristic successfully used in several and also different application contexts (e.g., [183–185]). Its main advantage is searching larger and complex neighborhoods so that better quality solutions can be found [186]. In particular, when the decision problem is defined with very tight constraints, it could be very easy to get stuck at a local minimum [187]. The LNS was proposed in Shaw [188] to solve the Capacitated Vehicle Routing Problem.

The main idea is that, starting from an initial solution (e.g., randomly generated), applying both *destroy* and *repair moves* at each iteration, a better solution can be detected. In order to reduce the computational time, it usually starts with a small-sized neighborhood, gradually increased during the search [188]. To the best of the author's knowledge, very few scientific contributions in the literature have already proposed a multi-objective version of it. In particular, a b-LNS is proposed for solving a bi-objective Tank Allocation Problem and a bi-objective Energy-Flexible Flow Shop Scheduling problem, respectively, in Schaus and Hartert [189] and Oddi et al. [190]. In the following, a b-LNS is specifically proposed for solving the b-CRP outlined in Algorithm 2. In the b-LNS, the *destroy moves* are those that aim at removing components from the current solution (i.e., *remove moves*), while the repair ones are those aimed at adding new components in the current solution (i.e., *add moves*).

The b-LNS receives a Time Limit (TL), a user-selected parameter γ and two integer number (num_R and num_A) denoting, respectively, the number of both remove and add moves implemented. The parameter γ, used only in two of the moves described in the following, denotes the percentage of either the components already repaired in the current solution that have to be removed or the components not repaired in the current solution that have to be selected in the new solution. It returns the set S_{best} of non-dominated solutions.

Algorithm 2 b-LNS outline

Input: TL, γ, num_R, num_A;
Output: Set of non − dominated feasible solutions S_{best}
1: $S_{best} := \emptyset$
2: $sol :=$ ***InitialSolution***();
3: $S_{best} := S_{best} \cup \{sol\}$;

4: ***while*** ! ***stop***(TL) ***do***
5: RM := **SelectRandom**(num_R);
6: IM := **SelectRandom**(num_A);
7: **if** $RM \neq 0 \vee IM \neq 0$ **then**
8: sol := **Neigh**(RM, IM, sol, γ);
9: **else**
10: IM := **SelectRandom**$(num_A - \mathbf{1}) + \mathbf{1}$;
11: sol := **Neigh**(RM, IM, sol, γ);
12: **end if**
13: **if** ! **Dominated**(sol) **then**
14: $S_{best} := S_{best} \cup \{sol\}$;
15: **end if**
16: sol := **SelectRandom**$(|S_{best}|)$;
17: **end while**

The routine *InitialSolution* generates an initial solution in which the components to repair are selected by decreasing repair times; equal, by increasing breakage probability; equal, by decreasing repair cost. In any case, the selection is performed respecting both the total budget (B) and the total repair time (T_{max}) available. The routine *stop* returns TRUE if *TL* is reached; FALSE, otherwise.

The routine *SelectRandom* receives an integer number (i.e., num_R or num_A) and returns an integer number, randomly generated, between 0 and the input number, representing the id of either a remove or an add move to apply (steps 5−6). In steps 7−12, a new solution is found by starting from the current one and by applying the moves selected in the previous steps. In particular, the routine *Neigh* applies the remove move and the add move to the current solution *sol*. The *Remove* moves are detailed in the following:

(1) *RemoveRandom*: randomly selects a component to be removed from those repaired in *sol*;
(2) *RemoveBP*: selects the component with the lowest breakage probability to be removed from those repaired in *sol*;
(3) *RemoveTime*: selects the component with the highest repair time to be removed from those repaired in *sol*;

(4) *RemoveBPT*: selects the component c_j to be removed from those repaired in *sol* (i.e., $\overline{C}(sol)$) such that:

$$c_j := \underset{c_k \in \overline{C}(sol)}{\operatorname{argmin}} \left\{ bp_{c_k} + \frac{1 - rt_{c_k}}{T_{max}^{sol}} \right\} \tag{21}$$

where T_{max}^{sol} represents the maximum repair time over all the repair times of the components selected in *sol*;

(5) *RemoveGamma*: randomly selects γ components already repaired in *sol* to be removed.

It is worth remarking that more than one component may be individuated by the RemoveBP, RemoveTime, and RemoveBPT moves to be potentially removed. This is due to the fact that, especially in the application context of this work, more than one component may satisfy the requirements imposed by each of these moves.

The *Add* moves are described in the following:

(1) *AddRandom*: randomly selects a component to be added from those not repaired in *sol*;
(2) *AddBP*: selects the component with the highest breakage probability to be added from those not repaired in *sol*;
(3) *AddTime*: selects the component with the lowest repair time to be added from those not repaired in *sol*;
(4) *AddBPT*: selects the component c_j to be added from those not repaired in *sol* (i.e., $C \setminus \overline{C}$ (*sol*)) such that:

$$c_j := \underset{c_k \in C \setminus \overline{C}(sol)}{\operatorname{argmin}} \left\{ bp_{c_k} + \frac{1 - rt_{c_k}}{T'_{max}} \right\} \quad (22)$$

where T'_{max} denotes the maximum repair time of the components not repaired in the solution sol;

(5) *AddGamma*: randomly selects γ components not repaired in sol to be added.

It is worth noting that the add moves are applied respecting the constraints of budgeting and the total repair time. It means that, for instance, the component randomly chosen in the *AddRandom* move is checked, respecting both the constraints. Moreover, the *RemoveBPT* aims at selecting the component to remove that is a trade-off between a low breakage probability and a high repair time. Similarly, the *AddBPT* adds the component that is a good compromise between a high breakage probability and a low repair time. In addition to the moves described above, there are two moves (*Switch* and *NoMove*), shared between the remove and the add moves and applied in the case in which the move id is, respectively, equal to 6 and 0. In particular, the *Switch* move randomly selects one component repaired in *sol* and one component not repaired in *sol* such that the former is removed and the latter is added in the new solution, considering the respect of the problem constraints. While the *NoMove* move does not apply any remove/add move to *sol*. However, we avoid the situation in which both the ids are equal to 0 and then that, at an iteration, no components are removed and added from/to *sol* (steps 10-11).

The routine *Dominated* (step 13) returns TRUE if *sol* is dominated; FALSE, otherwise. In order to establish if a solution is dominated, it is compared with all the non-dominated ones found so far and already stored in S_{best}. If a new non-dominated solution sol is found, it is added to S_{best} and then, non-dominated solutions already found but dominated by sol are consequently removed from S_{best}.

In step 16, the routine *SelectRandom* returns a randomly generated number that is the position in S_{best} of the new starting solution to select at the next iteration. This step is introduced to shake the algorithm, trying to explore different areas of the searching space.

It is worth remarking that, in order to speed-up the b-LNS, each time a new solution is found the value of the objective (11) can be computed as the gap with that of the current solution. More specifically, let $BP(sol)$ and $BP(sol')$ be respectively the values of (11) with reference to the current solution *sol* and to the solution *sol'* found after applying a remove and an add move to *sol*. Let $C_{remove} \subseteq \overline{C}(sol)$ and $C_{add} \subseteq C \setminus \overline{C}(sol)$ be the list of components to remove and to add, respectively. Then, $BP(sol')$ can be computed as

$$BP(sol') = BP(sol) + \sum_{j \in C_{remove}} bp_{c_j} - \sum_{i \in C_{add}} bp_{c_i}.$$

Concerning RT^{max}, let $RT^{max}(sol)$ and $RT^{max}(sol')$ be the maximum repair time of sol and sol', respectively. Let

$$\overline{rt} = max_{c_j \in C_{add}}\{rt_{c_j}\} \text{ and } \widehat{rt} = max_{c_j \in C_{remove}}\{rt_{c_j}\}$$

be the maximum repair time of the components belonging to C_{add} and C_{remove} respectively. If $RT^{max}(sol) \leq \overline{rt}$ then $RT^{max}(sol') = \overline{rt}$, else if $RT^{max}(sol) > \widehat{rt}$ then $RT^{max}(sol') = RT^{max}(sol)$, otherwise

$$RT^{\max}(sol') = max_{c_j \in (\overline{C}(sol) \setminus C_{remove}) \cup C_{add}}\{rt_{c_j}\}.$$

Since the number of components to add and remove at each iteration is less than $|\bar{C}(sol)|$ (at most $\gamma|\bar{C}(sol)|$), the time to compute $RT^{max}(sol')$ is on average less than the time to compute the maximum on the whole set $\bar{C}(sol)$. In the worst case, the complexity of each iteration of the b-LNS is $O(\Gamma\log(\Gamma))$, where $\Gamma = max\{|\bar{C}|, |C \setminus \overline{C}|\}$. Indeed, in the worst case, the complexity of **RemoveBP** and **RemoveTime** is $O(|\bar{C}|log(|\bar{C}|))$, the complexity of **AddBP** and **AddTime** is $O(|C \setminus \overline{C}|log(|C \setminus \overline{C}|))$.

4.3.2.4 Computational results

Experimental setting

The parameter settings are fixed in collaboration with a domain expert. Moreover, seven instances are generated and grouped into three sets, namely *small* (*i.e., I*20, *I*40, *I*80), *medium* (*i.e. I*160, *I*320) and *large* (*i.e. I*640, *I*1280) as reported in Table 22.

Table 22 Parameter settings and instances generation

Parameter	**Value**
B	170000 €
T_{max}	1440 min
C_{work}	30 €/h
ΔT	1month
Instance name	**|C|**
*I*20	20
*I*40	40
*I*80	80
*I*160	160
*I*320	320
*I*640	640
I1280	1280

Each instance was randomly generated from the case study distribution. It is worth remarking that the largest number of components is 1280 since it is greater by far than the number of components of very complex case study like the one considered, hence it is not reasonable to have more than that.

The AUGMECON method was run with an increasing step equal to one, while *eps* and the total time limit were set to 104 and 3600 seconds, respectively. Furthermore, each MILP, at each iteration of AUGMECON, was also solved with a CPU time limit of 3600 seconds.

Concerning the b-LNS, *TL* was set to 3600 seconds. Its results were collected in specific time instants, namely 0.1, 0.2, 0.5, 1, 2, 3, 5, 10, 20, 30, 50 seconds, then from 100 to 1000 seconds in steps of 100 and from 1000 to 3600 seconds in steps of 200. However, for the sake of simplicity, only the best results obtained are reported. The γ parameter was set to 0.20 for both *small* and *medium* sets. Instead, for *large* instances, it was varied from 0.05 to 0.20, and the best value found experimentally was taken. Specifically, for *I*640 and *I*1280, it was fixed to 0.19 and 0.09, respectively.

The results of both AUGMECON and b-LNS were evaluated considering the following multi-objective metrics:

- the number of non-dominated solutions (η);
- the number of non-dominated solutions of one of the two approaches that are actually dominated by the other ($\hat{\eta}$);
- *Spacing* (*S*), i.e., a metric introduced in Schott [191] that measures the range variance of neighboring solutions in the front. In other words, it measures the distribution of the solutions along the front, and it is defined as follows:

$$S = \sqrt{\frac{1}{\eta}\sum_{i=1}^{\eta}\left(d_i - \bar{d}\right)^2} \tag{23}$$

where $\bar{d}$ is the average of all the distances d_i, for all *i* = 1, . . . , η, and the *i*th distance d_i is computed as:

$$d_i = \min_{j \in F:\, i \neq j}\left\{\left|\widetilde{BP}_i - \widetilde{BP}_j\right| + \left|\widetilde{RT}_i^{max} - \widetilde{RT}_j^{max}\right|\right\} \tag{24}$$

where F denotes the front, whereas $\widetilde{BP}_i$ and $\widetilde{RT}_i^{max}$ represent, respectively, the normalized value of the two objective functions of the *i*th non-dominated solution. For example, $\widetilde{BP}_i$ is computed as follows:

$$\widetilde{BP}_i = \frac{BP_i - \min_{j \in F} BP_j}{\max_{j \in F} BP_j - \min_{j \in F} BP_j} \tag{25}$$

It is worth remarking that the smaller the *S* value, the higher is the diversification of *F*.

Numerical results

Table 23 reports the numerical results of the experiments. Considering the small- and the medium-sized instances, both approaches always give the same front and then the same S value. In particular, on the small instances, the computational time required by the AUGMECON (i.e., 48.51, 92.55, 139.74, respectively) is higher by far than that of b-LNS (i.e., 0.5, 0.5, 0.5 seconds, respectively). On *I*160, the percentage increment of the computational time required by AUGMECON (i.e. 223.39 seconds) with respect to that of b-LNS (i.e., three seconds) is 7346%. Finally, on *I*320, the computational time required by AUGMECON is higher by far than that of b-LNS, i.e., 640 seconds against 100 seconds.

Table 23 Numerical results of experiments on small, medium and large sets.

		AUGMECON			b-LNS		
	Instance	**η**	$\hat{\eta}$	**S**	**η**	$\hat{\eta}$	**S**
Small	***I*20**	9	0	0.069	9	0	0.069
	***I*40**	13	0	0.055	13	0	0.055
	***I*80**	12	0	0.1	12	0	0.1
Medium	***I*160**	12	0	0.083	12	0	0.083
	***I*320**	8	0	0.155	8	0	0.155
Large	***I*640**	8	0	0.038	8	0	0.038
	***I*1280**	6	0	0.057	9	1	0.042

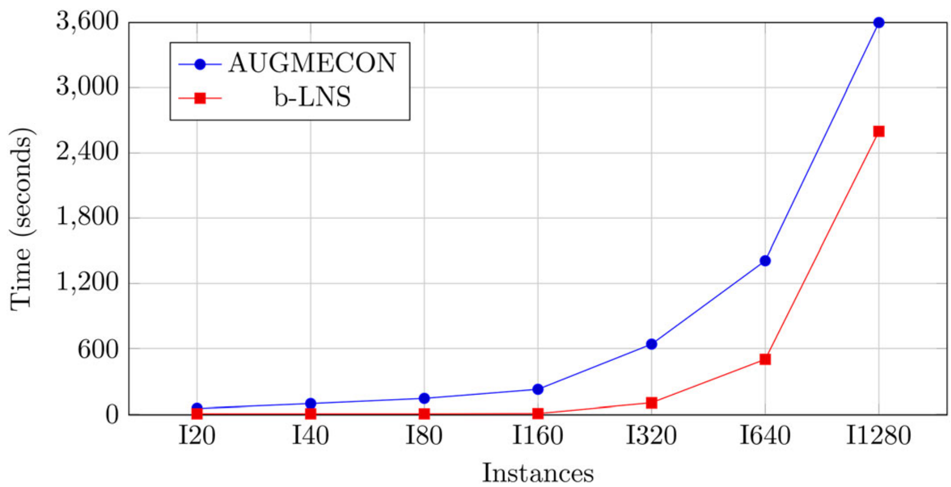

Figure 10 Computational time of AUGMECON and time for the best front of b-LNS.

As for the *large* set of instances, on *I*640, both approaches give the same front. The computational time required by AUGMECON is higher than that of b-LNS, i.e., 1409.45 seconds against 500 seconds, respectively. Instead, on I1280, AUGMECON, which also reaches the time limit, returns six non-dominated solutions. For b-LNS, η is nine, but one solution is actually dominated by those of AUGMECON.

In order to obtain a better front, AUGMECON was run without a time limit, but after 12 hours of computation, it failed to finish, fully saturating the memory. The time required by b-LNS to return the best front on *I*1280 was 2600 seconds. Comparing the fronts obtained by the two approaches, AUGMECON was not able to find the three solutions with lower values of *BP*. Instead, the solution with a higher value of *BP* for b-LNS was actually dominated by one solution found by AUGMECON.

In general, as expected, the computational times of AUGMECON are on average about two orders of magnitude greater than those of b-LNS for obtaining the same front (see Figure 10), although both the solution methods are comparable from the non-dominated solutions point of view. Indeed, AUGMECON on average solves 3157, 8624 and 20098 MILPs on *small*, *medium* and *large* sets, respectively.

Furthermore, b-LNS never reaches the time limit, whereas AUGMECON reaches one hour of computation on I1280. For *small* and *medium* sets, the execution time of b-LNS is lower than 100 seconds, whereas for *large* sets, whatever the value of γ, it always requires less than 2600 seconds. It is noteworthy that AUGMECON uses CPLEX for solving each MILP, which runs in multi-threaded mode exploiting all eight threads of the processor. Instead, b-LNS runs sequentially.

Regarding the γ parameter, it is worth noting that b-LNS returns the best results with high values up to *I*640, whereas small values are used for *I*1280, as already anticipated in the previous section. In fact, the parameter γ denotes the percentage of components in a solution that has to be removed/added. In this way, it represents a perturbation (i.e., *shaking*) applied to the search space. Numerical results suggest that, for *small/medium* instances, shaking the current solution a lot may help the algorithm finding new non-dominated solutions. Instead, for *large* instances, shaking a bit at a time is preferable. This phenomenon mainly depends on the time needed to execute the *AddGamma* and the *RemoveGamma* moves. Indeed, with high values of γ, both moves require computational times to be executed higher than those needed with small values. Therefore, few moves can be applied in the CPU time limit of 3600 seconds. This may have a negative impact on the large instances where the solution space increases. Figure 11 shows the trend of $\hat{\eta}$ on $I1280$, by varying the value of γ from 0.05 to 0.20. In particular, with higher values ($\gamma \in [0.15, 0.20]$), the b- LNS performance deteriorates since its results are worse than those obtained with lower values of this parameter.

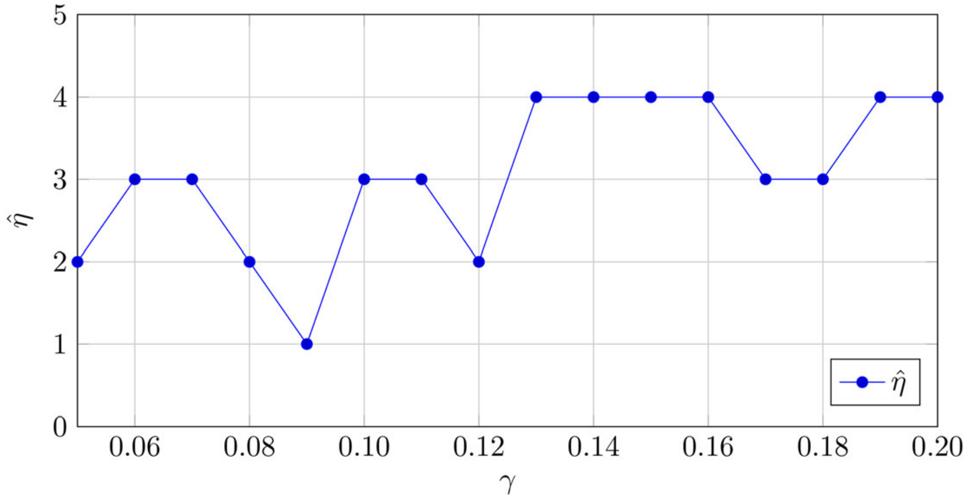

Figure 11 Values of $\hat{\eta}$ on $I1280$ varying the parameter γ .

4.3.3 Control layer: A data-driven analysis of the moves effectiveness

In this section, the effectiveness of the moves of b-LNS is studied through a data-driven analysis performed on $I1280$. On it, in fact, b-LNS provides more non-dominated solutions than those detected by AUGMECON (which reaches the one-hour time limit) but with a value of $\hat{\eta}$ equal to one. This means that one of the non-dominated solutions of b-LNS is actually dominated by AUGMECON. The data-driven analysis is carried out by running b-LNS on $I1280$ for 100 seconds, gathering information on a total number of 6,491,782 pairs of moves. Each pair is made up of both a *Remove* and an *Add* move and has been classified as either an ND or D pair in the case in which it returned either a Non-Dominated or a Dominated solution, respectively.

Table 24 reports the percentage of times in which a given pair of moves has been selected and has been classified as an ND pair (i.e., it has been effective). Among the 6,491,782 pairs of moves used in 100 seconds, 204 of them have been effective, giving a non-dominated solution. In particular, the most profitable remove move is *RemoveTime,* i.e., in almost 31.37% of cases, while the second most profitable one is *NoMove* (i.e., 21.57% of cases). The most profitable *Add* move is *AddGamma*, in about 52.45% of cases. It is worth noting that each solution has 13.89 components on average, and then *AddGamma* adds three components on average. The second most profitable *Add* move is *AddTime*, in about 14.22% of cases. The pairs of moves (*RemoveBP, AddGamma*), (*NoMove, AddTime*), (*RemoveTime, AddGamma*) and (*RemoveRandom, AddGamma*) perform the best. It is worth noting that all of them add more components than those removed (at most one at a time).

The *Switch* move is effective only in 8.33% of cases, and it is also the most time-consuming move. In addition, *RemoveBPT* and *AddBPT* are effective in only 4.90% and 4.58% of cases, respectively. For this reason, all of these three moves may be candidates to be removed from the set of available moves. Therefore, b-LNS is run on *I*1280 without *Switch, RemoveBPT* or *AddBPT* moves, obtaining η = 9 (greater than that of AUGMECON—equal to six) but with $\hat{\eta}$ equal to zero, meaning that no solution is dominated by those of AUGMECON. Moreover, in this case, the best front is obtained by b-LNS in only 1200 seconds.

Table 24 Data-driven analysis of the moves effectiveness.

	AddRandom	**AddBP**	**AddTime**	**NoMove**	**AddBPT**	**AddGamma**	**Switch**
	(%)	**(%)**	**(%)**	**(%)**	**(%)**	**(%)**	**(%)**
RemoveRandom	0.49	0.49	0	0	0.98	9.8	0.49
RemoveBP	0.98	0.49	0	0	0	13.73	0
RemoveTime	2.45	4.41	1.47	5.39	0.98	13.73	2.94
NoMove	4.41	1.47	10.29	0	1.47	3.92	0
RemoveBPT	0	0.49	0	1.47	0	2.94	0
RemoveGamma	1.47	0.49	2.45	0	1.96	3.43	0
Switch	0	0	0	0	0	4.9	0

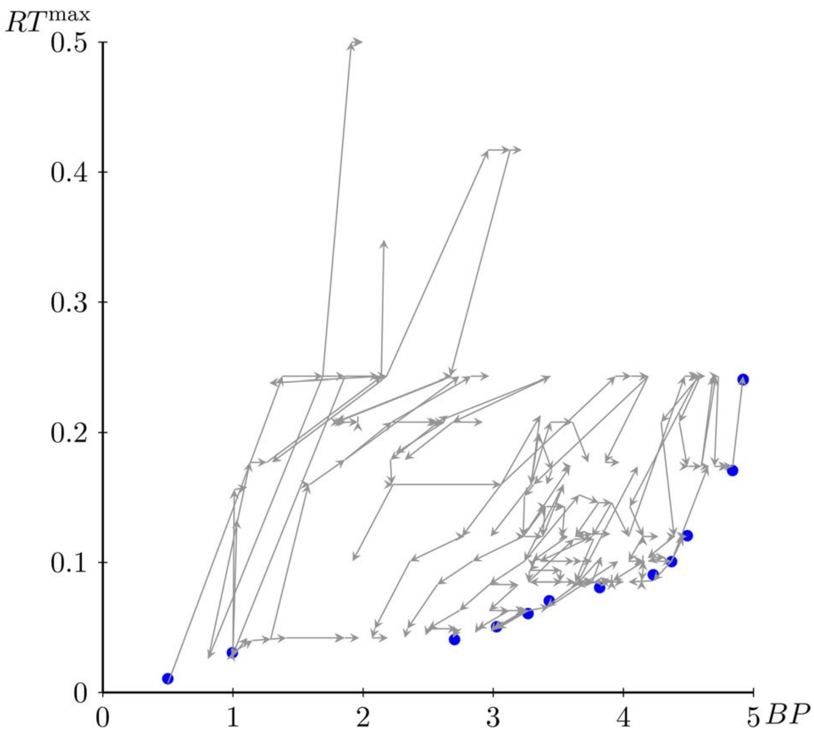

Figure 12 Solutions generated by b-LNS on *I*160.

Figure 12 shows the b-LNS execution on *I*160. In particular, each point represents a new nondominated solution generated during a specific iteration. The arc going from point *p* to point *p*' indicates that the non-dominated solution *p*' has been generated starting from the non-dominated solution *p*. Points with only ingoing arcs are those that, although they have been explored further, have not generated new non-dominated solutions (*e.g.* the point with $BP = 2$ and $RT^{\max} = 0.5$). It is worth noting that the solutions concentrate on the neighborhood of the solutions belonging to the front (denoted by the points). In particular, it is interesting to observe that the higher concentration of those solutions is in the right-hand side of the plot, where there is the highest density of solutions. This indicates that around those solutions b-LNS introduces only small adjustments. In fact, the average length of arcs linking points having $BP>2.7$ and $RT^{\max} \leq 0.17$ (i.e., the region with 9 of 12 solutions) is 0.153, while the average length of other arcs is 0.263.

4.4 Data-driven maintenance policy based on Social Network Analysis

In this section, a further approach is proposed, basing the decision support model on the network built considering the relationships among components failures. The data-driven decision support system developed in this section aims at defining the optimal set of components to predictively be maintained in order to achieve high-reliability levels, avoiding the occurrence of failures. The analysis carried out in this layer is organized in three steps, capitalizing on two predictive analytics techniques, Association Rule Mining (ARM) and Social Network Analysis (SNA), and on the optimization of an Integer Linear Programming (ILP) model.

4.4.1 Data gathering and management

The data belong to a medium-sized Italian oil refinery plant, specifically to the topping sub-plant. Details on the case study are not reported, since they can be found in previous paragraphs. The time interval of reference for the analysis regards a period of three years, during which operational data (e.g., flows, density, pressures) have been monitored: in case of missing data, the mean value of the previous working day has been used to replace them, as well as in the case of anomalous measurement reported (e.g., out of scale values). In addition, the work orders and maintenance activities required for the components of the plants have been considered, compared, and integrated with the notes taken by the supervisors of the plant, in order to define whether all the activities performed on the plant have been inserted into the information system and to ensure consistency among the two information sources. As required by the first layer of the decision support system, the integrated data are taken from the CMMS of the refinery.

4.4.2 Data analytics

Like for the other case studies, the data analytics layer is divided into three stages.

4.4.2.1 Preliminary analysis

Considering the data of the CMMS, the preliminary failure analysis is carried out: in all, 82 components are monitored in the sub-plant. Statistically, 46 of them have been considered for the analysis since they caused 615 failures over the 767 failures in three years, which is more than 80%. In order to define the time interval to consider the failures “concurrent”, the maintenance department members have been interviewed to understand, based on their experience, which interval could be suitable for searching related failures. According to the interviewees, the maximum interval is set to two weeks: this means that the relations searched in the data concern component failures taking place at a distance of a maximum of two weeks.

4.4.2.2 Breakage probability estimation

Through the ARM, the concurrent failures analysis is carried out: from the data stored in the CMMS, information on the failures that occurred on the analyzed asset is extracted to identify the sets of components frequently failing together and the corresponding failure probability. Indeed, ARM aims at individuating attribute-value conditions frequently occurring together: in this way, the knowledge of the asset behavior is deepened by extracting previously unknown patterns from the data. Then, the association rules describing such relations are mined. An excerpt from the rules extracted is reported in Table 25. The rules can be interpreted as follows: a failure of component C15 is followed by the failure of C2 within a two-week time interval with the confidence of 0.866, hence in 88.6% of the cases. Remarkably, when C2 fails, also C15 fails as well in the following two weeks. In Appendix B, the list of the components’ ID and their related name is reported.

Table 25 Excerpt of the association rules mined.

α	→	β	Confidence
C15	→	C2	0.866
C2	→	C15	1
C15	→	C40	0.657
C15	→	C13	0.657
C40	→	C15	0.92

C13	→	C15	0.958
C2	→	C40	0.677
C2	→	C13	0.677
C40	→	C2	0.84
C13	→	C2	0.875
……			

4.4.2.3 Decision support model based on Social Network Analysis

After the breakage probability estimation, performed through the ARM, the Social Network Analysis (SNA) is used to relate the components frequently failing together and identify the possible failure propagations among the related components. In this context, the graph theory underlying the SNA facilitates the understanding of the association among component failures, and provides a global view of the interactions among the components frequently failing together.

As defined by Otte and Rousseau [88], a Social Network (SN) is the representation of a social structure. It can be described by an ordered pair of vertices (or nodes) and connected by edges (E), G=(V, E). The classical application of SNA regards the analysis of social structures and the interactions among a set of actors: the actors are the nodes of the network, while the interactions are the edges. An SN is generated considering the associations among components extracted in the previous step of the analysis. In the current approach, the SNA is used to represent and analyze the relations among components frequently failing together, which is for representing the association rules mined.

Specifically, in the proposed framework, the actors, thus the nodes, are the components, while the interactions (arcs) are the concurrent failures: that is, if node i is directly connected to node j (i→j), it means that the rule i→j is mined in the previous step, indicating that when component i fails, usually components j fails as well. The confidence of the rule gives the probability of such a conditional event. The confidence value of the rule represents the weight of the arc.

In order to define which nodes might be more critical in terms of failure probability, two SN metrics are applied for the analysis:

(a) Out-Degree (OD): is calculated as the weighted sum of the arcs outgoing from a node [192]. Specifically, OD represents a measure of how much a node is connected to another: the higher the OD, the higher the probability that one of the following components fails.

(b) Betweenness Centrality (BC): is determined as follows: the shortest weighted paths between all couple of nodes are determined; the BC value equals the sum of the shortest weighted paths on which the node appears [193]. In other words, the BC measures how much a node is influent across the network [194] since a node having a high BC value can be considered as a bridge among separate portions of the network. Thus, if a component fails, the right candidate for predictive maintenance would be a component characterized by a high BC value.

In the current framework, the OD (a) is considered an indicator of the risk of failure of subsequent components: indeed, the OD is calculated for each node, sorting them in descending order. The components at the top of the list should be carefully monitored since they represent the most critical ones. The BC (b), instead, is considered when a failure on a component occurs: the failed component is definitely replaced, but also a predictive intervention on the consequent ones could be performed, with the aim of avoiding the propagation of a failure chain across the network. In defining the best set of components to be predictively replaced, the decision-maker is supported by an ILP model, whose formulation can be interpreted as follows:

$$\max \sum_j BC_j x_j \tag{26}$$

$$\sum_j c_j x_j \leq \boldsymbol{B^{max}} \tag{27}$$

$$\sum_j t_j x_j \leq \boldsymbol{T^{max}} \tag{28}$$

$$\sum_j R_j x_j \leq \boldsymbol{R^{max}} \tag{29}$$

$$x_j \in \{0,1\} \, \forall \, j \qquad (30)$$

the decision variable x_j represents the j components of the assets. It can assume a value of 1 if component j is selected for the predictive maintenance, or 0, otherwise, as expressed in constraint (30). The objective function (26), to maximize, assures that the components having the highest BC are selected. Constraint (27) requires that the selection is performed according to a predefined maximum budget ($\boldsymbol{B^{max}}$), considering the cost of each component (c_j). Constraints (28) and (29), similarly, require the selected components to respect a maximum amount of time ($\boldsymbol{T^{max}}$) and resources ($\boldsymbol{R^{max}}$) to perform the intervention. Indeed, among the data, the time required to replace a component (t_j) and the number of operators necessary for replacing the component (r_j) is known.

The following bullet list aims at summarizing the main stages of the predictive maintenance strategy explained in the previous sections, providing a useful roadmap to be followed by the maintenance department.

During the normal operating conditions of the plant, the procedure proposed in this application is the following:

1. Monitor the components having high values of OD, specifically all the components j such that $OD_j > OD_{max}$ in order to detect failures early;
2. When a failure on component *i* is detected:
   - 2.1 Perform a corrective intervention on *i*;
   - 2.2 Extract the set of consequent components using the ARM (all components *j* such that $i \rightarrow j$);
   - 2.3 Create the SN graph and calculate the BC_j for all j components;
   - 2.4 Solve the ILP model (26)-(30);
   - 2.5 Perform a predictive maintenance intervention on the optimal set of components identified in 2.4.
3. Return to 1.

4.4.2.4 Application of the proposed approach

Considering all the association rules mined, whose excerpt is reported in Table 25, the graph representing the relationships among the component failures is created (Figure 13), and the Social Network Analysis is performed. The 46 components represent the nodes of the SN, while their relations are the AR identified in the previous step. In all, 724 arcs connect the 46 nodes. As noted before, the weights assigned to the arcs are the respective confidence values of the corresponding rule. The thickness of each arc is proportional to the confidence of the relationship represented. For example, according to the representation, the confidence of the rule C41→C15 (confidence = 1.000), is higher than the one of C41→C25 (confidence = 0.375). For the sake of clarity, the weights are not reported in Figure 13.

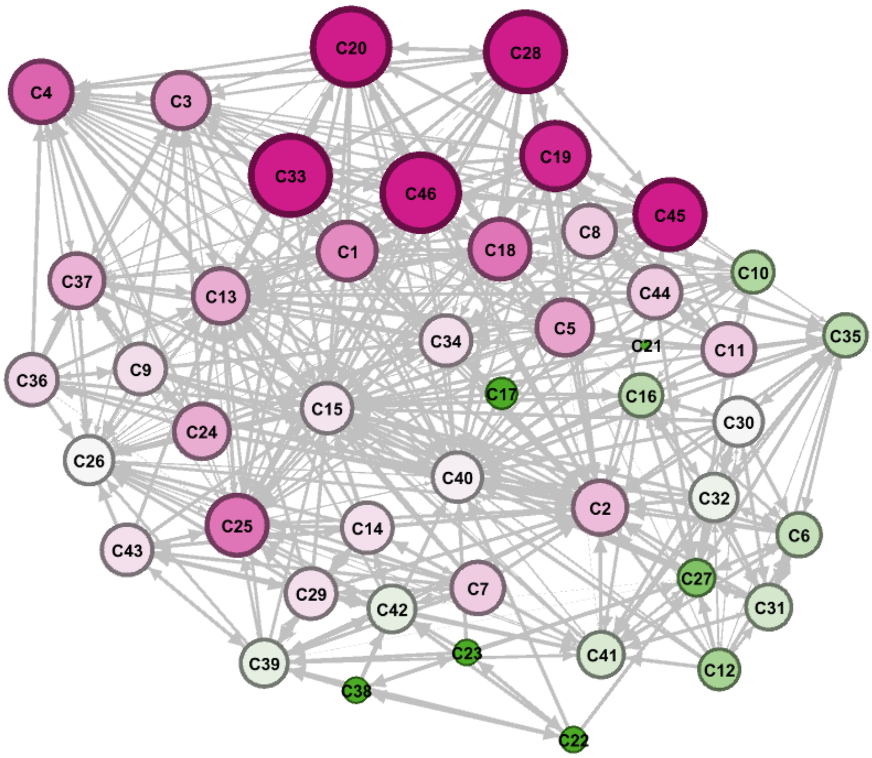

Figure 13 Representation of the Social Network characterized by 46 nodes and 724 arcs

The size of the nodes, instead, is proportional to the OD of the node itself; even its color is furtherly indicative of the OD: in particular, pink nodes are characterized by a high level of OD, while green ones by a lower level and the more intense the corresponding color, the higher the OD.

At this stage of the analysis, the calculation of Social Network Analysis is required, and hence, for each node, the OD is determined and reported in Table 26. Then the OD_{max} threshold has to be defined in order to identify the components that need to be carefully monitored by the operators: as explained before, the higher the OD, the higher the probability of failure of one of the consequent components.

Table 26 Out Degree values of the Social Network's nodes

Component	OD	Component	OD
C33	15.83	C43	10.67
C28	15.83	C29	10.67
C46	15.43	C14	10.67
C20	15.43	C15	10.61
C45	14.13	C40	10.44
C19	13.67	C30	10.29
C4	12.71	C26	10.28
C25	12.44	C32	10.13
C18	12.42	C42	10.00
C1	12.07	C39	10.00
C3	11.79	C41	9.75
C5	11.67	C31	9.67
C13	11.50	C6	9.40
C24	11.50	C35	9.25
C37	11.40	C16	9.22
C2	11.26	C10	9.00
C7	11.00	C12	8.80
C44	11.00	C27	8.08

C11	11.00	C17	7.00
C7	11.00	C38	6.00
C36	10.80	C23	6.00
C9	10.71	C22	6.00
C34	10.69	C21	3.00

Undoubtedly, the selection of the OD_{max} threshold has an impact on the operations of the maintenance department members. The higher the threshold value, the lower the number of components to be monitored. At the same time, it corresponds to a higher risk of failure. On the contrary, if the threshold is too low, there would be a high number of components to be carefully monitored, and the effort may not be repaid by the benefit. Considering the values reported in Table 26, an OD_{max}= 14.00 has been identified by company maintenance managers as a good compromise since it would require the careful monitoring of five components (C33, C28, C46, C20, C45). Lowering the threshold, for instance, to 12, would imply the double of the components (C33, C28, C46, C20, C45, C19, C4, C25, C18, C1) to be monitored, making this activity more onerous in terms of man-hours.

When a failure on a component occurs, on the other hand, it is necessary to decide whether to perform a predictive intervention on the consequent ones. For this purpose, the ILP model presented in the previous section is used. For instance, let us consider the failure and the replacement of component C15 (this component presents the highest value of Betweenness Centrality - BC). According to the association rules mined and the consequent SN created, 40 components (the ones highlighted in yellow in Figure 14) are related to C15. Hence, it would not be realistic to replace all of them in advance.

Therefore, the Betweenness Centrality (BC) value for all the components is calculated and given as the objective function of the ILP model, as well as the other data reported in Table 27, ranking them in descending BC. Such ranking allows us to visualize the most influential among the network, i.e., the ones with a higher criticality, at the top of the table; while, as we descend along the table, the remaining components will gradually become less influential and, therefore, less troublesome.

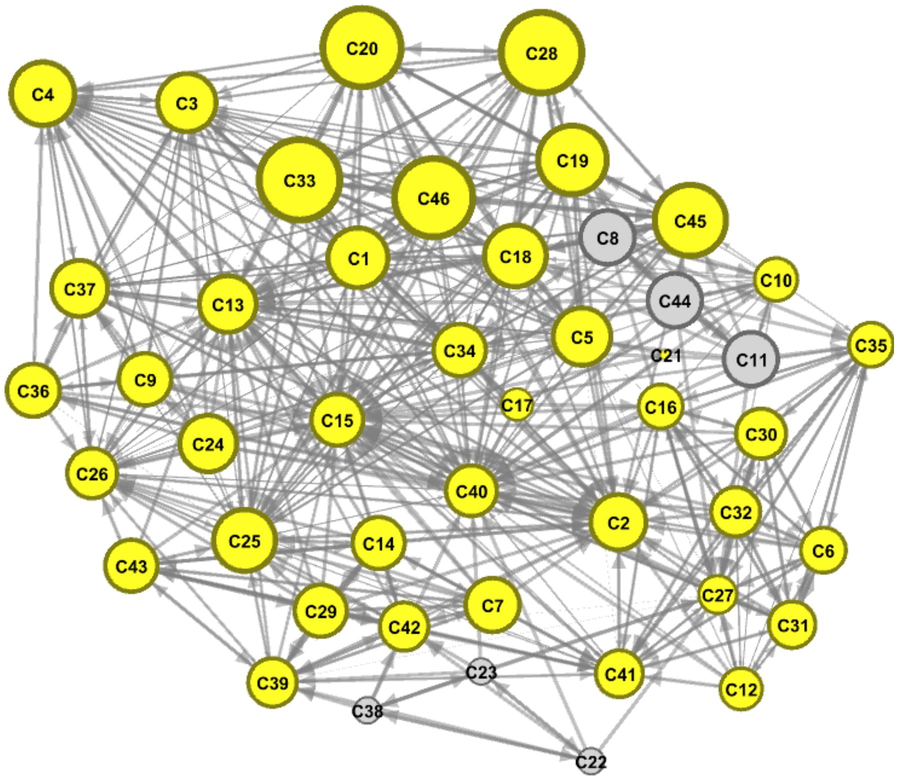

Figure 14 Social Network representation highlighting C15's consequent components.

Table 27 List of C15's consequent components and their associated repair cost (c_j), time (t_j), number of operators required (R_j) and Betweenness Centrality (BC_j).

Component	cj [€]	tj [min]	Rj	BC
C40	5931	600	2	192.79
C2	1184	250	1	188
C13	2300	750	1	104.53
C42	1311	286	1	74.66
C39	1274	175	1	74.66
C25	80	10	3	55.79

C41	235	299	1	54.17
C26	289	300	1	51.66
C5	2881	223	1	32.58
C27	190	60	2	30.22
C3	4094	255	1	20.66
C34	650	300	1	17.95
C4	1009	120	2	13.88
C37	2100	150	3	13.88
C35	1627	495	1	9.1
C18	845	90	1	8.43
C19	2103	66	1	7.81
C45	735	300	1	7.43
C1	3074	146	1	6.88
C9	1281	423	1	5.65
C36	3288	248	1	5.13
C16	2500	800	1	4.29
C6	1010	206	1	1.75
C12	2118	357	1	1.66
C43	2950	386	1	0.18
C29	577	529	1	0.18
C14	207	299	1	0.18
C31	1233	607	1	0.09
C32	402	68	1	0.09
C30	4063	333	1	0.09
C46	2930	329	1	0
C20	5041	122	1	0
C33	2061	212	1	0
C28	2302	340	2	0

The parameters of the work are set in collaboration with the maintenance department of the topping sub-plant, considering that a participatory approach allows a larger view of the entire context [195]. In addition, this decision enables the decision-makers to be consistent with their actual policies. Specifically, the maximum budget allowed for predictive maintenance of this plant ($\boldsymbol{B^{max}}$) is set to 3,000 € per week, while the maximum time ($\boldsymbol{T^{max}}$) is 350 minutes. In addition, a maximum of 5 operators ($\boldsymbol{R^{max}}$) can take part in the predictive maintenance activities. Considering these parameters and the data provided in Table 27, after the failure of C15, the results obtained recommend the replacement of components C2, C25, and C18, obtaining a total BC value of 252.22 (see Experiment 1 in Table 28). As presented in Figure 15, the items C15, C2, C25, and C18 are closely connected and are characterized by a vast number of ingoing and outgoing edges, making them critical in terms of influence among the network. The available time is saturated, as well as the number of operators employed in the operations. The budget needed to satisfy the requirement of such a solution, on the other hand, is lower than the $\boldsymbol{B^{max}}$ (2109 € out of 3000 €).

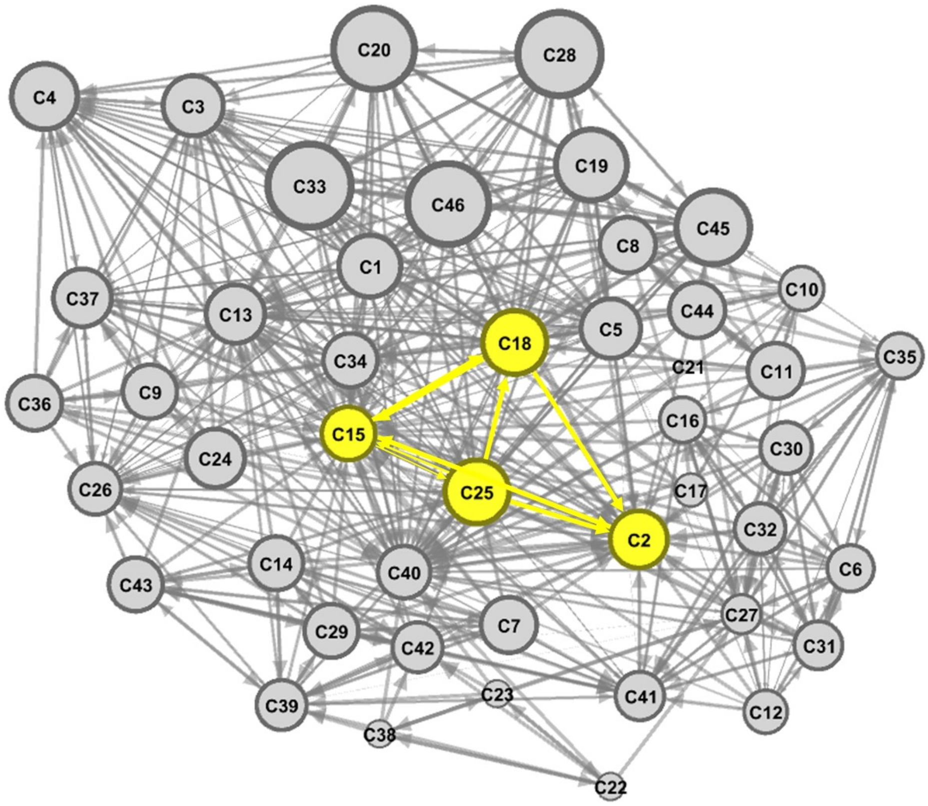

Figure 15 Social Network representation highlighting the components selected in Experiment 1.

4.4.3 Control layer: what-if scenario

A sensitivity analysis is performed to understand whether a relevant improvement could be obtained by adjusting the parameters. Indeed, budget, time, and human resources allocation is a critical activity for decision-makers, especially in large organizations, thus it is important to verify the impact of their decision and, possibly, adjust them.

Increasing the $\boldsymbol{R^{max}}$, without modifying the other parameters, has no impact on the solution found, as presented in Experiment 2 of Table 28; while, increasing the $\boldsymbol{T^{max}}$ by 25% - hence, extending it to 437.5 minutes - allows an improvement of the selected components:

C2, C25, and C39 are selected for predictive maintenance (Experiment 3 in Table 28). Even in this case, the constraint on $\boldsymbol{R^{max}}$ is saturated, while budget and time ones are not. Leaving $\boldsymbol{T^{max}}$ unchanged while increasing $\boldsymbol{R^{max}}$, provides the same solution; hence, it is decided to furtherly increase both the $\boldsymbol{T^{max}}$ up to 525 minutes and the $\boldsymbol{R^{max}}$ up to 10. The solution provided by this scenario recommends the replacement of four components (C2, C25, C27, and C39) as reported in Table 28 (Experiment 4). Introducing a further increment on the budget, hence increasing the $\boldsymbol{B^{max}}$ by 25% and 50%, assures the selection of 5 (C2, C25, C4, C18, and C39) and 6 (C2, C25, C4, C18, C32, and C39) components, respectively. As shown in lines with Experiments 5 and 6 of Table 28, there is no relevant increment in terms of objective function: this is justified by the fact that the further selected component (C32) has a low value in BC (0.09). Hence, in this case, it may not be convenient to increase the budget so much.

Table 28 Summary of the what-if scenarios

	Selected Components	BC	T^{max}	$\sum_j \boldsymbol{t_j x_j}$	B^{max}	$\sum_j \boldsymbol{c_j x_j}$	R^{max}	$\sum_j \boldsymbol{r_j x_j}$
Experiment 1	C2, C25, C18	252.22	350	350	3000	2109	5	5
Experiment 2	C2, C25, C18	252.22	350	350	3000	2109	10	5
Experiment 3	C2, C25, C39	318.45	437.5	435	3000	2538	5	5
Experiment 4	C2, C25, C27, C39	348.67	525	495	3000	2728	10	7
Experiment 5	C2, C25, C4, C18, C39	362.55	700	615	3750	3737	10	9
Experiment 6	C2, C25, C4, C18, C32, C39	362.65	700	683	4650	4139	10	10

According to the ARs mined, several relationships among component failures have been identified. Such failures might not be the ones expected by the plant technicians, even though they have extensive experience in the field. Indeed, one of the main theoretical contributions proposed in this work is that it is data-driven. Hence, the driver followed for defining the components to be replaced is the information extracted by the data, rather than the technical and physical structure of the process. For instance, rule C15→C2 indicates that, when a failure occurs on component C15, a controller, the coupling C2 also requires a replacement within two weeks, quite likely since the confidence is 0.886. This means that the controller C15 and the coupling C2 are likely to be replaced concurrently in a two-week time interval. Similarly, the ARs C15→C40 (confidence = 0.657) indicates that in more than 65.7% of cases after the failure of the controller C15, even the sealing device C40 has to be replaced. The explication of such relationships is evident from the data since the application of the proposed approach relies on a solid dataset and an appropriate amount of data, which is fundamental to deploy a data-driven framework.

Considering the prioritization of the components to be replaced, results show that the rationale is similar to the mining of the association rules. Indeed, the objective function of the ILP model takes into account the influence of each component across the SNA since it aims at selecting those having the highest BC, respecting the constraints. Recalling the example proposed in Section 4.4.2.4, we can say that when the failure of the controller C15 occurs, the most critical successors according to their BC value would be C40, C2, C13, C42, C39, C25, that are, respectively, the sealing device, the coupling, the insulation, the transmitting device, the measurement instrumentation, and the lighting. In all the what-if scenarios tested, the coupling and the lighting systems are selected, the instrumentation is selected in Experiments 3-4-5-6. In contrast, the other ones are excluded to respect the constraints imposed by the company policies in favor of components characterized by lower BC and lower resource requirements. For instance, in the "case-base", i.e., Experiment 1, the drainer (C18) is replaced or maintained together with the controller (that is, the one effectively experiencing the failure), the coupling (C2), and the lighting (C25). As previously stated, data provide the support for the execution of such interventions, even though there might not seem to be any actual relations, furtherly highlighting the benefits driven by the implementation of the approach. Indeed, the reliability of the plant is ensured through the adoption of the proposed framework since the domino effect among failures frequently occurring together is limited by anticipating the maintenance of critical components.

Chapter 5.

Research approach application to the extension of failure analysis

Sections 5.1 and 5.2 provide the implementation of the proposed general framework from a different point of view. Indeed, data related to the failure analysis traditionally carried out by the organization are used, and decision models based on the Social Network Analysis are proposed. Two different case studies are developed using this methodology.

5.1 Data-driven extension of failure analysis: the case study of an on-shore/off-shore platform

5.1.1 Data gathering and management

The input data are the equipment list and their characteristics, in which each item represents the plant components, or parts of them, whose maintenance policies are under investigation. Moreover, in order to evaluate the frequency of the failure modes, it is necessary to consult the maintenance policies currently implemented by the company, as well as the historical data of previous failures of the plant components and reliability databases (OREDA, EIREDA, or IEEE). The failure rate λ (i.e., number of failures per hour) for each failure mode of each equipment is retrieved from them. Moreover, on-field reports are integrated into the historical data recorded in the information systems to create a spectrum of analysis as accurate as possible. In Table 29, a summary of the units, a brief description and the number of items analyzed, and the related failure modes are reported. In all, 501 items and 31 different failure modes referring to the 15 units are analyzed in this application.

Table 29 Summary of the units, items, and the related failure modes

Unit	Description	Items monitored	Failure modes
UNIT 100	Oil&Gas Production Wellhead	108	232
UNIT 130	Production Flowlines	6	27
UNIT 190	Launching Trap	23	15
UNIT 200	Oil Production Separation	10	63
UNIT 220	Crude Oil Transport	22	32
UNIT 230	Flare and Blow-Down	12	58
UNIT 360	Gas Compression	26	107
UNIT 390	Glycol Injection Pumps	44	6
UNIT 420	Fuel Gas	55	137
UNIT 450O	Oil Wellhead Control Panel	24	58
UNIT 450G	Gas Wellhead Control Panel	24	58
UNIT 460	Air Compressor, Air Dryer and Filters and Receivers	88	199
UNIT 470	Main Power Generation System	24	46
UNIT 500	Sea Water System	20	42
UNIT 550	Closed Drain System	15	44

5.1.2 Data analytics

5.1.2.1 Preliminary analysis

During the preliminary analysis of the analytics layer, the data collected in the previous one are analyzed with traditional methods in order to extract valuable information about failure mode, risk events, and maintenance policies. The FMECA is performed at this stage, following the recommendation provided in the US Military Standard [196,197]. A bottom-up approach is adopted for its execution, breaking the system under investigation down to identify its elementary components (sub-systems or parts) that are separately analyzed. The objective of the breaking-down is, indeed, to provide an accurate description of the failure modes, effects, and the criticality itself. The approach followed to carry out the FMECA is collaborative. It involves discussing the main features of the system among the interdisciplinary groups of people engaged in the system's functioning at different levels (e.g., O&M engineers, managers, technicians, on-field personnel). In this way, several perspectives are taken into account, and a complete understanding of the system is guaranteed. Additionally, due to the contributions provided by the multi-disciplinary team involved in the FMECA, it is possible to limit the subjectivity characterizing each role and to avoid the related uncertainty. In case of disturbances on data, it is possible to clear them during this phase since the team has to compare the records on the information system and on-field reports. The output of this step is a dataset containing the components of the system, the potential failure modes, effects, frequency, severity, and a measure of the criticality; this information is used to define actions and follow-ups. Indeed, the identification of component criticality is aimed to guide the actions to undertake in order to anticipate failures or correct them. In this way, the components that should be strictly monitored are determined so that specific inspection policy can be defined, as well as roadmaps for the corrective interventions. These analyses are made by maintenance experts that are already aware of company policies in order to make the action and follow-ups feasible and coherent with the company maintenance strategy. These outcomes should be included in the dataset that will be analyzed in the following steps of the procedure in order to enlarge the possible identifiable connections. The FMECA is performed every year to reconsider the outcomes on the basis of the events verified during this time interval. In this way, the framework is always updated with the real behavior of the system.

A selection of a panel of experts made up of professional figures taking part in the FMECA process has been developed not to categorize the study. It is necessary to gather data from the on-field technicians who have a direct view of the processes and failures, as well as O&M managers who consider the process from a broader perspective, taking into account both the technical aspects related to single equipment and the main features of the whole process. Other experts of the maintenance team are necessary to globally extend the study to the field of machinery, civil structure, material, design, and process. The participants in the discussion panel, in yearly meetings, define how the equipment can fail by identifying the failure modes, their effects, and their criticality. Firstly, the functions of the equipment have to be identified in order to specify how failures can occur. It is then necessary to establish how often a failure event occurs on the specific component for each failure mode.

In this work, the attribution of the frequency class is performed according to the parameters reported in Table 30.

Table 30 Frequency class assigned to each component failure modes.

Annual Frequency Class (AFC)	Description	Increasing Annual Frequency – IAF (number of events per year)	Meaning
0	Practical non-credible occurrence	IAF $<10^{-6}$	Could happen
A	Rare occurrence	$10^{-6} \leq$ IAF $<10^{-4}$	Reported in this industry
B	Unlikely occurrence	$10^{-4} \leq$ IAF $<10^{-3}$	Occurred at least once in the company
C	Credible occurrence	$10^{-3} \leq$ IAF $<10^{-1}$	Occurred several times in the company
D	Probable occurrence	$10^{-1} \leq$ IAF <1	Happens several times/year in the company
E	Frequent occurrence	$1\leq$ IAF	Happens several times/year in one location

Five levels – 0 for a low level, 4 for a high one - are used to classify the severity of the effects of a failure in a qualitative way. In particular, three different categories are evaluated when rating the severity of an effect, i.e., safety, environmental, and assets (Table 31). The severity assignment does not follow objective criteria, i.e., measurable, but is defined by the panel of experts: for example, concerning the severity of the failure on production capacity (asset), Severity 1 can be assigned to equipment stoppage, Severity 2 to quality deviations of production parameters of the output, Severity 3 or 4 to the propagation of the effects, e.g., plant stoppage. The attribution of the criticality corresponding to each of the three aspects is performed semi-quantitatively, using the matrix reported in Table 31. The Criticality indexes are calculated by multiplying the severity of each effect category (safety, environmental, asset) by the annual frequency class:

- Safety Criticality Index, ICS = Severity Safety * Annual frequency class
- Production/Asset Criticality index, ICA = Severity Production/Asset * Annual frequency class

- Environment Criticality index, ICE = Severity Environment * Annual frequency class

The frequency class and severity of effects are then inserted in the risk matrix (Table 31) to be compared with the acceptability criteria: the risk matrix is applied to define the critical elements through thresholds or criteria of "acceptance" of the criticality.

Table 31 Risk matrix used in this case study

Severity				Annual Frequency Class					
	Safety	Environment	Asset	0	A	B	C	D	E
0	Slight health effect	Slight damage	Slight effect	C3: Continuous Improvement					
1	Minor health effect	Minor effect	Minor damage						
2	Major health effect	Local effect	Local damage			C2: Risk Reduction Measures		C1: Intolerable Risk	
3	One fatality	Major effect	Major damage						
4	Multiple fatalities	Extensive effect	Extensive damage						

Within the risk matrix, it is possible to read the three criticalities of the failure event, and therefore assess its acceptability threshold.

- C3: the risk is tolerable; no further impact reduction measures are necessary, but it is sufficient to monitor performance and manage it for continuous improvement.
- C2: The risk is intolerable; the risk will become tolerable after appropriate control measures have been identified and implemented.
- C1: The risk is intolerable; further impact reduction measures are needed.

The belonging of each item to one of the three categories is determined considering the AFC and discussing the severity of the effects by the multi-disciplinary team. In case of uncertainty or disagreement, the worst criticality is assigned. In addition, it is possible to calculate the Overall Critical Failure Mode (CFM), representing the overall impact of a failure mode on a specific item, considering all the types of effects it has; it is calculated as the worst of the three effects (i.e., safety, environmental and assets) of that failure mode. To identify the Criticality of the Item (CI), regardless of the specific failure mode, it is possible to calculate the following parameter:

$$CI = Min\ (CFM) \tag{31}$$

In other words, the CI is defined by considering the criticality of the most critical failure mode. In appendix C, an excerpt of the FMECA document regarding two different items is reported to provide an example of the basic information noted. In the current analysis, 1196 failure modes recorded in the previous five years concerning 501 items have been analyzed. For each failure mode, it is specified whether a corrective, cyclical (time-based) or condition-based maintenance policy is adopted. Remarkably, the company relies on a cyclical maintenance policy for preventing the occurrence of the majority of failure modes (693). The condition-based approach, instead, is adopted only in a low number of cases (34). The most critical plant section is Unit 100, which shows 248 failure modes.

5.1.2.2 Association Rule Mining

In the development of the data analytics layer, a top-down approach is followed, firstly analyzing the general situation and then exploring further the areas of interest, according to this sequence:

- STEP 1 - Relationship analysis among failure modes, effects, and criticalities of the whole plant: this analysis allows company managers to have an overview of failure mode and effects on the overall plant to highlight the riskiest components of the plant and the related measures of occurrence and severity.
- STEP 2 - Relationships among failure modes and effects of the single unit: more specific analysis of the riskiest plant unit is necessary. In fact, the individual functional units may behave differently to the entire production system. This analysis aims at highlighting the domino effect among failure modes within the unit, focusing on details that may be unclear with the general representation.
- STEP 3 - Relationships among effects, maintenance tasks, and policies on every unit. The decisions made in terms of maintenance policies are included in the input dataset for the ARM, allowing the analysis of the repetitive relationships among inputs, i.e., the attributes of the FMECA and outputs (e.g., maintenance policies

adopted, criticality levels). The objective is the assessment of the maintenance policies' effectiveness to identify best practices.

The thresholds' setting to define the interestingness of the rules should take into account two different factors. On the one hand, the size of the output: low support and confidence thresholds can cause the generation of many rules, possibly exponential with regards to the input size, yielding several uninteresting relationships. On the other hand, setting high thresholds can cause the loss of interesting patterns, leading to ineffective dataset exploitation. As noted in [198], there is no absolute optimal measure to set the thresholds; it strictly depends on the application case. For this reason, the expertise of the panel of experts is required to perform this process consistently. Specifically, for each case represented in the case study, the support threshold min_sup is initially set to a very low value (e.g., 0.01) and incremented to display the most statistically significant rules.

The analysis of the total rules at the same time is somewhat unrealistic, even though selecting appropriate thresholds means that only the most relevant are considered. In this application, the criticality of the items considered is used as a prioritization metric. When a failure mode occurs, corrective interventions or control and monitoring actions are performed according to the guidelines indicated in Table 32, prioritizing the items characterized by a higher criticality (CI). In case of more items belonging to the same category, if the interventions cannot be performed simultaneously, the support is used as a discriminant (the higher the support, the higher the priority of the item over the concurrent ones).

Table 32 Time scale for performing corrective actions or monitoring

Items criticality rate	**Policy adopted**
Criticality Index = 1	The analysis must be started immediately
Criticality Index = 2	The analysis should be started within 24-48 hours
Criticality Index = 3	The analysis should be started within a week

Before starting the actual analysis of the dataset, a comparison among different algorithms is performed. In this way, it is ensured that the more efficient one is adopted. According to the existing literature (e.g., [196,197]), the selected algorithms are FP-Growth and Apriori algorithm. The ARM is performed, and the execution time required for extracting the ARs, varying the minimum support threshold, is compared. Each experiment is carried out ten times using the software RapidMiner Studio on a two-core processor at 2.70GHz. The processing times reported in Table 35 are the mean values obtained for each case. Despite both algorithms are processed in a reasonable time, the FP-growth results in being the most effective. Hence, it is the one adopted for the case study.

Table 33 Average processing time for the FP-growth and Apriori algorithms, varying the minimum support threshold.

	Min_supp				
	0.010	**0.05**	**0.100**	**0.200**	**0.500**
FP-growth [sec]	0.287	0.276	0.275	0.275	0.275
Apriori [sec]	8.721	8.717	8.601	8.601	6.686

5.1.2.3 Decision support model based on Social Network Analysis

STEP 1: Relationships among failure modes and effects of the whole plant

The network reported in Figure 16 is comprehensive of the principal relationships among failure modes and effects occurring across the entire production system, as well as the annual frequency class of the failure modes and the severity levels assessed in terms of asset, environment, and safety. In addition, in order to identify the most critical units and items, this information is reported in the network. The ARs' number and the SN characteristics obtained in Step 1 are shown in Table 34, varying the minimum support threshold, which is initially set to 0.01, while the minimum confidence threshold is set to 0.10, leading to the extraction of 1648 association rules. In terms of strength of the rules (i.e., confidence), there is no limitation caused by the increase of the support from 0.01 to 0.05. However, the number of communities does not change incrementing the min_sup since the attributes involved in the analysis at this first stage are numerous; hence, none of the nodes are separated from the other ones. For the sake of clarity, Figure 16 shows the network obtained by increasing the support to 0.05 and the minimum confidence to 0.25 (370 associations referring to 44 nodes). The number of ARs selected in this case allows both an understandable visual representation and, in parallel, constitutes a considerable amount of relationships.

Table 34 ARs' number and SN characteristics varying the minimum support threshold

ARs and SN characteristics	min_sup				
	0.01	**0.05**	**0.1**	**0.2**	**0.5**
# of ARs	1648	370	179	75	20
Minimum confidence in the selected range of ARs	0.10	0.10	0.11	0.22	0.75
# of communities	1	1	1	1	1
Out-Degree (OD) range	[0.0; 11.0]	[1.0; 8.6]	[1.9; 7.0]	[0.9;6.6]	[6.9; 7.2]

In the network of Figure 16, the thickness of the edges represents the confidence of the rule, while the dimension and the color of the node indicate its out-degree level – i.e., the weighted sum of its outgoing edges: the smaller the node, the lower the OD is; similarly, the colors of the nodes indicate different OD values, following a pink-white-green scale: pink nodes are characterized by low OD, white by a medium OD and, the greener they become, the higher the OD is. Since the graph is directed, there are reciprocal relationships among the nodes. Figure 16 highlights that the most critical functional unit is UNIT100. Indeed, it is represented by the biggest and green node among the units, indicating that its out-degree is elevated (OD=6.64). As previously noticed, a high OD indicates a strong influence of the node across the network; hence, further analysis of the possible chain effects in these areas is worthy of investigation. Among the failure modes, the ones characterized by the highest out-degree is "Fail to close on demand" (OD=7.67) and "Fail to regulate" (OD=7.34). Both the failure modes are linked to the item "Flow control valve": the association rules "Failure Mode = Fail to regulate"→ "Item = Flow Control Valve" and "Failure Mode = Fail to close on demand"→ "Item = Flow Control Valve" are characterized by support of 0.05 and confidence of 0.65.

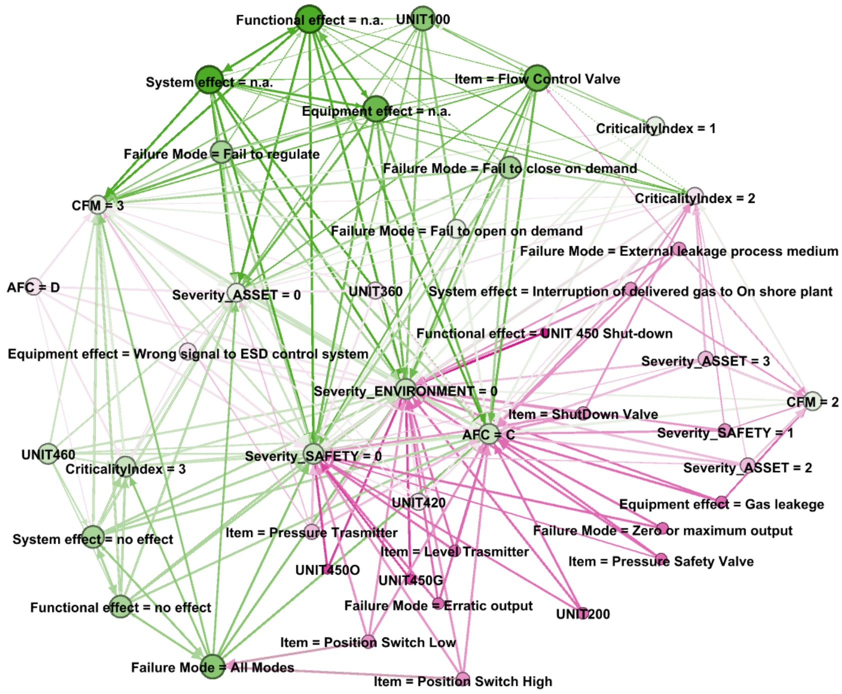

Figure 16 Relationships among items, criticalities, failure modes and effects of the whole plant; node dimension is proportional to the OD (the bigger the node, the higher the OD). Color scale, pick-white-green, represents growing OD levels.

STEP 2: Relationships among items, failure modes, and effects of every unit

The analysis carried out for the whole plant has been repeated for every single unit to develop further the knowledge of every single part of the plant. This analysis aims at highlighting the domino effect among failure modes within the unit. For example, the following SN (Figure 17) reports the relationships regarding failure modes and the effects of the Oil&Gas production wellhead unit (unit 100). In order to provide a clear representation of the SNs, following the same procedure deployed in Table 34, the minimum support threshold is set to 0.10.

Three communities of nodes are highlighted in the Unit 100 case (Figure 17a, b, and c). The first community of nodes reported in Figure 17a, “Item = Pressure indicator” ↔

“Equipment effect = wrong value indication”, indicates two events only related among them. In other words, both the rule “Item = Pressure indicator” → “Equipment effect = wrong value indication” and “Equipment effect = wrong value indication”→ “Item = Pressure indicator” are defined. The support and the confidence of both the rules are, respectively, 0.13 and 1.00: specifically, the two attribute-value relationships appear together in 13% of the dataset instances and, since the confidence is 100% for both sides of the rule, in every instance where “Item = Pressure indicator”, there is also “Equipment effect = wrong value indication”, and vice versa.

Similarly, in Figure 17b, the community is composed of four nodes: one is representative of all the failure modes (“Failure mode = all modes”), while the other three nodes regard the related effects. Specifically, at the equipment level, the effect that verifies in the case of every failure mode is a wrong signal recorded by the distributed computer system (“Equipment effect = Wrong signal to DCS”). In contrast, no effect is related to such a failure mode at the system and functional level. The rules describing this relationship among the failure mode and the three effects are characterized by the highest confidence (0.81): “Failure mode = all modes” → “Equipment effect = wrong signal to DCS”; “Failure mode = all modes” → “System effect = no effect”; “Failure mode = all modes” → “Functional effect = no effect”.

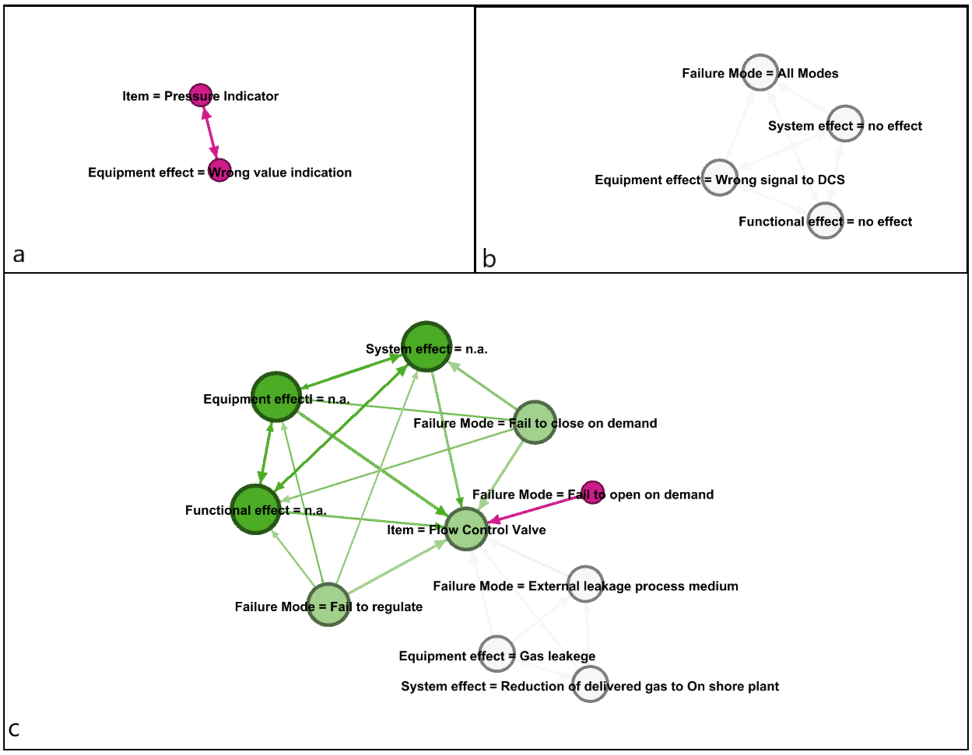

Figure 17 Relationships among items, failure modes, and effects of unit 100; node dimension is proportional to the OD (the bigger the node, the higher the OD). Color scale, pick-white-green, represents growing OD levels.

The community of nodes reported in Figure 17c is more complex: four failure modes are represented, as well as two effects at each level and an item. The node "Item = Flow control valve" has an important function. Indeed, it represents the joint between two opposite sides of the community. If major attention is dedicated to this item indeed, it is possible to control all the failure modes and effects related to it. For example, if the failure mode is "External leakage process medium", then in 65% of cases, there is a gas leakage at the equipment level ("Failure mode = leakage process medium" → "Equipment effect = gas leakage") and a reduction of delivered gas to the onshore plant ("Failure mode = leakage process medium" → "System effect = Reduction of delivered gas to Onshore plant") occurs with the same probability (confidence = 0.65).

Remarkably, in the SN reported in Figure 17c, there are two different categories of the effects: on the upper side of the SN, we can see the equipment, functional, and system effects labeled with "n.a.". This means that these effects are not assessable. Thus it is not possible to monitor or prevent them. On the other hand, effects like "Equipment effect = gas leakage" and "System effect = reduction of delivered gas to the onshore plant" can be critical in terms of safety or asset integrity. Hence, these aspects will be analyzed further in the third step of the data-analytics phase.

STEP 3: Relationships among effects, maintenance tasks, and policies on Unit 100

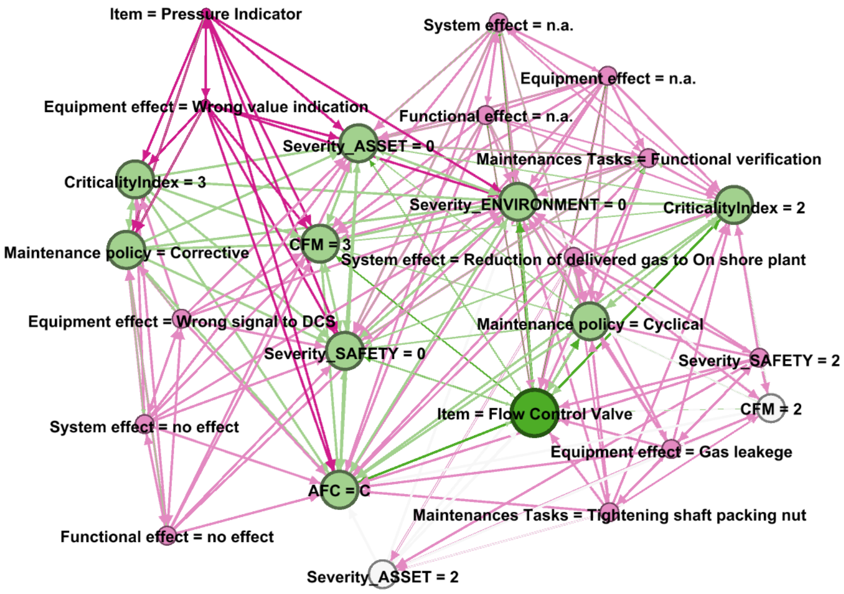

Figure 18 Relationships among effects, maintenance tasks, and policies of Unit 100; node dimension is proportional to the OD (the bigger the node, the higher the OD). Color scale, pick-white-green, represents growing OD levels.

The relationships among effects, maintenance tasks, and policies are analyzed to assess the effectiveness of the maintenance policies adopted and to identify best practices. In Figure

18, the interrelations represented regard maintenance tasks and policies, items, effects (at all levels), criticality indexes, and severity of Unit 100. In this case, the minimum support threshold is set to 0.1, while the minimum confidence at 0.20: 360 rules are presented in the SN of Figure 18; according to the procedure deployed in Table 34, these values represent the most appropriate trade-offs. It is also is pointed out how the node "Item = Flow control valve" should be considered for its influence across the SN. The severity impact of a failure on this item is always null for the environmental aspects ("Item = Flow control valve"→Severity_ENVIRONMENT = 0, confidence = 1.00), while in terms of asset, the following rules are defined:

- "Item = Flow control valve"→"Severity_ASSET = 0", confidence =0.72,
- "Item = Flow control valve"→"Severity_ASSET = 2", confidence =0.28.

Generally, the flow control valve is associated with a Criticality index = 2 (confidence = 1.00). Thus, it can be assumed that there is a need for measures aiming at reducing the impact of the effects related to this item.

From the SN, it is clear that when no effect is shown at a system and functional level, but only at an equipment level, the maintenance policy adopted is still corrective, as testified by the following rules:

- System effect = no effect → Maintenance policy = Corrective, confidence = 1.00;
- Functional effect = no effect → Maintenance policy = Corrective, confidence = 1.00;
- Equipment effect = Wrong signal to DCS→ Maintenance policy = Corrective, confidence = 1.00;
- Equipment effect = Wrong value indication → Maintenance policy = Corrective, confidence = 1.00;

However, when the maintenance policy defined for a specific failure mode is corrective, the effect related to it could also be non-assessable (n.a.) both at a system and functional level (Maintenance policy = Corrective → System effect = n.a., confidence = 0.21; Maintenance policy = Corrective → Functional effect = n.a., confidence = 0.21). In this case, the functional verification is foreseen with different percentages depending on the level of the effect (system, functional, or equipment level):

- In case of a non-assessable effect at the equipment level, the functional verification is always required (Equipment effect = n.a. → Maintenance task = Functional verification, confidence = 1.00);
- In case of a non-assessable effect at the functional level, the functional verification is required in 81% of the situations (Functional effect = n.a. → Maintenance task = Functional verification, confidence = 0.81);
- In case of a non-assessable effect at the system level, the functional verification is required in 83% of cases (System effect = n.a. → Maintenance task = Functional verification, confidence = 0.83).

Several effects are related to a time-based preventive maintenance policy. At the equipment level, both gas leakage and no gas flow are prevented through cyclical maintenance interventions ("Equipment effect = Gas leakage" → "Maintenance policy = Cyclical", confidence = 1.00; "Equipment effect = No gas flow" → "Maintenance policy = Cyclical", confidence = 1.00), as well as the reduction or absence of gas flow at a functional ("Functional effect = Reduction of delivered gas to Unit 190 Launching Trap" → "Maintenance policy = Cyclical", confidence = 1.00; "Functional effect = no gas flow to Unit 190 Launching Trap" → "Maintenance policy = Cyclical", confidence = 1.00) and system level ("System effect = Reduction of delivered gas to On shore plant" → "Maintenance policy = Cyclical", confidence = 1.00; "System effect = no gas flow to onshore plant" → "Maintenance policy = Cyclical", confidence = 1.00).

5.1.3 Control layer

5.1.3.1 An example of a decision-making step based on SNA

The analysis of the SN is useful for identifying the chain of nodes that leads to high-risk events and supports during the control phase of the decision model implementation. For example, in the case shown below, a potential misleading maintenance policy application leads to high-risk events. The knowledge of this sequence of nodes helps to break this chain with the aim of reducing the risk level of the processes under investigation. In Figure 19, for example, the SN showing the associations among items, equipment effects, maintenance policies, and criticality indexes is reported. The two biggest nodes of the network (i.e., having the highest OD values, so the strongest influence of the node across the network) are "Criticality Index = 1" and "Maintenance policy = Corrective". Moreover, there is an association rule between these nodes. Since the Criticality index = 1 indicates the highest risk for the plant, further investigation is needed to justify the association with a corrective maintenance policy. Specifically, the item associated with these values is the pressure indicator.

If a wrong indication does not affect the functional and system levels in some cases, sometimes it implies a wrong setting on the downstream machinery and uncontrolled pressure values on the downstream plant. A modification in the maintenance policy is then defined to avoid the occurrence of cascade effects: periodical inspection routes are established to control the physical integrity of the pressure indicators and specific alerts are set on the management system to advise the operators in case of unexpected pressure values.

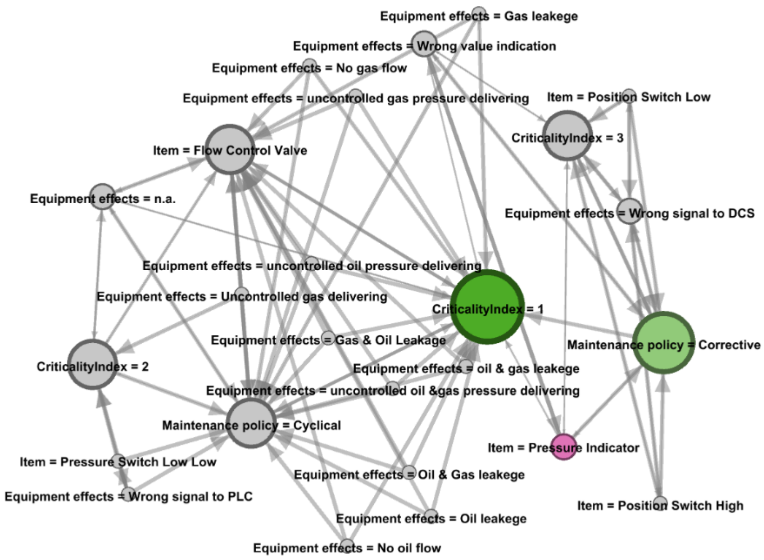

Figure 19 Relationships among effects, maintenance policies and criticality indexes; node dimension is proportional to the OD (the bigger the node, the higher the OD).

5.1.3.2 Procedure validation

The mining of the ARs on a well-known data-analytics platform ensures that the calculations are performed correctly. However, there is a three-step validation of the results before and during their implementation.

1. The first check is performed after the FMECA is carried out. Indeed, the objective is to avoid missing data at the first instance and prepare a starting dataset as much complete as possible. For this reason, the FMECA is carried out by multi-disciplinary teams, and the results have to be validated by every one of them, as well as by the plant chief. The document is updated every year in order to be adjourned with the latest events and keep up with modifications of the plant.
2. The procedure implemented on the selected software (i.e., RapidMiner, in the case study) is double-checked by the engineers in charge of this step. Any possible unnoticed error can be fixed before the actual implementation, thanks to this control.
3. During the first stages of the implementation, the failure events occurring are compared with the rules extracted to verify if the probability distribution of their occurrence and effects reflects the one described by the support and confidence of the ARs mined. The beginning of the procedure implementation dates back to April 2020. Data concerning the following eight months are considered: in 87.5%, the two compared cases correspond, showing an acceptable accuracy of the proposed approach.

The entire process is then reiterated every year once the FMECA is updated.

5.2 Data-driven extension of failure analysis: the case study of a hydro-electrical power plant

The proposed approach is applied to a Brazilian hydroelectric power plant (HPP). It is equipped with three hydro generators type Kaplan units, which operate at 166.25 MW. Kaplan hydro generators units can work where a small head of water is involved; the turbines are applied in sites having a head range of 2–40 m. Since the angles of their blades can be modified to adapt to the water flow, Kaplan turbines can also work efficiently at a broader range of water head, allowing for variations in the dam's water level. Three principal systems compose the hydro-generator Kaplan unit: speed governor, turbine system, and axis. In all, 152 components have been identified during the FMEA analysis of the HPP; thus, they are treated in the failure analysis.

5.2.1 Data collection and management

The hydroelectric industry requires a high level of availability and reliability. The FMEA is regularly carried out on the system to identify components' criticality and prioritize their maintenance. In this way, the risk involved in the production process is monitored; however, further knowledge of the HPP can be extracted by implementing the proposed approach. The FMEA is performed following the US Military Standard's recommendation, adopting a bottom-up approach: the system under investigation is broken down to analyze its elementary components separately. Through the breaking-down, the objective is to provide an accurate description of the failure modes, effects, and impact on safety, environment, and assets. The main advantage of taking the FMEA as a starting point is that several perspectives are questioned so that a complete understanding of the potential failures and effects is achieved. Additionally, due to the multi-disciplinary team's contributions in the FMEA, it is possible to limit the subjective bias related to each role and avoid the related uncertainty. Finally, in carrying out the FMEA, a dataset containing the system's equipment under investigation, the potential failure modes, and the associated effects are created and can be analyzed through the association rule mining. Additional information can be added, such as the mean time to repair (MTTR) or the failure mode criticalities. Starting from the FMEA has different advantages: on the one hand, it allows the company to improve the plant's knowledge further. On the other hand, the data-driven analysis is carried out basing on the expertise of the multi-functional team that is usually charged with deploying the FMEA—so benefiting from different and wide-ranges perspectives. A collaborative approach is adopted to deploy the FMEA. The HPP's main features are discussed by interdisciplinary groups of people involved in the system's operations at different levels (e.g., maintenance engineers, managers, on-field technical personnel). The dataset structure used as a starting point for the data-driven analysis is reported in Table 35. Specifically, data refer to the FMEA traditionally carried out by the company and regard:

1. System: one of the three main systems composing the HPP;
2. Name: one of the 152 components relevant for the study;
3. PFM: potential failure mode occurring on the component;
4. Main functions: effect of the PFM on the main functionality of the component;
5. FR: the failure rate of the component (it can be actual if the FM has already occurred or theoretical if the FM is potential);
6. MTTR: the mean time to repair, expressed in hours;
7. SAI: the impact of the FM occurrence on the availability of the system;
8. IOP: the impact of the FM occurrence on people;
9. EI: the impact of the FM occurrence on the environment.

Attributes 7–9 are evaluated by the multi-disciplinary team members responsible for performing the FMEA on a 1:9 scale.

Table 35 Structure of the Failure Mode and Effects Analysis (FMEA) dataset.

System	Name	PFM	Main Functions	FR	MTTR	SAI	IOP	EI
AXIS	Generator Shaft	Break	Provide rotation for electricity generation	0.000001	168	9	7	1

5.2.2 Data analytics

The second phase of the work regards the analytics execution, considering the FMEA dataset as a starting point.

5.2.2.1 Preliminary analysis

During the preliminary analysis, the attributes identified by the FMEA are taken into account. They are selected to consider only the ones relevant for the aim of the analysis.

The dataset, whose structure is presented in Table 35, comprises 432 transactions (rows of the dataset). The components analyzed are 152, while the distinct PFMs are 113: this means that the same failure mode can affect different components. The dimension of the dataset and the aim of the application suggest that to extract all the possible association rules worthy of investigation and not limit their extraction: null support and confidence thresholds are set (min_sup = 0; min_conf = 0).

5.2.2.2 Association Rule Mining

The second step of the analytics phase requires defining the relevant associations among the events extracted from the FMEA dataset. Specifically, relevant information may regard the failure modes frequently occurring on different items or the same effects deriving from different failure modes.

This exploratory analysis aims to extend the existing knowledge of the analyzed system. The larger the dataset, the more complex the data analysis is: in this sense, data-driven techniques overcome the traditional statistical ones, which are no longer able to provide useful insights alone, without the need for formulating hypotheses. Hence, the ARM selection represents a valid alternative [199] since it allows both the simultaneous analysis of a large amount of data and an intuitive results interpretation [72] due to the structure of the outcomes. In this sense, it is also easier to involve the non-expert of the data analytics field to understand and implement the insights obtained in the data-driven analysis. The applications of the ARM are widespread and can be found in different fields, such as the operations and production-related ones; however, the first one regards the extraction of hidden patterns from large datasets for marketing scopes [176].

5.2.2.3 Data-driven decision model based on Social Network Analysis

The ARs among all the attributes explained in Table 35 are mined. In all, 4147 associations among 362 itemsets are extracted and are represented using the open-source software Gephi. To limit the study to the relevant associations and to be able to analyze them properly, the following procedure is applied:

1. Create the SN using all the ARs;
2. Determine the most interesting node based on the OD;

3. Filter the ARs and create more specific SNs, limiting the analysis to the nodes considered more relevant;
4. Formalize the information extracted.

The turbine node has the highest OD (4.645) if compared to the axis (4.301) and the speed governor (4.419); hence, the ARs referring to this portion of the HPP is extracted. Therefore, the ARs referring to the turbine are extracted to focus the analysis on this branch of the system primarily. This filter leads to the mining of 1248 ARs (127 itemsets). To focus on the most relevant portions of the network, the attributes Item, PFM, and Functions are taken into account, creating an SN composed of 102 nodes and 308 arcs.

Interestingly, as reported in Figure 20, 13 communities of nodes originated, considering these ARs.

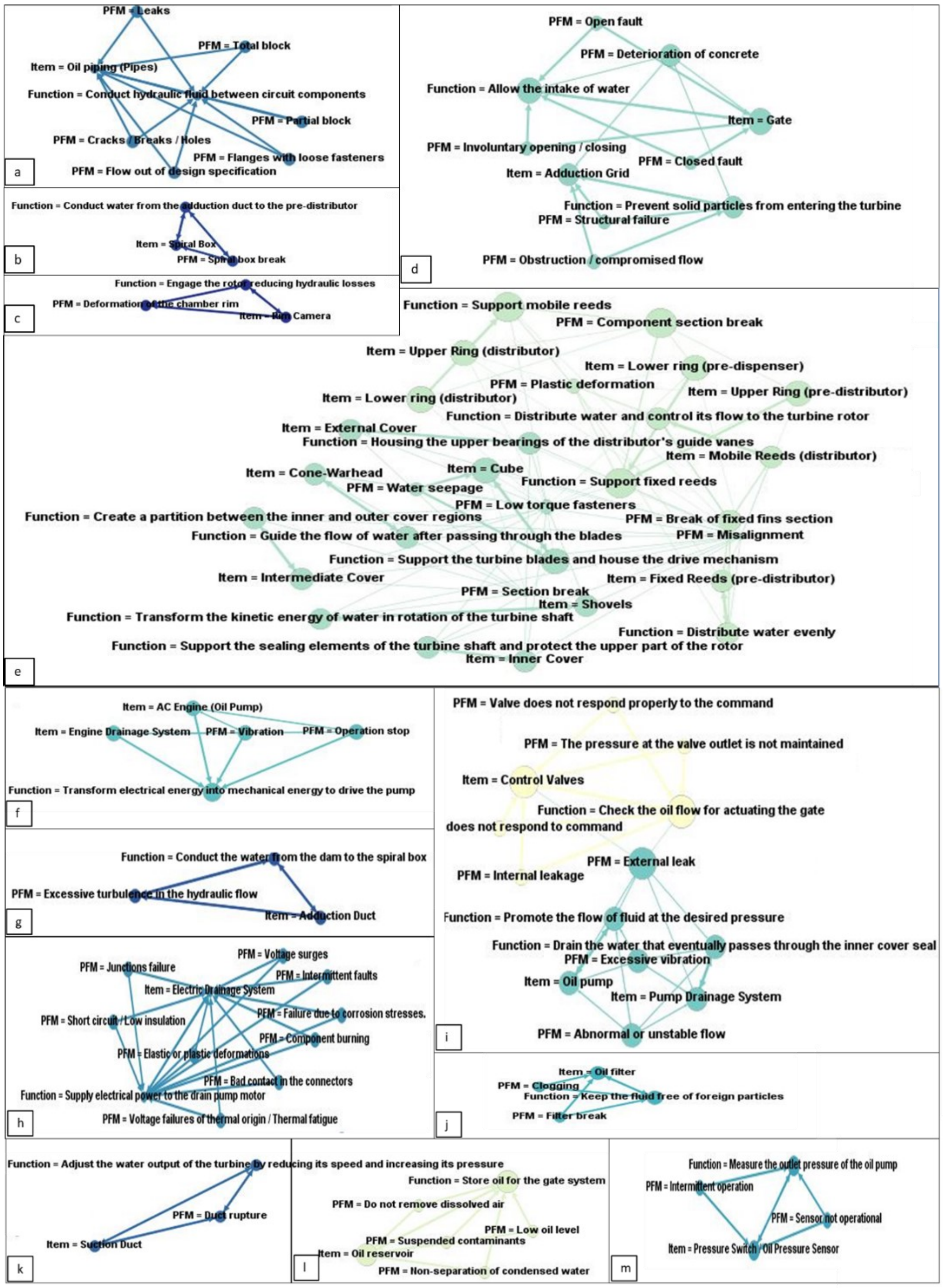

Figure 20 Social Network representing the relationships between Item, Potential Failure Mode (PFM) and Functions of the turbine system: (a–m); represent the thirteen communities of nodes originated from the analysis

This structure indicates that not all the nodes are connected among the others, limiting the potentiality of spreading their occurrence across the network. Indeed, if the nodes are not connected among them, there is no relation among the events represented by such nodes. This aspect limits the attention that the maintenance managers have to pay to the so-called domino effect. In particular, eight networks simply represent the connection among the item, the related function, and failure modes: this information is not new since it can be derived from the FMEA with no reason for extending the analysis through the data-driven framework. Indeed, the proposed approach aims to extend the current body of knowledge on the existing plant by extracting previously unknown relationships. On the contrary, three networks (Figure 20d, e, i) display relevant and previously unknown relationships. Indeed. These relationships involve more than one item and several PFM, supporting the maintenance managers in identifying potential combined inspections and actions to anticipate the potential failures across the plant.

For example, in Figure 21a—which deploys Figure 20i in detail, it can be noticed that the node PFM = External leak acts as a bridge among the two portions of the network: indeed, its BC is the highest in the SN (74.67). In this sense, the occurrence of an external leak may have an impact on control valves, the oil pump, and the pump drainage system, as evidenced in Table 36. The confidence associated with the three rules (PFM = external leak → Item = pump drainage system; PFM = external leak → Item = oil pump; PFM = external leak → Item = control valves) is 0.333 since it is equiprobable that, when an external leak occurs, the item is one of those listed. These connections highlight the need for establishing a protocol for the inspection of the item when an external leak occurs. Specifically, such protocol should require the verification of the normal functioning of the items, e.g., the flow of the fluid at the desired pressure (Function = promote the flow of fluid at the desired pressure → Item = Oil pump), the drainage of the water (Function = Drain the water that eventually passes through the inner cover seal → Item = Pump drainage system) and the control of the oil flow (Function = Check the oil flow for actuating the gate → Item = Control valves).

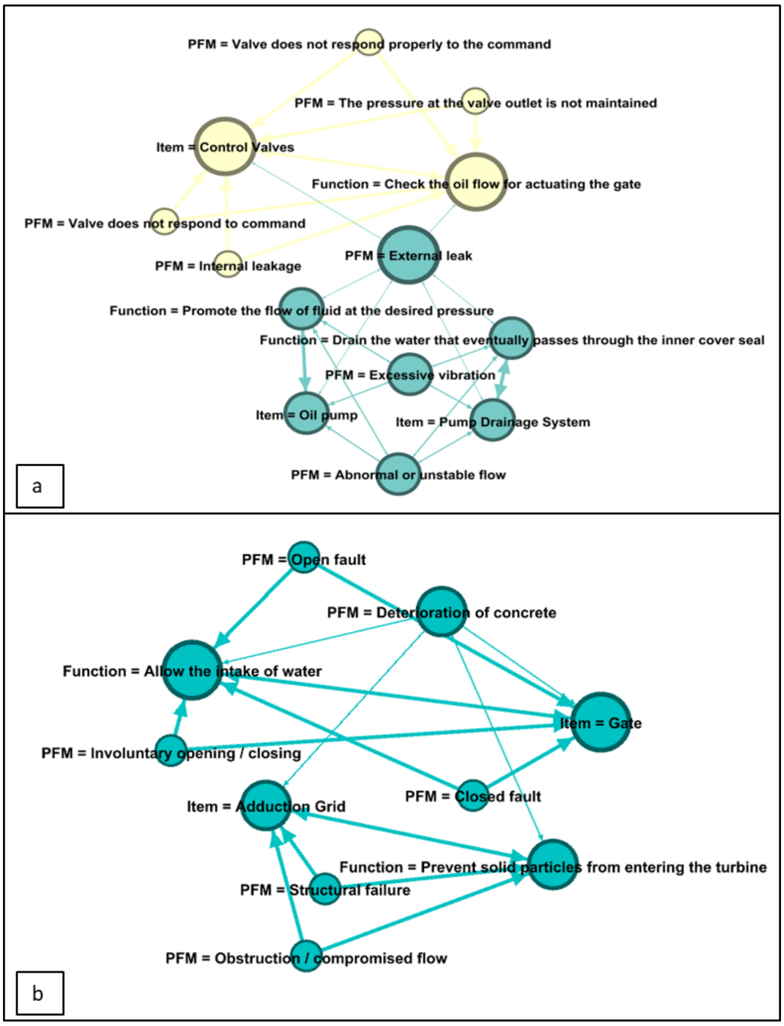

Figure 21 Relationships among potential failure modes, items, and functions. (a) refers to Figure 20i; (b) refers to Figure 20d.

The confidence is 100% for the three cases since each function is associated with a single item. Similarly, in Figure 21b, the communities of nodes reported in Figure 20e are reported. The considerations drawn for Figure 21a can be extended to this community too. Indeed, the two items noted in this network (i.e., gate and adduction grid) share a common potential failure mode (PFM = deterioration of concrete) that acts as a bridge for the two portions of the network. When this failure mode occurs is then essential to check whether both the items are normally functioning or if an intervention is needed. As noticeable from Table 37, when the potential failure mode “deterioration of concrete” occurs, the confidence of 50% indicates that it regards either the gate or the adduction grid (see the first two rules reported in Table 37).

Table 36 Association Rules (ARs) among the PFM, item, and function of the network’s portion reported in Figure 21a

Left-Hand Side	Right-Hand Side	Supp	Conf
PFM = External leak	Item = Pump drainage system	0.011	0.333
PFM = External leak	Item = Oil pump	0.011	0.333
PFM = External leak	Item = Control valves	0.011	0.333
PFM = External leak	Function = Promote the flow of fluid at the desired pressure	0.011	0.333
PFM = External leak	Function = Drain the water that eventually passes through the inner cover seal	0.011	0.333
PFM = External leak	Function = Check the oil flow for actuating the gate	0.011	0.333
Function = Check the oil flow for actuating the gate	Item = Control valves	0.056	1
Function = Drain the water that eventually passes through the inner cover seal	Item = Pump drainage system	0.033	1
Function = Promote the flow of fluid at the desired pressure	Item = Oil pump	0.033	1

Table 37 Excerpt of the ARs among the PFM, item of the portion of the network reported in Figure 21b.

Left-Hand Side	Right-Hand Side	Supp	Conf
PFM = Deterioration of concrete	Item = Gate	0.011	0.500
PFM = Deterioration of concrete	Item = Adduction grid	0.011	0.500
PFM = Deterioration of concrete	Function = Allow the intake of water	0.011	0.500
PFM = Deterioration of concrete	Function = Prevent solid particles from entering the turbine	0.011	0.500
Item = Gate	PFM = Deterioration of concrete	0.011	0.250
Item = Adduction grid	PFM = Deterioration of concrete	0.011	0.333
Function = Prevent solid particles from entering the turbine	PFM = Deterioration of concrete	0.011	0.333
Function = Allow the intake of water	PFM = Deterioration of concrete	0.011	0.250
Item = Gate	Function = Allow the intake of water	0.044	1.000
Item = Adduction grid	Function = Prevent solid particles from entering the turbine	0.033	1.000
Function = Allow the intake of water	Item = Gate	0.044	1.000
Function = Prevent solid particles from entering the turbine	Item = Adduction grid	0.033	1.000

It is noteworthy to evaluate the impact of a failure on the related items, taking Figure 21a as a reference: the ARs involving the item, the measures of the impacts on people, system availability, and environment are taken into consideration to create the SN reported in Figure 22. According to the experts’ opinion, failures on the three items cause low impact at a system availability (Item = Control Valves → System_Availability_Impact = 1; Item =

Oil pump → System_Availability_Impact = 1; Item = Pump Drainage System → System_Availability_Impact = 1) in all cases, since the confidence associated with these rules is 100%. At an environmental level, instead, the pump drainage system and the oil pump are associated with a value of 3 on the 1:9 scale, while control valves are less critical (1 out of 9). A score of 3 is assigned to the pump drainage system and the control valves, while the oil pump is less critical. These evaluations support the decision-makers in defining which areas should be monitored first after the occurrence of a malfunctioning, prioritizing the interventions in the area where the impact is higher: referring to Figure 22, for example, people safety is the primary concern (hence the first aspect to be investigated) in case of a failure on control valves, while both people and environment have the priority over the impact on system availability in case of a failure of the pump drainage system. In this way, the areas characterized by a higher risk are controlled and repaired firstly.

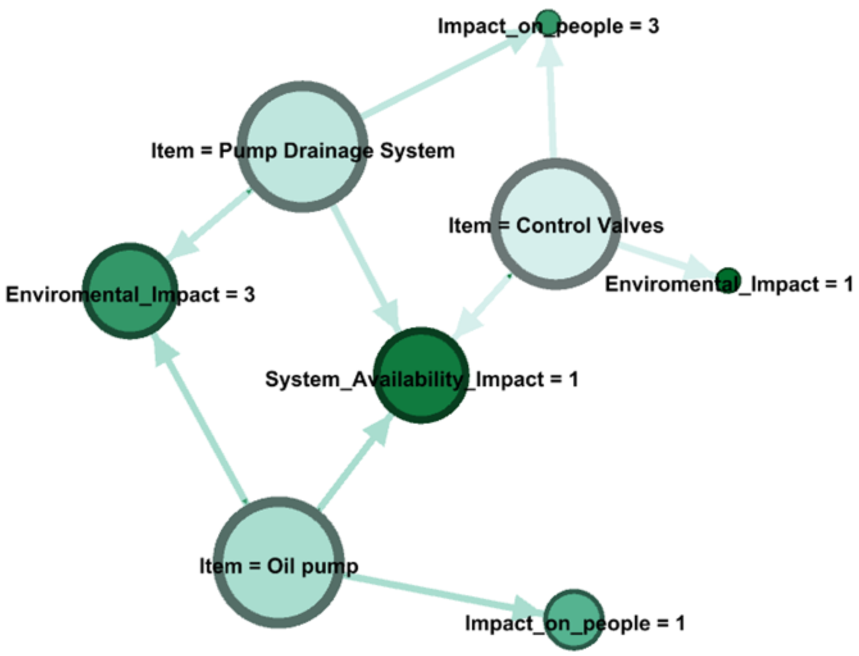

Figure 22 Relationships among items, potential failure modes, impact on safety, environment, and system availability.

5.2.3 Control layer

The last stage of the framework implementation regards the control of the framework implementation. For this specific application, the framework is still being implemented, so no tangible results can be listed. However, the monitored aspects are the following ones:

- The occurrence of the PFM and their effects, to analyze whether any benefit has already been evident;
- The evolution of the network describing the relationships among PFM, effects, and their impact to analyze whether the implementation of the proposed insights is beneficial for the plant.

Chapter 6.

Discussion

This session contains the general remarks, implications and insights related to each of the applications described in chapter 4 and 5. Specifically, the results are discussed critically and the usefulness of the analysis is remarked.

6.1 General remarks on the data-driven decision model based on threshold development

When analyzing refinery processes and maintenance activities, it should be considered that some maintenance interventions can be executed without interrupting the production processes, while others require a complete stoppage of the sub-plant. In the former case, the procedure presented in section 4.1.2.3 can be applied in order to decide whether it is preferable to perform a predictive intervention or if it is better to wait for the actual breakage of the component. In the latter case, instead, together with the cost of the component and the maintenance intervention, the production cost has to be added.

In particular, given the rule A→B, four cases can occur:

- Both A and B can be maintained continuing the production process: depending on the parameters set in the procedure and on the ones characterizing the rule, it is possible to perform a predictive or a corrective intervention. Indeed, these interventions do not have an impact on the production process;
- A can be maintained during the production process, while B requires process interruption: in this case, it might be convenient to wait for the actual breakdown of component B before intervening on it, while A can be replaced both predictively and correctively. If the probability associated with B's breakage in the

chosen timeframe (i.e., $\frac{\#\{B\}}{\#\{D\}} = confidence(B)$) is high, the functioning of B should be monitored utilizing proper KPIs. Moreover, its replacement can be planned, and, based on the probability of breakdown, a new component can be purchased or ordered.

- A requires an interruption of the production process, while B can be replaced while the sub-plant is operating: in this case, waiting for the breakdown that imposes the interruption of the production process is more convenient. Component A should be monitored in order to detect the breakage promptly. In addition, if the probability of the breakdown is high, the substitutive component should be purchased. When the breakdown of component A occurs, B's maintenance can be parallelized or executed after a breakage.
- Both A and B require an interruption of the production process to be maintained: in this case, waiting for the breakdown of one of the two components may result conveniently. Strict monitoring of the components could make fault detection more timely. Preventively purchasing two new components might also shorten the duration of the production interruption. The intervention could be performed on both A and B, possibly parallelizing the interventions or sequencing them in these conditions.

When implementing the proposed maintenance policy, it is also essential to schedule the updating of the extracted rules. Since the refinery continues processing, further stoppages may occur, and additional work orders could be emitted. A clever plan should foresee an updating interval proportional to the one chosen for the analysis. In this way, rules generated for the last timeframe would not be lost.

Moreover, during the update of the rules, maintenance department members can choose from two different strategies:

1) consider a fixed interval – three years in the shown application-, hence progressively exclude the oldest data;
2) progressively enlarge the time interval, adding new data to existing ones.

For instance, in case of modifications to refinery structure or components characteristics, the interval should be shortened. Otherwise, the second alternative should be preferred as it provides much more information and requires a limited processing time.

Expert maintenance members should define the parameters since it represents the core of the maintenance policy. For instance, setting values of support $(\sigma_{rule}, \sigma_{rep})$ and confidence(σ_{conf}) too low would imply the execution of predictive interventions even on components with a low probability of breakdown. On the other hand, if these parameters were too high, a wide range of interventions would be performed after the occurrence of a breakdown, overcoming the aim of the predictive maintenance policy.

Noteworthy, $\sigma_{rule}, \sigma_{rep}, \sigma_{conf}$, as well as the time interval, can be adjusted and modified during the run-time, allowing the adaptation of the maintenance policy to refinery necessity. For example, if it is required to skimp on maintenance $\sigma_{rule}, \sigma_{rep}, \sigma_{conf}$ - or at least some of them - can be scaled up, reducing the number of predictive interventions and maintaining the components only after the actual breakdown. Instead, if it is required to increase the safety of the process, support and confidence threshold could be lowered: in this case, the number of predictive actions would increase, eliminating the need for future corrective interventions.

6.2 General remarks on the data-driven decision model based on mathematical programming

6.2.1 Scalability analysis

As already remarked, the proposed mathematical programming approach does not require higher computational times (i.e., one minute, on average). However, in order to highlight the potentiality of the proposed methodology, a sensitivity analysis on the number of components is carried out. This aims at testing how the number of components given as input may affect the total computational time. For this purpose, 12 different instances are generated using the available real data and reasonably estimating the unavailable ones. The instances have a number of components ranging from 10 to 20,480, so that the i-th instance has 10^{2i-1} components. Each instance is tested five times, and the average computational time is considered for both the ARs extraction and the ILP solution. Mining the ARs of 10 and 20 components requires, on average, 7 s, while for the 40 components, it takes 21 s, on average. In the case of 80 components, 28 s are required on average, while for 160, 320, and 640 components, it takes on average about 33, 41, and 56 s, respectively. Increasing the number of components to 1280, 2560, and 5120, the ARs are extracted, on average in 67, 75, and 84 s, respectively, and in any case, it continues to be a reasonable time. In addition, also the large-sized instances (i.e., with 10,240 and 20,480 components) can be analyzed in a reasonable amount of seconds (i.e., on average, 123 and 180 s, respectively). For what instead concerns the total times required by the ILP model, we can conclude that the instances with 10, 20, 40, 80, 160, 320, 640, and 1280 components are solved in less than 1 s, on average. Moreover, the instances with 2560, 5120, and 10,240 components are solved on average in 1.33 s. Finally, the instance with 20,480 components is solved in about 3.2 s. These experiments remark that the proposed methodology scales well with the number of components.

6.2.2 Update intervals and plant modifications

An issue worthy of discussion regards the databases update. Indeed, during the application of the maintenance policy, other blockages may occur, as well as other maintenance activities, leading to AR changes. The update interval depends on the specific production process: in our case study, an update interval proportional to ΔT defined by members of the maintenance department, i.e., monthly, is a valid option. Moreover, the maintenance policy implementation modifies the correlations among component breakage. Thus, the database should be updated by adding new data gathered within the update interval (e.g., ΔT) and removing the oldest ones (i.e., related to the oldest update interval) to take into account the effect of the policy itself. Parameters setting surely has an impact on the set of components to maintain. For instance, the minimum support threshold could be critical: setting a high min_sup value implies the exclusion of some ARs from the analysis. On the contrary, a value too low may cause an increment of the time to execute the maintenance policy. However, in the current application, the optimal solution is computed in reasonable time also in the cases in which a high number of components is considered. However, if the amount of data stored in the database is significantly higher (e.g., in the case of streaming data), an increment of the min_sup could speed up the analysis.

It is worth noting that any structural modification of the (sub-)plant, as well as any other change in terms of components' characteristics, limits the available data validity. In the process industry, like the oil refinery considered in the case study, this is a reasonable hypothesis since structural modifications are very rare. Otherwise, it is necessary to create a new dataset collecting new data on the (sub-)plant blockages, components breakages, and maintenance activities.

After the experimental campaign carried out on a real-life case study, we can conclude that two are indeed the main limits of the proposed methodology: the number of available data and the fact that the focus is on a sub-plant at a time. In fact, the breakage of a component in a sub-plant could depend on the blockage of upstream sub-plants. Finally, one can observe that the extraction of the ARs depends on the number of components. However, it is de facto performed before the sub-plant is monitored, and therefore, it is a one-time procedure that requires at most 180 s in the case study with 20,480 components. While the computational time required by the optimization solver may increase in the cases with many components, it remains reasonable in any case.

6.3 General remarks on the data-driven decision model based on multi-objective optimization

The proposed approach aims to develop a predictive maintenance policy to identify the optimal set of components to replace for maximizing the plant reliability and minimizing the maximum time spent for repairs. This problem, modeled through bi-objective MILP, supports maintenance managers in implementing an effective maintenance policy, considering not only breakage probabilities but also resource constraints. The current approach is entirely data-driven since all parameters are derived from data on past failure events, are extracted from data on component characteristics, are constants provided by domain experts, or are breakage probabilities estimated by Algorithm 1. In this way, the proposed approach can (1) be applied to different application contexts, and (2) does not require parameter tuning. Moreover, the input parameters are those known at the moment of system stoppage. The proposed approach adapts well to changes in working conditions (e.g., B, Tmax, repair costs, and the number of operators needed for repairing each component). If the working conditions do not change, all available historical data on breakages can be used to estimate probabilities better. Otherwise, the oldest information can be weighted less with respect to the newest, or a sliding time window of observation can be used.

Additionally, depending on the plant characteristics, the decision-makers should define whether, considering the whole plant, to limit the implementation of the predictive maintenance policy to its portion or most critical components.

Considering the impact of the approach with a broader perspective, its implementation also involves other departments. In fact, one can run the proposed approach before a stoppage occurs. This way, it is possible to know in advance which components could be used for maintenance purposes, and hence to implement an appropriate supply policy.

6.4 General remarks on the data-driven decision model based on the Social Network Analysis

The proposed framework aims at helping maintenance managers come to better, more informed decisions in the day-to-day business practices in order to maximize availability, minimize failures, and optimize costs of Asset Maintenance. From this framework, some theoretical contributions and implications for management can be underlined.

The theoretical contribution provided in this work is essentially twofold: in the problem addressed and in the methodology used. The problem addressed regards a research gap in literature: the prediction of the domino effect between component failures. It is important to underline the importance of using the proposed framework in all companies where there is this domino effect between failures or malfunctioning of components. The results obtained have shown that this phenomenon often occurs in the analyzed plant. It is easy to predict that this behavior is present in many process industries, where the various components (pumps, valves, pipes, tanks, ...) are physically connected to each other.

In addition, this work adopts a data-driven perspective. Hence, the decision-maker implementing such a framework on the process industry relies on the information extracted by the data, rather than on the technical and physical structure of the system – that is, instead, the rationale followed by the model-driven paradigm. This vision expands the body of knowledge of the plant technicians by integrating it with the insights derived from the data analytics.

From a methodological point of view, different techniques have been combined in the proposed framework providing complementary contributions from a theoretical perspective. In particular, the Association Rule Mining method provided researchers with tools to overcome the problems related to the use of traditional statistical techniques such as the vast number of predictive variables, the independence hypothesis, and the non-homogeneity and non-linearity distribution of collected data. The intrinsic organization and complexity of the data collected might jeopardize the use of traditional tools for analysis since the variables showed some critical features. The method based on Association Rules offers many readable patterns (rules) that define the interaction between variables and also avoids the need to formulate a research hypothesis for each failure event before doing a formal evaluation that may become practically infeasible even for a moderately sized set of variables.

The key contribution of the Social Network Analysis concerns the possibility of identifying different communities in network rules and defining how these communities are connected to each other. A community is a cluster of nodes with dense connections internally. The identification of these communities within a network can provide insight into how network function and topology affect each other. Furthermore, identifying communities allows asset managers to predict missing links or false links in the network. During the examination of failure and maintenance events, some links were not understood by asset managers. Similarly, some links were falsely entered into the data because of the errors in the evaluation. Both these cases are well-managed by the community detection algorithm since it assigns the probability of the presence of an edge between a given pair of nodes. In existing literary contributions, as shown in the literature review, only Kim et al. [79] proposed the implementation of an SNA for the synchronous replacement of components. The main difference with their work resides both in the application area and in the definition of the relations among components: indeed, the application area is the construction industry while the relationships among components to be replaced is model-driven, i.e., based on the knowledge of the structure. The approach proposed in this work, on the other hand, is data-driven since the relationships among failures are derived from the records of previous breakages.

Moreover, the contribution provided by the use of both ARM and SNA must be inserted in the context of the process industry, where data derives from different sources affected by veracity problems and which are provided with distinct velocities. The gathering of massive, heterogeneous, and frequently-produced data created a significant management problem. The growth in the dataset volume and its complexity and volatility makes processing and analysis very hard to realize. This work addresses this problem by developing a framework that can be useful to merge and analyze complex data sets through Big Data Analytics techniques in order to extract useful information at different levels of detail. In addition, recurring to the ILP model optimization for selecting the optimal set of components to be replaced ensures the consideration of objective constraints, avoiding any bias possibly introduced by a decision-maker in choosing them arbitrarily.

From a practical point of view, the proposed framework - developing tools for monitoring critical components and predicting fault events - can help different refinery departments, as well as other process industries. The proposed data-driven decision support system enables asset managers to turn predictive analytics insight into prescriptive analytics action by converting information on what is likely to happen in maintenance activities, transforming the raw data into useful and applicable knowledge. In particular, the framework aims to be useful for the maintenance planner, who needs to decide when to maintain each asset, what tasks need to be done, and which parts need to be replaced at each maintenance interval in order to meet reliability targets at an optimal cost. The combined use of ARM and SNA highlights the domino effect among events, with both a visual perspective of the network and the relations existing among the components being determined through a data-driven technique. This is valid support in the decision-making process regarding the predictive component replacement in case of related failures of components: indeed, being a data-driven approach, the work usually carried out by the technicians having a thorough knowledge of the process is supported by the evidence provided by data. Visualizing the relations identified through a data-driven approach helps to identify previously unknown patterns and to take into consideration new perspectives rather than only the traditional ones (e.g., maintenance selection according to the model-driven characteristics of the plant).

Moreover, the integration of ILP helps the maintenance planner to schedule maintenance activities. It provides valid support in the definition of the components to be prioritized for the maintenance, taking into account the resource constraints (e.g., time, budget, number of employees) actually existing in the company. These tools are also important for the parts planner who needs to decide how many of each of the spare parts are needed in which locations and when, so that they can maximize first-time fix rate and reduce spare parts acquisition and holding costs.

Finally, the refinery maintenance technicians, who need to determine the root cause of failures, decide on the best fix and determine whether an asset should be repaired or replaced to minimize turn time, reduce repair cost, and eliminate rework. These decisions must be made for each asset, although each asset has a unique configuration, history, usage, environment, conditions, and parameters, which begins with the commissioning and start-up steps. In this context, the importance of Big Data Analytics tools to determine the best decision option and action plan for each asset becomes evident. Indeed, the proposed framework aims to integrate the analysis of large amounts of data in everyday processes to support real-time decision-making. Decisions in real-time drive efficient maintenance operations, increase equipment reliability, uptime, safety, and reduce overall costs. The proposed asset maintenance framework will not completely change the current oil refinery procedure as a case example. The analytics tools are introduced as an addition to the present one. Therefore, they have to be used both for on-line and off-line asset maintenance activities to ensure a resilient system, i.e., a system able to absorb and resist adverse occurrences.

6.5 General remarks on the data-driven failure extension through the proposed framework

From a theoretical perspective, combining different techniques within the same framework offers insights worthy of mention.

When wanting to analyze the outcome of the FMECA, especially in the case of large and critical plants, a tool enabling the analysis of a vast amount of data is more suitable than the traditional statistical techniques. Specifically, it is one of the benefits introduced by the ARM: indeed, it allows the extraction of patterns characterized by potentially-unknown relationships. Moreover, since the dataset can be analyzed all at once, it is not necessary to formulate further research hypotheses, leaving all data-driven search possibilities unrestricted. Defining all the possible item-sets represents an NP-hard problem since the possible combinations of the items have a size $2^n - 1$ (n being the number of items in the dataset under investigation), excluding the non-valid and empty sets. This issue makes the dimensionality of the input space a critical aspect. However, due to the anti-monotonicity (or downward-closure) property of the support, the definition of the frequent itemset is more efficient since none of the infrequent itemsets is a subset of a frequent itemset. Additionally, the data must be cleaned before starting the ARM process. Indeed, the presence of redundancy in data has an impact both on the quality of the results and on the efficiency of the algorithms.

Through the SNA, instead, it is possible to highlight different communities of nodes and study their interconnections in order to be aware of them and avoid the spread of failure events across the network, interrupting the failure chain. On the other hand, the missing connection among nodes can help experts to understand whether some critical details have been missed during the first stages of the analysis. In this way, the accuracy of the first two stages of the process, namely the data collection and the FMECA, is also verifiable, and corrections to the procedure can be implemented in reasonable time intervals. Including ARM and SNA in the context of process industry is also strategic from a managerial point of view: indeed, for the benefits highlighted, these data-driven techniques provide valid support in analyzing the data coming from different sources, at different velocities and, possibly, characterized by different level of veracity. In this sense, aggregating data from different datasets helps to define whether some replicated data from different sources are corrupted. Hence, it is also possible to correct them before undertaking inconsistent decisions.

Additionally, a complete view of the relationships among FMECA attributes helps the management to identify the improvement areas of the O&M processes. For example, in the SNs proposed in this study, a considerable amount of non-assessable events are presented. These unassessed effects might have critical consequences in a long term perspective since no corrective actions are undertaken to limit or avoid them. Indeed, it is in the best interest of the company to recognize the erratic procedures in the data acquisition process or in the failure effects assessment in order to improve the reliability of the process.

From a practical point of view, this framework enables a major control of the process analysis, specifically in the maintenance field. Firstly, having a procedure defining how the traditional FMECA can be used for furthering the analysis of the failure modes, effects, and maintenance procedure, can provide strategic support and a growth opportunity for the company. Indeed, the more reliable the failure analysis is, the more consistent the achievable benefits. Additionally, the identification of the potential failure modes, effects, and criticalities is useful for supporting the definition of which resources are to be destined to the maintenance procedure. The proposed method enables company managers to connect multidimensional and multidisciplinary concepts (e.g., failure mode, equipment criticality, failure effects at different levels, and maintenance policies adopted).

As stated before, SNA is useful for knowing how a failure mode impacts the possible effect identified and whether these effects are corrected or monitored cyclically, making the understanding of the interactions very intuitive both for domain experts and non-experts. On the other hand, a missing connection among nodes can help experts to understand whether some important details have been missed during the first stages of the analysis. In this way, it also easier to implement corrections and improvements, for instance, by integrating the actions defined after the FMECA or, even further upstream, by adding a failure mode or an effect to equipment.

From an engineering point of view, this visualization is useful for defining the event chains that are more critical since they can act as a trigger for other effects, not only at an equipment level but also at a functional or system level. In this way, specific resources can be dedicated to the analysis of the most critical areas, and not only the critical components, like defining ad-hoc strategies or analyzing the structural improvement of such areas. Furthermore, identifying the possible causes of concatenating failure modes and effects on items allows the identification of the critical items not only from a traditional risk-assessment perspective but also considering the patterns extracted from the data-driven analytics. Indeed, the nodes acting as bridges in the SN, thus connecting different communities of nodes, has to be taken into particular consideration. For such nodes (e.g., the flow control valve in Figure 17), specific monitoring activities can be planned to prevent the spreading of failures or effects across the network. Indeed, in process industries, it is important to be aware of the possible propagation of the effects due to the hazardous nature of the production process and the deriving danger.

The adoption of the proposed framework is also useful in case of re-layout of the plant or designing of similar ones. Indeed, the reliability and failure modes should be taken into account even in the early stages of the realization of a production system: anticipating these issues helps to define proper strategies to deal with them and organize the O&M activities accordingly.

The adoption of new techniques, like ARM and SNA, does not completely change the procedure for failure analysis in the company. They are adopted as an addition to the present one. In this way, the change for the workers is not radical and allows gradual habituation to the new methodologies. This aspect is fundamental in guaranteeing that the personnel accepts the introduction of new methodologies without completely abandoning the previous habits, avoiding possible resistance to change.

Chapter 7.

Conclusions

In this thesis, a framework supporting the implementation of data-driven maintenance policies based on the implementation of data analytics techniques is proposed. Specifically, the development of a framework for the decision-making process is developed to capitalize on the implementation of data-driven techniques, achieve satisfying levels of reliability, and avoid wasting resources using the available amount of data produced and collected during the production processes.

The research gap addressed in this thesis is filled by introducing an innovative decision-making tool in this critical activity through the proposed data-driven framework. Different approaches are proposed, all following the same general procedure.

The framework involves the data collection and management steps, which allow obtaining all the relevant information from the production processes, and it is followed by a thorough analysis of its outcomes. Indeed, during the analytics phase, a preliminary analysis is carried out in order to adjust the data collected in the first stages for the following analysis; then, there is a probability estimation phase – that in this work is carried out through the implementation of the Association Rule Mining or an appositely deployed algorithms. The last stage of the data analytics layer involves developing a decision model that drives the decision-making process, helping the decision-maker make an informed decision. Different decision models are presented in the proposed applications so that the wide applicability of the general framework is presented.

The fourth step of the framework, instead, aims to control the implementation of the data-driven techniques to check whether improvement or modifications to the proposed decision model are needed.

The same general procedure is also proposed in extending the failure analysis. Indeed, the Association Rule Mining is used to define the co-occurrence of events, like failure modes and the related effects, to have a clearer idea of the dynamics describing the possible failure in a company. Thanks to the Social Network Analysis, the Association Rules are represented as a network of nodes (the attributes of the Failure Modes Effects and Criticality Analysis) and direct edges (the Association Rules among the nodes). The possibility of having a graphical representation of the association rules facilitates the global understanding of the context, highlighting which failure modes and effects are related among them and detecting a possible lack of information in order to have a clear view of the process and implement improvement actions.

The proposed research approach is applied to real-life case studies, focusing on industrial plants characterized by high operational risk (e.g., oil refinery plants and on-shore/off-shore plants used for oil and gas extraction).

The results of these implementation highlight that having a precise framework to follow helps in making optimal decisions. If the decision-maker relies on the definition of opportune threshold, there is the need for a wide knowledge of the production plant and the failure frequency. Instead, relying on objective criteria, for example, through the single-objective or multi-objective mathematical programming, the decision-maker is excused from this problem, and the optimal solution is always defined.

The development of a Large Neighborhood Search heuristics to solve the Component Repair Problem as well as the lexicographic optimization represents an innovation in the maintenance management field, as well as the joint use of the Association Rule Mining and mathematical programming. Similarly, the application of the two data-driven techniques, Association Rule Mining and the Social Network Analysis, is rather innovative in the maintenance management field both singularly and jointly.

However, the innovative aspect is not on the algorithm of the single techniques adopted: it lies in how these techniques because are used. Indeed, they are applied in sequence providing information about communities and possible chains of components failing or failure mode propagation across the plant. The idea of combining them after the deployment of the traditional Failure Modes Effects and Criticality Analysis process implies the integration of these new methodologies into this kind of analysis, not its elimination.

From a practical point of view, the implementation of the proposed framework supports in having a major control of system status and of the procedure implemented. Moreover, being able to anticipate the occurrence of component failures and visualizing them ensures that decisions are made basing on data and considering a broad set of relationships, rather than focusing only on the associations known by the field experts. The possibility of applying different methodologies with the same objective ensures a flexible applicability of the proposed framework. Indeed, companies may prefer introducing the proposed approach gradually into their operations so they could start by applying a decision support model based on thresholds definition and, subsequently, extending through optimization methods. Similarly, SNA might be introduced after the ARM so that the initial benefit provided by the latter technique, i.e., the associations among component failures, is firstly acquired and comprehended by the technicians, that have to get used to the new policy; then, the possibility of visualizing the association through the SNA becomes the finisher of the new procedures, without providing excessive new insights all at once and, thus, avoiding the generation of too much information. In this way, the "resistance to change" typical of environment dealing with substantial innovative processes can be mitigated and the new procedures can be gradually accepted even by more expert operators.

Future research directions regard the development of further case studies so that standard procedure belonging to the same field of research might be provided, as well as the extension of the procedure to other industrial sectors. Specifically, applications to manufacturing companies of different dimensions is desirable so that comparisons can be carried out.

A further development regards the introduction of the environmental sustainability dimension inside the framework. Indeed, it is only considered indirectly in the current form since, for example, reducing the number and the impact of failures might avoid any spillage in the environment. On the other hand, being conscious of the impact of a predictive component replacement on the environment can drive the decision-making process in another direction, also considering the disposal of such components. To this end, a hypothesis could be considering the environmental impact derived, for instance, from a Life Cycle Assessment, in the objective function of the optimization models or as an additional constraint.

References

[1] Gröger C, Niedermann F, Mitschang B. Data mining-drivenmanufacturing process optimization. Lect Notes Eng Comput Sci 2012;3:1475–81.

[2] Laggoune R, Chateauneuf A, Aissani D. Impact of few failure data on the opportunistic replacement policy for multi-component systems. Reliab Eng Syst Saf 2010;95:108–19. https://doi.org/10.1016/j.ress.2009.08.007.

[3] Bevilacqua M, Ciarapica FE. Human factor risk management in the process industry: A case study. Reliab Eng Syst Saf 2018;169:149–59. https://doi.org/10.1016/j.ress.2017.08.013.

[4] Shahriar A, Sadiq R, Tesfamariam S. Risk analysis for oil & gas pipelines: A sustainability assessment approach using fuzzy based bow-tie analysis. J Loss Prev Process Ind 2012;25:505–23. https://doi.org/10.1016/j.jlp.2011.12.007.

[5] Garg A, Deshmukh SG. Maintenance management: Literature review and directions. J Qual Maint Eng 2006;12:205–38. https://doi.org/10.1108/13552510610685075.

[6] Groba C, Cech S, Rosenthal F, Gössling A. Architecture of a predictive maintenance framework. Proc. - 6th Int. Conf. Comput. Inf. Syst. Ind. Manag. Appl. CISIM 2007, 2007, p. 59–64. https://doi.org/10.1109/CISIM.2007.14.

[7] BSI Standard. 13306: 2017: Maintenance—Maintenance terminology 2017.

[8] Mobley R. An introduction to predictive maintenance. 2nd ed. New York, NY, USA: Elsevier Science; 2002.

[9] Kumar A, Chinnam RB, Tseng F. An HMM and polynomial regression based approach for remaining useful life and health state estimation of cutting tools. Comput Ind Eng 2019;128:1008–14. https://doi.org/10.1016/j.cie.2018.05.017.

[10] Selcuk S. Predictive maintenance, its implementation and latest trends. Proc Inst Mech Eng Part B J Eng Manuf 2017;231:1670–9. https://doi.org/10.1177/0954405415601640.

[11] Fayyad U, Piatesky-Shapiro G, Smyth P. From Data Mining to Knowledge Discovery in Databases. AI Mag 1996;17:37–54.

https://doi.org/10.5555/257938.257942.

[12] Gibert K, Swayne DA, Yang W, Voinov AA, Rizzoli A, Filatova T. Choosing the Right Data Mining Technique: Classification of Methods and Intelligent Recommendation. IemssOrg 2010.

[13] Crespo Márquez A, de la Fuente Carmona A, Antomarioni S. A Process to Implement an Artificial Neural Network and Association Rules Techniques to Improve Asset Performance and Energy Efficiency. Energies 2019;12:3454. https://doi.org/10.3390/en12183454.

[14] Yew Seng N, Srinivasan R. Data Mining for the Chemical Process Industry. Encycl. Data Warehous. Mining, Second Ed., IGI Global; 2009, p. 458–64. https://doi.org/10.4018/978-1-60566-010-3.CH072.

[15] Bennane A, Yacout S. LAD-CBM; New data processing tool for diagnosis and prognosis in condition-based maintenance. J Intell Manuf 2012;23:265–75. https://doi.org/10.1007/s10845-009-0349-8.

[16] Lin CC, Tseng HY. A neural network application for reliability modelling and condition-based predictive maintenance. Int J Adv Manuf Technol 2005;25:174–9. https://doi.org/10.1007/s00170-003-1835-3.

[17] Lopes Gerum PC, Altay A, Baykal-Gürsoy M. Data-driven predictive maintenance scheduling policies for railways. Transp Res Part C Emerg Technol 2019;107:137–54. https://doi.org/10.1016/j.trc.2019.07.020.

[18] Antomarioni S, Pisacane O, Potena D, Bevilacqua M, Ciarapica FE, Diamantini C. A predictive association rule-based maintenance policy to minimize the probability of breakages: application to an oil refinery. Int J Adv Manuf Technol 2019. https://doi.org/10.1007/s00170-019-03822-y.

[19] Antomarioni S, Bevilacqua M, Potena D, Diamantini C. Defining a data-driven maintenance policy: an application to an oil refinery plant. Int J Qual Reliab Manag 2019;36. https://doi.org/10.1108/IJQRM-01-2018-0012.

[20] Pisacane O, Potena D, Antomarioni S, Bevilacqua M, Emanuele Ciarapica F, Diamantini C. Data-driven predictive maintenance policy based on multi-objective optimization approaches for the component repairing problem. Eng Optim 2020.

https://doi.org/10.1080/0305215X.2020.1823381.
[21] Becattini N, Cascini G, Nikulin C. Modelling the Dynamics of Products and Processes Requirements. Procedia Eng 2015;131:661–71. https://doi.org/10.1016/j.proeng.2015.12.348.
[22] Khan R, B. Mad A, Osman K, Asyraf Abd Aziz M. Maintenance Management of Aging Oil and Gas Facilities. Maint. Manag., IntechOpen; 2020. https://doi.org/10.5772/intechopen.82841.
[23] Viveros P, Nikulin C, López-Campos M, Villalón R, Crespo A. Resolution of reliability problems based on failure mode analysis: an integrated proposal applied to a mining case study. Prod Plan Control 2018;29:1225–37. https://doi.org/10.1080/09537287.2018.1520293.
[24] Bevilacqua M, Ciarapica FE, Diamantini C, Potena D. Big data analytics methodologies applied at energy management in industrial sector: A case study. Int J RF Technol Res Appl 2017;8:105–22. https://doi.org/10.3233/RFT-171671.
[25] Bevilacqua M, Ciarapica FE, Mazzuto G. Fuzzy cognitive maps for adverse drug event risk management. Saf Sci 2018;102:194–210. https://doi.org/10.1016/j.ssci.2017.10.022.
[26] Carpitella S, Certa A, Izquierdo J, La Fata CM. A combined multi-criteria approach to support FMECA analyses: A real-world case. Reliab Eng Syst Saf 2018;169:394–402. https://doi.org/10.1016/j.ress.2017.09.017.
[27] Luo SH, Lee GG. Applying failure mode and effects analysis for successful knowledge management. Total Qual Manag Bus Excell 2015;26:62–75. https://doi.org/10.1080/14783363.2012.733263.
[28] Grunske L, Lindsay P, Yatapanage N, Winter K. An automated failure mode and effect analysis based on high-level design specification with behavior trees. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 3771 LNCS, Springer, Berlin, Heidelberg; 2005, p. 129–49. https://doi.org/10.1007/11589976_9.
[29] Tso KS, Tai AT, Chau SN, Alkalai L. On automating failure mode analysis and enhancing its integrity. Proc. - 11th Pacific Rim Int. Symp. Dependable Comput.

PRDC 2005, vol. 2005, IEEE; 2005, p. 287–92. https://doi.org/10.1109/PRDC.2005.42.
[30] Ben Said A, Shahzad MK, Zamai E, Hubac S, Tollenaere M. Experts' knowledge renewal and maintenance actions effectiveness in high-mix low-volume industries, using Bayesian approach. Cogn Technol Work 2016;18:193–213. https://doi.org/10.1007/s10111-015-0354-y.
[31] Aggogeri F, Adamini R, Aivaliotis P, Borboni A, Eytan A, Merlo A, et al. Robotic System Reliability Analysis and RUL Estimation Using an Iterative Approach. Adv. Intell. Syst. Comput., vol. 980, Springer Verlag; 2020, p. 134–43. https://doi.org/10.1007/978-3-030-19648-6_16.
[32] Ragab A, Yacout S, Ouali MS, Osman H. Prognostics of multiple failure modes in rotating machinery using a pattern-based classifier and cumulative incidence functions. J Intell Manuf 2019;30:255–74. https://doi.org/10.1007/s10845-016-1244-8.
[33] Antomarioni S, Bellinello MM, Bevilacqua M, Ciarapica FE, da Silva RF, de Souza GFM. A Data-Driven Approach to Extend Failure Analysis: A Framework Development and a Case Study on a Hydroelectric Power Plant. Energies 2020;13:6400. https://doi.org/10.3390/en13236400.
[34] Diamantini C, Potena D, Storti E. A virtual mart for knowledge discovery in databases. Inf Syst Front 2013. https://doi.org/10.1007/s10796-012-9399-0.
[35] Shahbaz M, Srinivas, Harding JA, Turner M. Product design and manufacturing process improvement using association rules. Proc Inst Mech Eng Part B J Eng Manuf 2006. https://doi.org/10.1243/095440506X78183.
[36] Harding JA, Shahbaz M, Srinivas, Kusiak A. Data mining in manufacturing: A review. J Manuf Sci Eng Trans ASME 2006;128:969–76. https://doi.org/10.1115/1.2194554.
[37] Braha D. Data Mining for Design and Manufacturing. vol. 3. Boston, MA: Springer US; 2001. https://doi.org/10.1007/978-1-4757-4911-3.
[38] Kusiak A. Engineering design: products, processes, and systems 1999.
[39] Skormin VA, Gorodetski VI, Popyack LJ. Data mining technology for failure

prognostic of avionics. IEEE Trans Aerosp Electron Syst 2002;38:388–403. https://doi.org/10.1109/TAES.2002.1008974.
[40] Oh S, Han J, Cho H. Intelligent Process Control System for Quality Improvement by Data Mining in the Process Industry, Springer, Boston, MA; 2001, p. 289–309. https://doi.org/10.1007/978-1-4757-4911-3_12.
[41] Maki H, Teranishi Y. Development of automated data mining system for quality control in manufacturing. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 2114, Springer Verlag; 2001, p. 93–100. https://doi.org/10.1007/3-540-44801-2_10.
[42] Gardner M, Bieker J. Data Mining Solves Tough Semiconductor Manufacturing Problems. Proc. sixth ACM SIGKDD Int. Conf. Knowl. Discov. data Min., 2000, p. 376-383.
[43] Batanov D, Nagarur N, Nitikhunkasem P. EXPERT-MM: A knowledge-based system for maintenance management. Artif Intell Eng 1993;8:283–91. https://doi.org/10.1016/0954-1810(93)90012-5.
[44] Romanowski CJ, Nagi R. Analyzing Maintenance Data Using Data Mining Methods, Springer, Boston, MA; 2001, p. 235–54. https://doi.org/10.1007/978-1-4757-4911-3_10.
[45] Bumblauskas D, Gemmill D, Igou A, Anzengruber J. Smart Maintenance Decision Support Systems (SMDSS) based on corporate big data analytics. Expert Syst Appl 2017;90:303–17. https://doi.org/10.1016/j.eswa.2017.08.025.
[46] Manco G, Ritacco E, Rullo P, Gallucci L, Astill W, Kimber D, et al. Fault detection and explanation through big data analysis on sensor streams. Expert Syst Appl 2017;87:141–56. https://doi.org/10.1016/j.eswa.2017.05.079.
[47] Kabir MJ, Oo AMT, Rabbani M. A brief review on offshore wind turbine fault detection and recent development in condition monitoring based maintenance system. 2015 Australas. Univ. Power Eng. Conf. Challenges Futur. Grids, AUPEC 2015, Institute of Electrical and Electronics Engineers Inc.; 2015. https://doi.org/10.1109/AUPEC.2015.7324871.
[48] Victor Antony J, Nasira GM. Enabling Predictive Maintenance Strategy in Rail

Sector: A Clustering Approach. WSEAS Trans. Comput., 2014, p. 118–28.

[49] Sammouri W, Côme E, Oukhellou L, Aknin P, Fonlladosa CE. Pattern recognition approach for the prediction of infrequent target events in floating train data sequences within a predictive maintenance framework. 2014 17th IEEE Int. Conf. Intell. Transp. Syst. ITSC 2014, Institute of Electrical and Electronics Engineers Inc.; 2014, p. 918–23. https://doi.org/10.1109/ITSC.2014.6957806.

[50] Jin W, Shi Z, Siegel D, Dersin P, Douziech C, Pugnaloni M, et al. Development and evaluation of health monitoring techniques for railway point machines. 2015 IEEE Conf. Progn. Heal. Manag. Enhancing Safety, Effic. Availability, Eff. Syst. Through PHAf Technol. Appl. PHM 2015, Institute of Electrical and Electronics Engineers Inc.; 2015. https://doi.org/10.1109/ICPHM.2015.7245016.

[51] Ming Z. Decision approach of maintenance for urban rail transit based on equipment supervision data mining. Proc. 2015 IEEE 8th Int. Conf. Intell. Data Acquis. Adv. Comput. Syst. Technol. Appl. IDAACS 2015, vol. 1, Institute of Electrical and Electronics Engineers Inc.; 2015, p. 376–80. https://doi.org/10.1109/IDAACS.2015.7340761.

[52] Sipos R, Fradkin D, Moerchen F, Wang Z. Log-based predictive maintenance. Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., New York, New York, USA: Association for Computing Machinery; 2014, p. 1867–76. https://doi.org/10.1145/2623330.2623340.

[53] Wang KS. Key techniques in intelligent predictive maintenance (IPdM) - A framework of intelligent faults diagnosis and prognosis system (IFDaPS). Adv. Mater. Res., vol. 1039, Trans Tech Publications Ltd; 2014, p. 490–505. https://doi.org/10.4028/www.scientific.net/AMR.1039.490.

[54] Saybani MR, Wah TY. Data mining and data gathering in a refinery. Int. Conf. Appl. Comput. Sci. - Proc., 2010, p. 62–6.

[55] Köksal G, Batmaz I, Testik MC. A review of data mining applications for quality improvement in manufacturing industry. Expert Syst Appl 2011;38:13448–67. https://doi.org/10.1016/j.eswa.2011.04.063.

[56] Zhong NN, Wang X. Application of data mining in refinery CIMS. Int. Conf.

Mach. Learn. Cybern., vol. 1, 2003, p. 118–22. https://doi.org/10.1109/icmlc.2003.1264454.

[57] Wang Q, Gao J. Research and application of risk and condition based maintenance task optimization technology in an oil transfer station. J Loss Prev Process Ind 2012;25:1018–27. https://doi.org/10.1016/j.jlp.2012.06.002.

[58] Friedemann JD, Varma A, Bonissone P, Iyer N. Subsea condition monitoring: A path to increased availability and increased recovery. Soc. Pet. Eng. - Intell. Energy Conf. Exhib. Intell. Energy 2008, vol. 1, Society of Petroleum Engineers; 2008, p. 298–308. https://doi.org/10.2118/112051-ms.

[59] Li JR, Khoo LP, Tor SB. RMINE: A rough set based data mining prototype for the reasoning of incomplete data in condition-based fault diagnosis. J Intell Manuf 2006;17:163–76. https://doi.org/10.1007/s10845-005-5519-8.

[60] Moura MC, Lins ID, Ferreira RJ, Droguett EL, Jacinto CMC. Predictive maintenance policy for oil well equipment in case of scaling through Support Vector Machines. Adv. Safety, Reliab. Risk Manag. - Proc. Eur. Saf. Reliab. Conf. ESREL 2011, 2012, p. 503–7. https://doi.org/10.1201/b11433-70.

[61] Hu J, Zhang L, Liang W, Wang Z. Incipient mechanical fault detection based on multifractal and MTS methods. Pet Sci 2009;6:208–16. https://doi.org/10.1007/s12182-009-0034-8.

[62] Zhang S, Wu X. Fundamentals of association rules in data mining and knowledge discovery. Wiley Interdiscip Rev Data Min Knowl Discov 2011;1:97–116. https://doi.org/10.1002/widm.10.

[63] Wang K. Applying data mining to manufacturing: The nature and implications. J Intell Manuf 2007;18:487–95. https://doi.org/10.1007/s10845-007-0053-5.

[64] Buddhakulsomsiri J, Siradeghyan Y, Zakarian A, Li X. Association rule-generation algorithm for mining automotive warranty data. Int J Prod Res 2006. https://doi.org/10.1080/00207540600564633.

[65] Rygielski C, Wang JC, Yen DC. Data mining techniques for customer relationship management. Technol Soc 2002;24:483–502. https://doi.org/10.1016/S0160-791X(02)00038-6.

[66] Choudhary AK, Harding JA, Tiwari MK. Data mining in manufacturing: A review based on the kind of knowledge. J Intell Manuf 2009;20:501–21. https://doi.org/10.1007/s10845-008-0145-x.
[67] Chen MC. Configuration of cellular manufacturing systems using association rule induction. Int J Prod Res 2003;41:381–95. https://doi.org/10.1080/0020754021000024184.
[68] Kamsu-Foguem B, Rigal F, Mauget F. Mining association rules for the quality improvement of the production process. Expert Syst Appl 2013. https://doi.org/10.1016/j.eswa.2012.08.039.
[69] Martínez-De-Pisón FJ, Sanz A, Martínez-De-Pisón E, Jiménez E, Conti D. Mining association rules from time series to explain failures in a hot-dip galvanizing steel line. Comput Ind Eng 2012. https://doi.org/10.1016/j.cie.2012.01.013.
[70] Djatna T, Alitu IM. An Application of Association Rule Mining in Total Productive Maintenance Strategy: An Analysis and Modelling in Wooden Door Manufacturing Industry. Procedia Manuf 2015. https://doi.org/10.1016/j.promfg.2015.11.049.
[71] Bastos, P., Lopes, I., & Pires LCM. A maintenance prediction system using data mining techniques. In World Congress on Engineering 2012, Newswood Ltd.; 2012, p. 2065.
[72] Chen WC, Tseng SS, Wang CY. A novel manufacturing defect detection method using association rule mining techniques. Expert Syst Appl 2005. https://doi.org/10.1016/j.eswa.2005.06.004.
[73] Wang Z, Shao X, Zhang G, Zhu H. Integration of variable precision rough set and fuzzy clustering: An application to knowledge acquisition for manufacturing process planning. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 3642 LNAI, Springer, Berlin, Heidelberg; 2005, p. 585–93. https://doi.org/10.1007/11548706_62.
[74] Agard B, Kusiak A. Data mining for subassembly selection. J Manuf Sci Eng Trans ASME 2004;126:627–31. https://doi.org/10.1115/1.1763182.
[75] Da Cunha C, Agard B, Kusiak A. Data mining for improvement of product quality. Int. J. Prod. Res., vol. 44, Taylor & Francis Group ; 2006, p. 4027–41.

https://doi.org/10.1080/00207540600678904.
[76] Li F, Upadhyaya BR, Coffey LA. Model-based monitoring and fault diagnosis of fossil power plant process units using Group Method of Data Handling. ISA Trans 2009;48:213–9. https://doi.org/10.1016/j.isatra.2008.10.014.
[77] Liu Y, Hu X, Zhang W. Remaining useful life prediction based on health index similarity. Reliab Eng Syst Saf 2019;185:502–10. https://doi.org/10.1016/j.ress.2019.02.002.
[78] Wu J, Hu K, Cheng Y, Zhu H, Shao X, Wang Y. Data-driven remaining useful life prediction via multiple sensor signals and deep long short-term memory neural network. ISA Trans 2020;97:241–50. https://doi.org/10.1016/j.isatra.2019.07.004.
[79] Kim J, Han S, Hyun CT. Identification and Reduction of Synchronous Replacements in Life-Cycle Cost Analysis of Equipment. J Manag Eng 2019;35:04018058. https://doi.org/10.1061/(ASCE)ME.1943-5479.0000673.
[80] Baptista M, Sankararaman S, de Medeiros IP, Nascimento C, Prendinger H, Henriques EMP. Forecasting fault events for predictive maintenance using data-driven techniques and ARMA modeling. Comput Ind Eng 2018;115:41–53. https://doi.org/10.1016/j.cie.2017.10.033.
[81] Tang K, Parsons DJ, Jude S. Comparison of automatic and guided learning for Bayesian networks to analyse pipe failures in the water distribution system. Reliab Eng Syst Saf 2019;186:24–36. https://doi.org/10.1016/j.ress.2019.02.001.
[82] Bangalore P, Tjernberg LB. An artificial neural network approach for early fault detection of gearbox bearings. IEEE Trans Smart Grid 2015;6:980–7. https://doi.org/10.1109/TSG.2014.2386305.
[83] Gharoun H, Keramati A, Nasiri MM, Azadeh A. An integrated approach for aircraft turbofan engine fault detection based on data mining techniques. Expert Syst 2019;36:e12370. https://doi.org/10.1111/exsy.12370.
[84] Baptista M, Medeiros IPD, Malere JP, Prendinger H, Nascimento CL, Henriques EMP. Improved Time-Based Maintenance in Aeronautics with Regressive Support Vector Machines. Annu. Conf. Progn. Heal. Soc., 2016, p. 1–10.
[85] Medjaher K, Tobon-Mejia DA, Zerhouni N. Remaining useful life estimation of

critical components with application to bearings. IEEE Trans Reliab 2012;61:292–302. https://doi.org/10.1109/TR.2012.2194175.

[86] Chen Z, Li Y, Xia T, Pan E. Hidden Markov model with auto-correlated observations for remaining useful life prediction and optimal maintenance policy. Reliab Eng Syst Saf 2019;184:123–36. https://doi.org/10.1016/j.ress.2017.09.002.

[87] Sattari F, Macciotta R, Kurian D, Lefsrud L. Application of Bayesian network and artificial intelligence to reduce accident/incident rates in oil & gas companies. Saf Sci 2021;133:104981. https://doi.org/10.1016/j.ssci.2020.104981.

[88] Otte E, Rousseau R. Social network analysis: a powerful strategy, also for the information sciences. J Inf Sci 2002;28:441–53. https://doi.org/10.1177/016555150202800601.

[89] Scott J. Social Network Analysis. Sociology 1988;22:109–27. https://doi.org/10.1177/0038038588022001007.

[90] Ruan X, Ochieng DEG, Price AD, Egbu CO. Time for a Real Shift to Relations: Appraisal of Social Network Analysis Applications in the UK Construction Industry. SSRN Electron J 2015. https://doi.org/10.2139/ssrn.2623032.

[91] Chan A, Javed AA, Wong F, Hon C, Zahoor H, Lyu S. The application of social network analysis in the construction industry of Hong Kong. Int Conf Emerg Trends Eng Manag Sci 2014.

[92] Eteifa SO, El-Adaway IH. Using Social Network Analysis to Model the Interaction between Root Causes of Fatalities in the Construction Industry. J Manag Eng 2018;34:04017045. https://doi.org/10.1061/(ASCE)ME.1943-5479.0000567.

[93] Ciarapica F, Bevilacqua M, Antomarioni S. An approach based on association rules and social network analysis for managing environmental risk: A case study from a process industry. Process Saf Environ Prot 2019;128:50–64. https://doi.org/10.1016/j.psep.2019.05.037.

[94] Datong L, Yu P, Xiyuan P. Fault prediction based on time series with online combined kernel SVR methods. 2009 IEEE Intrumentation Meas. Technol. Conf. I2MTC 2009, IEEE Computer Society; 2009, p. 1163–6. https://doi.org/10.1109/IMTC.2009.5168630.

[95] Purarjomandlangrudi A, Ghapanchi AH, Esmalifalak M. A data mining approach for fault diagnosis: An application of anomaly detection algorithm. Meas J Int Meas Confed 2014;55:343–52. https://doi.org/10.1016/j.measurement.2014.05.029.

[96] Langone R, Alzate C, De Ketelaere B, Vlasselaer J, Meert W, Suykens JAK. LS-SVM based spectral clustering and regression for predicting maintenance of industrial machines. Eng Appl Artif Intell 2015;37:268–78. https://doi.org/10.1016/j.engappai.2014.09.008.

[97] Saravanan N, Siddabattuni VNSK, Ramachandran KI. Fault diagnosis of spur bevel gear box using artificial neural network (ANN), and proximal support vector machine (PSVM). Appl Soft Comput J 2010;10:344–60. https://doi.org/10.1016/j.asoc.2009.08.006.

[98] Salahshoor K, Kordestani M, Khoshro MS. Fault detection and diagnosis of an industrial steam turbine using fusion of SVM (support vector machine) and ANFIS (adaptive neuro-fuzzy inference system) classifiers. Energy 2010;35:5472–82. https://doi.org/10.1016/j.energy.2010.06.001.

[99] Kankar PK, Sharma SC, Harsha SP. Fault diagnosis of ball bearings using machine learning methods. Expert Syst Appl 2011;38:1876–86. https://doi.org/10.1016/j.eswa.2010.07.119.

[100] Wang D. K-nearest neighbors based methods for identification of different gear crack levels under different motor speeds and loads: Revisited. Mech Syst Signal Process 2016;70–71:201–8. https://doi.org/10.1016/j.ymssp.2015.10.007.

[101] Schlechtingen M, Ferreira Santos I. Comparative analysis of neural network and regression based condition monitoring approaches for wind turbine fault detection. Mech Syst Signal Process 2011;25:1849–75. https://doi.org/10.1016/j.ymssp.2010.12.007.

[102] Kusiak A, Verma A. Analyzing bearing faults in wind turbines: A data-mining approach. Renew Energy 2012;48:110–6. https://doi.org/10.1016/j.renene.2012.04.020.

[103] Yam RCM, Tse PW, Li L, Tu P. Intelligent predictive decision support system for condition-based maintenance. Int J Adv Manuf Technol 2001;17:383–91.

https://doi.org/10.1007/s001700170173.

[104] Izquierdo J, Crespo Márquez A, Uribetxebarria J. Dynamic artificial neural network-based reliability considering operational context of assets. Reliab Eng Syst Saf 2019;188:483–93. https://doi.org/10.1016/j.ress.2019.03.054.

[105] Mazhar MI, Kara S, Kaebernick H. Remaining life estimation of used components in consumer products: Life cycle data analysis by Weibull and artificial neural networks. J Oper Manag 2007;25:1184–93. https://doi.org/10.1016/j.jom.2007.01.021.

[106] Li D, Wang W, Ismail F. Enhanced fuzzy-filtered neural networks for material fatigue prognosis. Appl Soft Comput J 2013;13:283–91. https://doi.org/10.1016/j.asoc.2012.08.031.

[107] Létourneau S, Famili F, Matwin S. Data mining to predict aircraft component replacement. IEEE Intell Syst Their Appl 1999;14:59–60. https://doi.org/10.1109/5254.809569.

[108] Bahga A, Madisetti VK. Analyzing massive machine maintenance data in a computing cloud. IEEE Trans Parallel Distrib Syst 2012;23:1831–43. https://doi.org/10.1109/TPDS.2011.306.

[109] Kumar G, Jain V, Gandhi OP. Availability analysis of mechanical systems with condition-based maintenance using semi-Markov and evaluation of optimal condition monitoring interval. J Ind Eng Int 2018;14:119–31. https://doi.org/10.1007/s40092-017-0212-z.

[110] Sharma S, Cui Y, He Q, Mohammadi R, Li Z. Data-driven optimization of railway maintenance for track geometry. Transp Res Part C Emerg Technol 2018;90:34–58. https://doi.org/10.1016/j.trc.2018.02.019.

[111] Abbassi R, Bhandari J, Khan F, Garaniya V, Chai S. Developing a quantitative risk-based methodology for maintenance scheduling using Bayesian network. Chem Eng Trans 2016;48:235–40. https://doi.org/10.3303/CET1648040.

[112] Dekker R. Applications of maintenance optimization models: A review and analysis. Reliab Eng Syst Saf 1996;51:229–40. https://doi.org/10.1016/0951-8320(95)00076-3.

[113] Bortolini M, Ferrari E, Gamberi M, Pilati F, Faccio M. Assembly system design in the Industry 4.0 era: a general framework. IFAC-PapersOnLine 2017;50:5700–5. https://doi.org/10.1016/j.ifacol.2017.08.1121.

[114] Alkhamis TM, Yellen J. Refinery units maintenance scheduling using integer programming. Appl Math Model 1995;19:543–9. https://doi.org/10.1016/0307-904X(95)00032-F.

[115] Pistikopoulos EN, Vassiliadis CG, Papageorgiou LG. Process design for maintainability: An optimization approach. Comput. Chem. Eng., vol. 24, Elsevier Science Ltd; 2000, p. 203–8. https://doi.org/10.1016/S0098-1354(00)00514-7.

[116] Goel HD, Grievink J, Weijnen MPC. Integrated optimal reliable design, production, and maintenance planning for multipurpose process plants. Comput Chem Eng 2003;27:1543–55. https://doi.org/10.1016/S0098-1354(03)00090-5.

[117] Alrabghi A, Tiwari A. State of the art in simulation-based optimisation for maintenance systems. Comput Ind Eng 2015;82:167–82. https://doi.org/10.1016/j.cie.2014.12.022.

[118] Allaoui H, Artiba A. Integrating simulation and optimization to schedule a hybrid flow shop with maintenance constraints. Comput Ind Eng 2004;47:431–50. https://doi.org/10.1016/j.cie.2004.09.002.

[119] Rezg N, Xie X, Mati Y. Joint optimization of preventive maintenance and inventory control in a production line using simulation. Int J Prod Res 2004;42:2029–46. https://doi.org/10.1080/00207540310001638235.

[120] Sarker R, Haque A. Optimization of maintenance and spare provisioning policy using simulation. Appl Math Model 2000;24:751–60. https://doi.org/10.1016/S0307-904X(00)00011-1.

[121] Cassady CR, Bowden RO, Liew L, Pohl EA. Combining preventive maintenance and statistical process control: A preliminary investigation. IIE Trans (Institute Ind Eng 2000;32:471–8. https://doi.org/10.1080/07408170008963924.

[122] Gharbi A, Kenné JP. Maintenance scheduling and production control of multiple-machine manufacturing systems. Comput. Ind. Eng., vol. 48, Pergamon; 2005, p. 693–707. https://doi.org/10.1016/j.cie.2004.12.007.

[123] Tagaras G. An integrated cost model for the joint optimization of process control and maintenance. J Oper Res Soc 1988;39:757–66. https://doi.org/10.1057/jors.1988.131.

[124] Bertolini M, Bevilacqua M. A combined goal programming - AHP approach to maintenance selection problem. Reliab Eng Syst Saf 2006;91:839–48. https://doi.org/10.1016/j.ress.2005.08.006.

[125] Lee HT, Yang DL, Yang SJ. Multi-machine scheduling with deterioration effects and maintenance activities for minimizing the total earliness and tardiness costs. Int J Adv Manuf Technol 2013;66:547–54. https://doi.org/10.1007/s00170-012-4348-0.

[126] Kenne JP, Boukas EK, Gharbi A. Control of production and corrective maintenance rates in a multiple-machine, multiple-product manufacturing system. Math Comput Model 2003;38:351–65. https://doi.org/10.1016/S0895-7177(03)90093-8.

[127] Vilarinho S, Lopes I, Oliveira JA. Preventive Maintenance Decisions through Maintenance Optimization Models: A Case Study. Procedia Manuf 2017;11:1170–7. https://doi.org/10.1016/j.promfg.2017.07.241.

[128] Mokhtari H, Mozdgir A, Kamal Abadi IN. A reliability/availability approach to joint production and maintenance scheduling with multiple preventive maintenance services. Int J Prod Res 2012;50:5906–25. https://doi.org/10.1080/00207543.2011.637092.

[129] Irawan CA, Ouelhadj D, Jones D, Stålhane M, Sperstad IB. Optimisation of maintenance routing and scheduling for offshore wind farms. Eur J Oper Res 2017;256:76–89. https://doi.org/10.1016/j.ejor.2016.05.059.

[130] Xia T, Xi L, Zhou X, Lee J. Condition-based maintenance for intelligent monitored series system with independent machine failure modes. Int J Prod Res 2013;51:4585–96. https://doi.org/10.1080/00207543.2013.775524.

[131] Chalabi N, Dahane M, Beldjilali B, Neki A. Optimisation of preventive maintenance grouping strategy for multi-component series systems: Particle swarm based approach. Comput Ind Eng 2016;102:440–51. https://doi.org/10.1016/j.cie.2016.04.018.

[132] Wang S, Liu M. Multi-objective optimization of parallel machine scheduling integrated with multi-resources preventive maintenance planning. J Manuf Syst 2015;37:182–92. https://doi.org/10.1016/j.jmsy.2015.07.002.

[133] Hadjaissa B, Ameur K, Ait cheikh SM, Essounbouli N. Bi-objective optimization of maintenance scheduling for power systems. Int J Adv Manuf Technol 2016;85:1361–72. https://doi.org/10.1007/s00170-015-8053-7.

[134] Shafiee M, Sørensen JD. Maintenance optimization and inspection planning of wind energy assets: Models, methods and strategies. Reliab Eng Syst Saf 2019;192:105993. https://doi.org/10.1016/j.ress.2017.10.025.

[135] Ding SH, Kamaruddin S. Maintenance policy optimization—literature review and directions. Int J Adv Manuf Technol 2014;76:1263–83. https://doi.org/10.1007/s00170-014-6341-2.

[136] Xia T, Xi L, Lee J, Zhou X. Optimal CBPM policy considering maintenance effects and environmental condition. Int J Adv Manuf Technol 2011;56:1181–93. https://doi.org/10.1007/s00170-011-3235-4.

[137] Ilgin MA, Tunali S. Joint optimization of spare parts inventory and maintenance policies using genetic algorithms. Int J Adv Manuf Technol 2007;34:594–604. https://doi.org/10.1007/s00170-006-0618-z.

[138] Ruiz R, Carlos García-Díaz J, Maroto C. Considering scheduling and preventive maintenance in the flowshop sequencing problem. Comput Oper Res 2007;34:3314–30. https://doi.org/10.1016/j.cor.2005.12.007.

[139] Marseguerra M, Zio E, Podofillini L. Condition-based maintenance optimization by means of genetic algorithms and Monte Carlo simulation. Reliab Eng Syst Saf 2002;77:151–65. https://doi.org/10.1016/S0951-8320(02)00043-1.

[140] Marseguerra M, Zio E, Podofillini L. Optimal reliability/availability of uncertain systems via multi-objective genetic algorithms. IEEE Trans Reliab 2004;53:424–34. https://doi.org/10.1109/TR.2004.833318.

[141] Huang HZ, Qu J, Zuo M. Genetic-algorithm-based optimal apportionment of reliability and redundancy under multiple objectives. IIE Trans (Institute Ind Eng 2009;41:287–98. https://doi.org/10.1080/07408170802322994.

[142] Okasha NM, Frangopol DM. Lifetime-oriented multi-objective optimization of structural maintenance considering system reliability, redundancy and life-cycle cost using GA. Struct Saf 2009;31:460–74. https://doi.org/10.1016/j.strusafe.2009.06.005.

[143] Min LX, Yong WJ, Yuan Y, Yan XW. Multiobjective optimization of preventive maintenance schedule on traction power system in high-speed railway. Proc. - Annu. Reliab. Maintainab. Symp., 2009, p. 365–70. https://doi.org/10.1109/RAMS.2009.4914704.

[144] Raad D, Sinske A, Van Vuuren J. Robust multi-objective optimization for water distribution system design using a meta-metaheuristic. Int Trans Oper Res 2009;16:595–626. https://doi.org/10.1111/j.1475-3995.2009.00705.x.

[145] Berrichi A, Yalaoui F, Amodeo L, Mezghiche M. Bi-Objective Ant Colony Optimization approach to optimize production and maintenance scheduling. Comput Oper Res 2010;37:1584–96. https://doi.org/10.1016/j.cor.2009.11.017.

[146] Carlos S, Sanchez A, Martorell S, Villanueva JF. Particle Swarm Optimization of safety components and systems of nuclear power plants under uncertain maintenance planning. Adv. Eng. Softw., vol. 50, Elsevier Ltd; 2012, p. 12–8. https://doi.org/10.1016/j.advengsoft.2012.04.004.

[147] Gjorgiev B, Kančev D, Čepin M. Risk-informed decision making in the nuclear industry: Application and effectiveness comparison of different genetic algorithm techniques. Nucl Eng Des 2012;250:701–12. https://doi.org/10.1016/j.nucengdes.2012.06.018.

[148] Tian Z, Lin D, Wu B. Condition based maintenance optimization considering multiple objectives. J Intell Manuf 2012;23:333–40. https://doi.org/10.1007/s10845-009-0358-7.

[149] Loganathan MK, Gandhi OP. Maintenance cost minimization of manufacturing systems using PSO under reliability constraint. Int J Syst Assur Eng Manag 2016;7:47–61. https://doi.org/10.1007/s13198-015-0374-2.

[150] Moghaddam KS, Usher JS. A new multi-objective optimization model for preventive maintenance and replacement scheduling of multi-component systems.

Eng Optim 2011;43:701–19. https://doi.org/10.1080/0305215X.2010.512084.
[151] Moghaddam KS. Multi-objective preventive maintenance and replacement scheduling in a manufacturing system using goal programming. Int J Prod Econ 2013;146:704–16. https://doi.org/10.1016/j.ijpe.2013.08.027.
[152] Ebrahimipour V, Najjarbashi A, Sheikhalishahi M. Multi-objective modeling for preventive maintenance scheduling in a multiple production line. J Intell Manuf 2013;26:111–22. https://doi.org/10.1007/s10845-013-0766-6.
[153] Seif J, Yu AJ, Rahmanniyay F. Modelling and optimization of a bi-objective flow shop scheduling with diverse maintenance requirements. Int J Prod Res 2018;56:3204–25. https://doi.org/10.1080/00207543.2017.1403660.
[154] Arunraj NS, Maiti J. Risk-based maintenance policy selection using AHP and goal programming. Saf Sci 2010;48:238–47. https://doi.org/10.1016/j.ssci.2009.09.005.
[155] Moghaddam KS. Preventive maintenance and replacement optimization on CNC machine using multiobjective evolutionary algorithms. Int J Adv Manuf Technol 2015;76:2131–46. https://doi.org/10.1007/s00170-014-6391-5.
[156] Fan Y, Xia X. A multi-objective optimization model for energy-efficiency building envelope retrofitting plan with rooftop PV system installation and maintenance. Appl Energy 2017;189:327–35. https://doi.org/10.1016/j.apenergy.2016.12.077.
[157] Wu Z, Xia X, Wang B. Improving building energy efficiency by multiobjective neighborhood field optimization. Energy Build 2015;87:45–56. https://doi.org/10.1016/j.enbuild.2014.10.079.
[158] Su C, Liu Y. Multi-objective imperfect preventive maintenance optimisation with NSGA-II. Int J Prod Res 2020;58:4033–49. https://doi.org/10.1080/00207543.2019.1641237.
[159] Huang HZ, Qu J, Zuo M. Genetic-algorithm-based optimal apportionment of reliability and redundancy under multiple objectives. IIE Trans (Institute Ind Eng 2009;41:287–98. https://doi.org/10.1080/07408170802322994.
[160] Certa A, Enea M, Galante G, Lupo T. A multi-objective approach to optimize a periodic maintenance policy. Int J Reliab Qual Saf Eng 2012;19. https://doi.org/10.1142/S0218539312400025.

[161] Certa A, Enea M, Lupo T. ELECTRE III to dynamically support the decision maker about the periodic replacements configurations for a multi-component system. Decis Support Syst 2013;55:126–34. https://doi.org/10.1016/j.dss.2012.12.044.

[162] Wang CH, Tsai SW. Optimizing bi-objective imperfect preventive maintenance model for series-parallel system using established hybrid genetic algorithm. J Intell Manuf 2014;25:603–16. https://doi.org/10.1007/s10845-012-0708-8.

[163] Savino MM, Brun A, Riccio C. Integrated system for maintenance and safety management through FMECA principles and fuzzy inference engine. Eur J Ind Eng 2011;5:132–69. https://doi.org/10.1504/EJIE.2011.039870.

[164] Xu Z, Dang Y, Munro P, Wang Y. A data-driven approach for constructing the component-failure mode matrix for FMEA. J Intell Manuf 2020;31:249–65. https://doi.org/10.1007/s10845-019-01466-z.

[165] Chang KH, Cheng CH. Evaluating the risk of failure using the fuzzy OWA and DEMATEL method. J Intell Manuf 2011;22:113–29. https://doi.org/10.1007/s10845-009-0266-x.

[166] Liu HC, Chen YZ, You JX, Li H. Risk evaluation in failure mode and effects analysis using fuzzy digraph and matrix approach. J Intell Manuf 2016;27:805–16. https://doi.org/10.1007/s10845-014-0915-6.

[167] Khorshidi HA, Gunawan I, Ibrahim MY. Data-Driven System Reliability and Failure Behavior Modeling Using FMECA. IEEE Trans Ind Informatics 2016;12:1253–60. https://doi.org/10.1109/TII.2015.2431224.

[168] Li X, Ran Y, Zhang G, He Y. A failure mode and risk assessment method based on cloud model. J Intell Manuf 2019. https://doi.org/10.1007/s10845-019-01513-9.

[169] Lv J, Xu S, Zhang R, Xiao H;, Chen Z. Safety Analysis of Metro Turnouts Based on Fuzzy FMECA. 2018 IEEE Int. Conf. Internet Things IEEE Green Comput. Commun. IEEE Cyber, Phys. Soc. Comput. IEEE Smart Data, vol. 0, 2018, p. 599–606. https://doi.org/10.1109/Cybermatics_2018.2018.00123.

[170] Ma H, Chu X, Xue D, Chen D. Identification of to-be-improved components for redesign of complex products and systems based on fuzzy QFD and FMEA. J Intell

Manuf 2019;30:623–39. https://doi.org/10.1007/s10845-016-1269-z.
[171] Olafsson S, Li X, Wu S. Operations research and data mining. Eur J Oper Res 2008;187:1429–48. https://doi.org/10.1016/j.ejor.2006.09.023.
[172] Mangasarian OL. Mathematical programming in data mining. Data Min Knowl Discov 1997;1:183–201. https://doi.org/10.1023/A:1009735908398.
[173] Jardine AKS, Lin D, Banjevic D. A review on machinery diagnostics and prognostics implementing condition-based maintenance. Mech Syst Signal Process 2006;20:1483–510. https://doi.org/10.1016/j.ymssp.2005.09.012.
[174] Ergu D, Kou G, Peng Y, Shi Y, Shi Y. The analytic hierarchy process: Task scheduling and resource allocation in cloud computing environment. J Supercomput 2013;64:835–48. https://doi.org/10.1007/s11227-011-0625-1.
[175] Crespo Marquez A, Gupta JND. Contemporary maintenance management: Process, framework and supporting pillars. Omega 2006;34:313–26. https://doi.org/10.1016/j.omega.2004.11.003.
[176] Agrawal R, Imieliński T, Swami A. Mining association rules between sets of items in large databases. ACM SIGMOD Rec 1993. https://doi.org/10.1145/170036.170072.
[177] Han J, Cheng H, Xin D, Yan X. Frequent pattern mining: Current status and future directions. Data Min Knowl Discov 2007. https://doi.org/10.1007/s10618-006-0059-1.
[178] Mavrotas G. Effective implementation of the ε-constraint method in Multi-Objective Mathematical Programming problems. Appl Math Comput 2009;213:455–65. https://doi.org/10.1016/j.amc.2009.03.037.
[179] Mavrotas G, Florios K. An improved version of the augmented s-constraint method (AUGMECON2) for finding the exact pareto set in multi-objective integer programming problems. Appl Math Comput 2013;219:9652–69. https://doi.org/10.1016/j.amc.2013.03.002.
[180] Aghaei J, Amjady N, Shayanfar HA. Multi-objective electricity market clearing considering dynamic security by lexicographic optimization and augmented epsilon constraint method. Appl Soft Comput J 2011;11:3846–58.

https://doi.org/10.1016/j.asoc.2011.02.022.

[181] Yu H, Solvang WD. An Improved Multi-Objective Programming with Augmented ε-Constraint Method for Hazardous Waste Location-Routing Problems. MdpiCom 2016. https://doi.org/10.3390/ijerph13060548.

[182] Ehrgott M, Gandibleux X. A survey and annotated bibliography of multiobjective combinatorial optimization. OR Spektrum 2000;22:425–60. https://doi.org/10.1007/s002910000046.

[183] Jain S, Hentenryck P Van. Large Neighborhood Search for Dial-a-Ride Problems Using Twitter for rapid assessment of disaster damage View project Empirical Model Learning View project Large Neighborhood Search For Dial-a-Ride Problems. Springer 2011;6876 LNCS:400–13. https://doi.org/10.1007/978-3-642-23786-7_31.

[184] Goel V, Furman KC, Song JH, El-Bakry AS. Large neighborhood search for LNG inventory routing. J Heuristics 2012;18:821–48. https://doi.org/10.1007/s10732-012-9206-6.

[185] Sinclair K, Cordeau JF, Laporte G. Improvements to a large neighborhood search heuristic for an integrated aircraft and passenger recovery problem. Eur J Oper Res 2014;233:234–45. https://doi.org/10.1016/j.ejor.2013.08.034.

[186] Pisinger D, Ropke S. Large Neighborhood Search, Springer, Boston, MA; 2010, p. 399–419. https://doi.org/10.1007/978-1-4419-1665-5_13.

[187] Aksen D, Kaya O, Salman F, Operational ÖT-EJ of, 2014 U. An adaptive large neighborhood search algorithm for a selective and periodic inventory routing problem. Elsevier 2014.

[188] Shaw P. Using constraint programming and local search methods to solve vehicle routing problems. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 1520, Springer Verlag; 1998, p. 417–31. https://doi.org/10.1007/3-540-49481-2_30.

[189] Schaus P, Hartert R. Multi-objective large neighborhood search. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 8124 LNCS, 2013, p. 611–27. https://doi.org/10.1007/978-3-

642-40627-0_46.
[190] Oddi A, Rasconi R, Cesta A. A multi-objective large neighborhood search methodology for scheduling problems with energy costs. Proc. - Int. Conf. Tools with Artif. Intell. ICTAI, vol. 2016- January, IEEE Computer Society; 2016, p. 453–60. https://doi.org/10.1109/ICTAI.2015.74.
[191] Schott JR. Fault tolerant design using single and multicriteria genetic algorithm optimization. 1995.
[192] Knoke D, Yang S. Network fundamentals. MethodsSagepubCom n.d.
[193] Brandes U. A faster algorithm for betweenness centrality. J Math Sociol 2001;25:163–77. https://doi.org/10.1080/0022250X.2001.9990249.
[194] Scott J, Carrington P. The SAGE Handbook of Social Network Analysis. SAGE Publications Ltd; 2015. https://doi.org/10.4135/9781446294413.
[195] Marinakis V, Doukas H, Xidonas P, Zopounidis C. Multicriteria decision support in local energy planning: An evaluation of alternative scenarios for the Sustainable Energy Action Plan. Omega (United Kingdom) 2017;69:1–16. https://doi.org/10.1016/j.omega.2016.07.005.
[196] US Military Standard. MIL-STD-1629a. Procedures for performing a failure mode, effect and criticality analysis. 1980.
[197] US Military Standard. MIL-STD-1629A. Procedures for performing a failure mode, effect and criticality analysis. 1983.
[198] Tan PN, Kumar V, Srivastava J. Selecting the right objective measure for association analysis. Inf Syst 2004;29:293–313. https://doi.org/10.1016/S0306-4379(03)00072-3.
[199] Hand DJ. Data mining: Statistics and more? Am Stat 1998;52:112–8. https://doi.org/10.1080/00031305.1998.10480549.

Appendix

Appendix A

Excerpt of the rules extracted for topping sub-plant, considering a one-month timeframe and a slow-down stoppage.

Sub-plant	Time interval	Stoppage	Premises	Conclusion	Support	Confidence
Topping	1 month	SLOW_DOWN	accoppiamento	controllore	0.837838	1
Topping	1 month	SLOW_DOWN	controllore	accoppiamento	0.837838	0.885714
Topping	1 month	SLOW_DOWN	coibentazione	controllore	0.621622	0.958333
Topping	1 month	SLOW_DOWN	tenuta	controllore	0.621622	0.92
Topping	1 month	SLOW_DOWN	controllore	tenuta	0.621622	0.657143
Topping	1 month	SLOW_DOWN	controllore	coibentazione	0.621622	0.657143
Topping	1 month	SLOW_DOWN	coibentazione	accoppiamento	0.567568	0.875
Topping	1 month	SLOW_DOWN	tenuta	accoppiamento	0.567568	0.84
Topping	1 month	SLOW_DOWN	accoppiamento	tenuta	0.567568	0.677419
Topping	1 month	SLOW_DOWN	accoppiamento	coibentazione	0.567568	0.677419
Topping	1 month	SLOW_DOWN	indicatore	controllore	0.486486	1
Topping	1 month	SLOW_DOWN	indicatore	coibentazione	0.486486	1
Topping	1 month	SLOW_DOWN	coibentazione	indicatore	0.486486	0.75
Topping	1 month	SLOW_DOWN	tracciatura	controllore	0.432432	1
Topping	1 month	SLOW_DOWN	illuminazione	controllore	0.432432	1
Topping	1 month	SLOW_DOWN	tracciatura	accoppiamento	0.432432	1
Topping	1 month	SLOW_DOWN	illuminazione	coibentazione	0.432432	1
Topping	1 month	SLOW_DOWN	indicatore	accoppiamento	0.432432	0.888889
Topping	1 month	SLOW_DOWN	coibentazione	illuminazione	0.432432	0.666667
Topping	1 month	SLOW_DOWN	presa campione	controllore	0.405405	0.9375
Topping	1 month	SLOW_DOWN	illuminazione	accoppiamento	0.405405	0.9375
Topping	1 month	SLOW_DOWN	presa campione	coibentazione	0.405405	0.9375
Topping	1 month	SLOW_DOWN	coibentazione	presa campione	0.405405	0.625
Topping	1 month	SLOW_DOWN	amperometro	controllore	0.378378	1
Topping	1 month	SLOW_DOWN	allarme	controllore	0.378378	1
Topping	1 month	SLOW_DOWN	-	controllore	0.378378	1
Topping	1 month	SLOW_DOWN	amperometro	coibentazione	0.378378	1

Topping	1 month	SLOW_DOWN	rilevatore	controllore	0.351351	1
Topping	1 month	SLOW_DOWN	rilevatore	coibentazione	0.351351	1
Topping	1 month	SLOW_DOWN	rilevatore	presa campione	0.351351	1
Topping	1 month	SLOW_DOWN	amperometro	accoppiamento	0.351351	0.928571
Topping	1 month	SLOW_DOWN	amperometro	tenuta	0.351351	0.928571
Topping	1 month	SLOW_DOWN	allarme	tenuta	0.351351	0.928571
Topping	1 month	SLOW_DOWN	allarme	coibentazione	0.351351	0.928571
Topping	1 month	SLOW_DOWN	-	coibentazione	0.351351	0.928571
Topping	1 month	SLOW_DOWN	amperometro	allarme	0.351351	0.928571
Topping	1 month	SLOW_DOWN	allarme	amperometro	0.351351	0.928571
Topping	1 month	SLOW_DOWN	presa campione	rilevatore	0.351351	0.8125
Topping	1 month	SLOW_DOWN	area	controllore	0.324324	1
Topping	1 month	SLOW_DOWN	dreno	coibentazione	0.324324	1
Topping	1 month	SLOW_DOWN	rilevatore	indicatore	0.324324	0.923077
Topping	1 month	SLOW_DOWN	allarme	accoppiamento	0.324324	0.857143
Topping	1 month	SLOW_DOWN	presa campione	accoppiamento	0.324324	0.75
Topping	1 month	SLOW_DOWN	presa campione	tenuta	0.324324	0.75
Topping	1 month	SLOW_DOWN	presa campione	indicatore	0.324324	0.75
Topping	1 month	SLOW_DOWN	indicatore	presa campione	0.324324	0.666667
Topping	1 month	SLOW_DOWN	indicatore	rilevatore	0.324324	0.666667
Topping	1 month	SLOW_DOWN	dreno	controllore	0.297297	0.916667
Topping	1 month	SLOW_DOWN	dreno	-	0.297297	0.916667
Topping	1 month	SLOW_DOWN	rilevatore	accoppiamento	0.297297	0.846154
Topping	1 month	SLOW_DOWN	-	accoppiamento	0.297297	0.785714
Topping	1 month	SLOW_DOWN	amperometro	presa campione	0.297297	0.785714
Topping	1 month	SLOW_DOWN	allarme	presa campione	0.297297	0.785714
Topping	1 month	SLOW_DOWN	-	presa campione	0.297297	0.785714
Topping	1 month	SLOW_DOWN	-	dreno	0.297297	0.785714
Topping	1 month	SLOW_DOWN	illuminazione	indicatore	0.297297	0.6875
Topping	1 month	SLOW_DOWN	presa campione	amperometro	0.297297	0.6875
Topping	1 month	SLOW_DOWN	presa campione	allarme	0.297297	0.6875
Topping	1 month	SLOW_DOWN	presa campione	-	0.297297	0.6875
Topping	1 month	SLOW_DOWN	indicatore	illuminazione	0.297297	0.611111
Topping	1 month	SLOW_DOWN	ausiliare	controllore	0.27027	1
Topping	1 month	SLOW_DOWN	ausiliare	accoppiamento	0.27027	1
Topping	1 month	SLOW_DOWN	ausiliare	tracciatura	0.27027	1

Topping	1 month	SLOW_DOWN	ausiliare	livello	0.27027	1
Topping	1 month	SLOW_DOWN	livello	controllore	0.27027	0.833333
Topping	1 month	SLOW_DOWN	livello	accoppiamento	0.27027	0.833333
Topping	1 month	SLOW_DOWN	area	accoppiamento	0.27027	0.833333
Topping	1 month	SLOW_DOWN	livello	tenuta	0.27027	0.833333
Topping	1 month	SLOW_DOWN	livello	tracciatura	0.27027	0.833333
Topping	1 month	SLOW_DOWN	livello	ausiliare	0.27027	0.833333
Topping	1 month	SLOW_DOWN	rilevatore	amperometro	0.27027	0.769231
Topping	1 month	SLOW_DOWN	-	tenuta	0.27027	0.714286
Topping	1 month	SLOW_DOWN	amperometro	-	0.27027	0.714286
Topping	1 month	SLOW_DOWN	-	amperometro	0.27027	0.714286
Topping	1 month	SLOW_DOWN	amperometro	rilevatore	0.27027	0.714286
Topping	1 month	SLOW_DOWN	allarme	-	0.27027	0.714286
Topping	1 month	SLOW_DOWN	-	allarme	0.27027	0.714286
Topping	1 month	SLOW_DOWN	tracciatura	livello	0.27027	0.625
Topping	1 month	SLOW_DOWN	tracciatura	ausiliare	0.27027	0.625

Appendix B

List of the components' ID and the corresponding name.

ID	Component name	ID	Component name
C1	Undefined Component	C24	Joint
C2	Coupling	C25	Lighting
C3	Alarm	C26	Indicator
C4	Ammeter	C27	Liquid Level Indicator
C5	Area	C28	Level Switch
C6	Auxiliary	C29	Lubrication
C7	Shovel	C30	Engine
C8	Keg	C31	Shovels
C9	Battery	C32	Oil Seal
C10	Burner	C33	Flooring
C11	Bypass	C34	Sampling Valve
C12	Strap	C35	Button Panel
C13	Insulation	C36	Refrigerant
C14	Condensation Indicator	C37	Detector
C15	Controller	C38	Blower
C16	Bearing	C39	Instrumentation
C17	Caliber Disc	C40	Sealing Device
C18	Drainer	C41	Tracing
C19	Ecos	C42	Transmitting Device
C20	Electrode	C43	Overfilling Indicator
C21	Tube Bundle	C44	Pipeline
C22	Filter	C45	Valve
C23	Fittings	C46	Shifter

Appendix C

Excerpt of the FMECA of the on-shore/off-shore platform

In Table C, an excerpt of the FMECA document regarding two different items is shown to provide an example of the basic information reported. The table reports the ID number of the item analyzed, followed by its synthetic description that makes it understandable, and the list of failure modes. In addition, the effects deriving from the failure modes are reported. A specification is made in considering the effects, individuating three different levels:

- Equipment effect, describes the failure at the item level;
- Functional effect, describes the failure at a facility functional level (i.e., inside the same unit);
- System effect, describes the effect on the whole plant (i.e., more than one unit is involved);.

Moreover, the annual frequency class, severity levels and their combination (ICA, ICS, ICE) are reported, as well as the criticality in terms of assets, environment and safety.The overall criticality failure mode CFM (for each failure mode) and the item criticality CI (for each item) are also expressed.

Table C Excerpt of the data coming from the FMECA process

TAG Number	Item description	Failure Mode Description	Equipment effect	Functional effect	System effect	AFC	Severity Safety	Severity Asset	Severity Environment	ICS	ICA	ICE	CSS	CSA	CSE	CFM	CI
0130361000SSSV070	Flow Control Valve	Fail to close on demand	n.a.	n.a.	n.a.	C	0	0	0	C0	C0	C0	3	3	3	3	2
	Flow Control Valve	Fail to open on demand	No oil flow	no oil flow to unit 200 Oil Production Separation	no oil flow to on shore plant	C	0	0	0	C0	C0	C0	3	3	3	3	
	Flow	Fail to	n.a.	n.a.	n.a.	C	0	0	0	C	C	C	3	3	3	3	

Control Valve	regulate								0	0	0				
Flow Control Valve	External leakage process medium	Gas & Oil Leakage	Reduction of delivered oil & GAS to Unit 200 Oil Production Separation	Reduction of delivered oil & gas to On shore plant	C	1	0	0	C1	C0	C0	3	3	3	3
Flow Control Valve	External leakage process medium	Gas & Oil Leakage	Reduction of delivered oil &	Reduction of delivered oil &	C	1	2	0	C1	C2	C0	3	2	3	2

				GAS to Unit 200 Oil Production Separation	gas to On shore plant												
0130361000SSV071	Flow Control Valve	Fail to close on demand	n.a.	n.a.	n.a.	C	0	0	0	C0	C0	C0	3	3	3	3	2
	Flow Control Valve	Fail to open on demand	No oil flow	no oil flow to unit 200 Oil Production Separatio	no oil flow to on shore plant	C	0	0	0	C0	C0	C0	3	3	3	3	

n

Flow Control Valve	Fail to regulate	n.a.	n.a.	n.a.	C	0	0	0	C0	C0	C0	3	3	3	3
Flow Control Valve	External leakage process medium	Gas & Oil Leakage	Reduction of delivered oil & GAS to Unit 200 Oil Production Separation	Reduction of delivered oil & gas to On shore plant	C	0	0	0	C0	C0	C0	3	3	3	3

Flow Control Valve	External leakage process medium	Gas & Oil Leakage	Reduction of delivered oil & GAS to Unit 200 Oil Production Separation	Reduction of delivered oil & gas to On shore plant	C	1	2	0		C1	C2	C0	3	2	3	2