



A predictive association rule-based maintenance policy to minimize the probability of breakages: application to an oil refinery

Sara Antomarioni¹ · Ornella Pisacane² · Domenico Potena² · Maurizio Bevilacqua¹ · Filippo Emanuele Ciarapica¹ · Claudia Diamantini²

Received: 2 August 2018 / Accepted: 16 April 2019 / Published online: 10 May 2019
© Springer-Verlag London Ltd., part of Springer Nature 2019

Abstract

Effective maintenance policies can support companies to deal with process interruptions and consequently, to prevent significant profit losses. Moreover, the proliferation of structured and unstructured data due to production plants validates the application of knowledge discovery in databases techniques to increase processes' reliability. In this paper, an innovative maintenance policy is proposed. It aims at both predicting components breakages through association rule mining and determining the optimal set of components to repair in order to improve the overall plant's reliability, under time and budget constraints. An experimental campaign is carried out on a real-life case study concerning an oil refinery plant. Finally, numerical results are discussed considering different blockage categories and number of components and by varying some significant input parameters.

Keywords Predictive maintenance · Association rules · Integer linear programming · Knowledge discovery in databases · Industry 4.0

1 Introduction

Production efficiency plays a key role in the development of any company. It 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 have negative effects on the whole system and therefore, on profits [32]. The more complex the system, the higher the number of components potentially subject to breakage is.

In the Industry 4.0 era, sensors are used by companies to gather a huge amount of data related to production, maintenance events, and components' breakages, for instance. Knowledge discovery in databases (KDD) techniques can significantly support the automatic extraction of valid, useful, and unknown relations from data [22]. In particular, the use of KDD techniques can successfully support a maintenance policy, especially in the case of process industries. Indeed, several variables, i.e., flow rates, liquid level, chemical properties, have to be measured and controlled. Consequently, a large amount of data has to be managed, making it suitable for the KDD techniques' application. The main contributions of this work can be summarized as follows:

- A predictive optimization-based maintenance policy, under the assumption that a component fails, based on:
 1. The definition of association rules (ARs) aimed at describing relationships among components' breakages

✉ Sara Antomarioni
s.antomarioni@pm.univpm.it

Ornella Pisacane
o.pisacane@univpm.it

Domenico Potena
d.potena@univpm.it

Maurizio Bevilacqua
m.bevilacqua@univpm.it

Filippo Emanuele Ciarapica
f.e.ciarapica@univpm.it

Claudia Diamantini
c.diamantini@univpm.it

¹ Dipartimento di Ingegneria Industriale e Scienze Matematiche, Università Politecnica delle Marche, Via B. Bianche, 12, 60131, Ancona, Italy

² Dipartimento di Ingegneria dell'Informazione, Università Politecnica delle Marche, Via B. Bianche, 12, 60131, Ancona, Italy

2. The formulation of an integer linear programming model aimed at selecting the set of components to repair in order to improve the overall robustness to breakages of the plant, respecting both the total repair time available and the budget given
- An experimental campaign carried out on a real case study of an oil refinery
 - A detailed sensitivity analysis on some parameters of the mathematical model

The remainder of the paper is organized as follows. Section 2 overviews the contributions presented in the literature with reference to both AR mining and mathematical optimization for maintenance policies. Section 3 describes the solution approach proposed in this study, while Section 4 presents the case study. In Section 5, numerical results are discussed and a detailed sensitivity analysis is presented varying some significant input parameters. Section 6 concludes the paper and draws future research directions worthy of investigation.

2 Literature review

2.1 Association rule mining in production management

All the activities of any company, e.g., production scheduling and operations monitoring, generate data to analyze during the decision-making phase. Thus, KDD, an interdisciplinary process for extracting knowledge from huge amount of data [19], can support companies throughout the transformation of data into value. In particular, AR mining is a valuable research area of KDD and can be successfully used for effectively representing relationships among data [56]. According to Wang [51], it can be considered for performing a predictive data analysis, since it relates a specific variable to others included in the same dataset. As stated by Buddhakulsomsiri et al. [11], 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, a variety of applications exists, ranging from customers' buying habits to product design specifications, as well as production process control [16]. For instance, Rygielski et al. [44] indicate AR mining as a valuable methodology for behavior description in retail, banking, telecommunication, and marketing fields, while Chen [14] applies it for group-technology cell definition, presenting quality results in both small- and large-sized instances. Agard and Kusiak [2] apply AR mining with the aim of analyzing the most convenient sub-assemblies to produce

in advance on the basis of the orders previously received from customers. Bevilacqua and Ciarapica [8] propose an AR-based methodology for the evaluation of human-related practices on high-risk situation in an oil refinery plant. Wang et al. [53] use AR mining in manufacturing process planning combining it with variable precision rough set and fuzzy clustering.

As observed by Harding et al. [27], data mining techniques, in general, are often applied to breakages and malfunctioning detection. For example, Chen et al. [15] propose a defect detective model based on ARs, aiming at discovering the relations between machines and products' flaws. In Da Cunha et al. [17], an application of AR mining to discover the sequences causing breakages in an assembly process is presented, showing that the method leads to a product quality improvement. Also, Kamsu-Foguem et al. [30] use AR mining for extracting information on fault causes in a drilling process. In Martínez-de Pisón et al. [41], it is shown that applying AR-based algorithms can guarantee a significant improvement from a production rate point of view. Djatna and Alitu [21] apply AR mining in a total productive maintenance strategy, considering a wooden door manufacturing industry. This way, an increment of both the time and the cost effectiveness of the company can be guaranteed.

2.2 Mathematical optimization-based approached for planning maintenance activities

Defining the best maintenance policy can become complex and, at the same time, significant for improving the performances 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 [18]. Moreover, optimization methods can be applied in solving problems where a large amount of data is used and/or requiring real-time decisions [9]. For example, Alkamis and Yellen [3] 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. [42] propose a mixed integer linear programming (MILP) model for the simultaneous design, production, and maintenance planning. Similarly, Goel et al. [24] develop a MILP model integrating design, production, and maintenance plan, focused on improving the operational availability at the design stage through the selection of a more reliable equipment. As noted by Alrabghi and Tiwari [5], 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 [4] apply a simulation-optimization approach for a flow shop

scheduling problem subject to maintenance constraints, due dates, and system availability. Moreover, simulation-optimization approaches can be also applied to deal with both the inventory control and the maintenance planning [43, 45]. The combination of preventive maintenance and the statistical process control can also be addressed through simulation-optimization approaches as in Cassady et al. [12], as well as maintenance scheduling and production control [23]. The use of combining simulation techniques with optimization ones is also investigated in Nedić et al. [38] in which the hydraulic systems used in forestry equipment are analyzed. Tagaras [49] formulates a combined model for process control and maintenance activities, under the hypothesis of a Markovian distribution deterioration. Moreover, the analytical hierarchy process is combined with the goal programming for centrifugal pumps maintenance in a refinery plant [7]. Lee et al. [33] propose a model to optimize the jobs 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. Kenné et al. [31] develop a near-optimal policy using numerical techniques for production planning and corrective maintenance interventions scheduling in a manufacturing system. The work of Vilarinho et al. [50] 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. [35] 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. [29] formulate an optimization model for routing and scheduling offshore wind turbines maintenance aiming at the total 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. In Marseguerra et al. [34], for example, a genetic algorithm is applied to identify the optimal degradation level for maintenance execution in a multi-component system, in order to simultaneously minimize the cost and maximize the system's reliability. Laggoune et al. [32], instead, formulate a model for the reduction of the whole system down-times and maintenance costs through the preventive replacement of groups of components. Xia et al. [55] develop a maintenance procedure to reduce the total maintenance costs of the production system, scheduling the optimal time windows for periodic interventions, while in Chalabi et al. [13], a particle swarm-based optimization approach is applied aimed to the minimization of the total maintenance cost and the maximization of the process availability. The use of meta-heuristics for controlling complex systems, especially

for the gain tuning, is not new. For example, the particle swarm optimization-based approach of Nedic et al. [36], the firefly algorithm proposed by Nedic et al. [37], the parameter search scheme based on the bat algorithm designed by Stojanovic et al. [47], and the parameter search scheme based on the cuckoo algorithm of Stojanovic et al. [48] are used to improve the tracking accuracy. Wang and Liu [52] 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. [25] 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 [46] propose a cost-effective maintenance strategy for both reducing the interruption of systems operating conditions and limiting the maintenance costs. In the literature review proposed by Ding and Kamaruddin [20], 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. [54] 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 [28], 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.

Although integrating data mining techniques with those provided by operations research and specifically, by mathematical programming, is not a new topic (e.g., [40] and [39]), to the best of our knowledge, our work represents the first contribution in which they are combined to each other for defining a predictive maintenance policy.

3 Solution approach

In this section, we describe the solution approach proposed for defining a new predictive optimization-based maintenance policy. For this purpose, we firstly describe the methodology applied for deriving the ARs used in the maintenance policy (Section 3.1). Then, we introduce the mathematical model formulated with the aim of selecting the components to be maintained, minimizing the breakages' probability, under budget and time constraints (Section 3.2). In Section 3.3, we provide an overview of the solution approach, validated in Section 3.4. The predictive

maintenance policy proposed is mainly based on the integration of AR mining and optimization techniques. Indeed, our proposal is aimed to define the optimal maintenance plan of a set of components in a plant. In fact, in large process industry, 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 components' breakage. Given the plants' 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 productive plant is often implemented through a sequence of activities performed by specific sub-plants (refer to Section 4). Without loss of generality, hereafter, we focus on a sub-plant at a time. In fact, our 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 a predictive maintenance depends on both the time available for maintenance planning and their repair cost. To this purpose, a mathematical model is formulated with the aim of selecting the optimal set of components to predictively maintain under time and budget constraints, maximizing the overall plant's reliability (i.e., minimizing the probability of future breakages).

3.1 Association rule mining

The AR mining aims at individuating interesting and hidden relations in wide datasets, in order to support decision-making processes. The notation and the assumptions used throughout the paper are introduced in the following. Let $D = \{d_1, d_2, \dots, d_n\}$ be the set of n boolean data, called items and $T = \{t_1, t_2, \dots, t_m\}$ be the set of m transactions. Every transaction t_i represents an item-set, i.e., a subset of items, selected from D . In our scenario, an item is a component of a plant and a transaction is the set of components that have broken in a given time interval.

Definition 1 Given two item-sets Γ and Θ , such that $\Gamma \subseteq D$, $\Theta \subseteq D$, and $\Gamma \cap \Theta = \emptyset$, an association rule is the implication $\Gamma \rightarrow \Theta$ where Γ and Θ are the body and the head of the rule, respectively.

In order to assess about the AR quality, several metrics can be applied. However, the most used ones are (a) the support and (b) the confidence. In particular:

- (a) The support of the rule $\Gamma \rightarrow \Theta$ is computed as

$$sup(\Gamma \rightarrow \Theta) = \frac{|\Gamma \cup \Theta|}{m}$$

where $|\Gamma \cup \Theta|$ represents the number of transactions in T containing the item-sets Γ and Θ . Then, the support represents the probability of finding a transaction containing both the item-sets Γ and Θ ;

- (b) The confidence of the rule $\Gamma \rightarrow \Theta$ is computed as

$$conf(\Gamma \rightarrow \Theta) = \frac{sup(\Gamma \rightarrow \Theta)}{sup(\Gamma \rightarrow True)}$$

It is the conditional probability of finding the item-set Θ , given the occurrence of the item-set Γ . In other words, it measures the rule strength.¹

The procedure we use to discover ARs is given as follows:

1. Extract the frequent item-sets, i.e., item-sets that appear more frequently in T with regard to a user-defined threshold (i.e., a minimum support). To this end, we use the FP-growth algorithm [26].
2. Define the ARs from the frequent item-sets. For each frequent item-set F , all rules $Y \rightarrow Z$ are generated such that $Y \cup Z = F$.

3.2 An integer linear programming formulation for an optimal maintenance planning

In this subsection, we describe the ILP model 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 1. 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 $c_{ij} = conf(i \rightarrow j)$ that expresses its breakage probability, given the breakage of

¹The term $sup(\Gamma \rightarrow True)$ represents the number of transactions containing the item-set Γ . With a slight abuse of notation, we refer to $sup(\Gamma \rightarrow True)$ as the support of Γ .

Table 1 Nomenclature of the input data

Parameter	Meaning
C	Set of components
c_{ij}	Confidence of the rule $i \rightarrow j$ (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

a 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:

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

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

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

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

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 (T_{max}). In such a constraint, the parameter α can be properly modified for a scenario analysis. Constraint (3) imposes a maximum budget B that can be used for maintenance. Finally, constraints (4) provide the variable nature.

3.3 Maintenance policy definition

The predictive optimization-based maintenance policy consists of the steps outlined in the following and it is briefly schematized in Fig. 1.

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

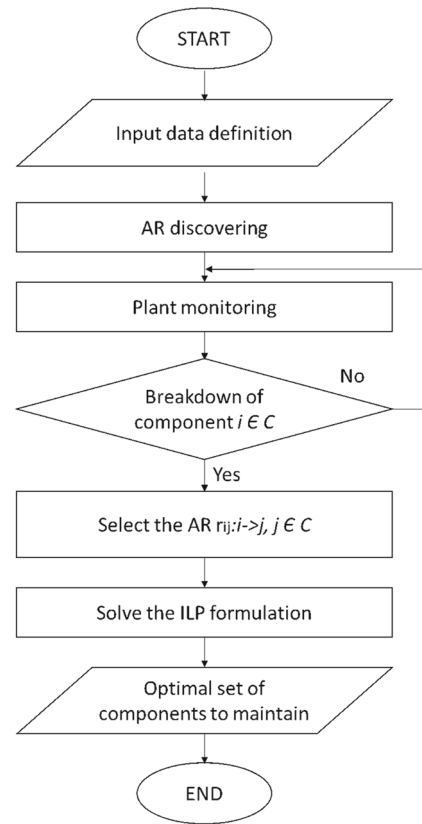


Fig. 1 Main steps of the maintenance procedure

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 $j \in C, i \neq j$.
 - (b) Solve the ILP model described in Section 3.2 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

components' 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 a 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 Antomarioni et al. [6] in which a maintenance policy, based on a 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.

3.4 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 $C_i \forall i = 1, \dots, 20$) we have to repair in a predictive way while the plant is stopped to repair \bar{C} . The repair time T_i and the repair cost $RC_i, \forall i = 1, \dots, 20$ 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 C_i , 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 C_i 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* $TM_i, \forall i = 1, \dots, 20$). By following the proposed methodology, after obtaining the confidence of each of the 20 rules of the type $\bar{C} \rightarrow C_i$, the ILP model is then solved by setting T_{max} and B equal to 350 and 10,000, respectively. For each testing month TM_i , the square *confusion matrix* CM_i of order 2 has been defined as follows:

$$CM_i = \begin{bmatrix} RR_i & RN_i \\ NR_i & NN_i \end{bmatrix}$$

where:

- RR_i denotes the number of components to be repaired in TM_i and actually selected by the ILP model.
- RN_i is the number of components to be repaired in TM_i but not selected by the ILP model.
- NR_i represents the number of components not to be repaired in TM_i but selected by the ILP model.
- NN_i counts the number of components not to be repaired in TM_i and actually not selected by the ILP model.

Then, for each TM_i , the accuracy η_i has been calculated as:

$$\eta_i = \frac{NN_i + RR_i}{NN_i + RR_i + NR_i + RN_i}$$

Then, the average accuracy has been computed over the 20 testing months. We have run 10 simulations (varying T_i , RC_i , 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., RN_i and NR_i) 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 of 321 and 3646.3, respectively. It is worth noting that 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 good estimation of the actual breakage probability, given that the breakage of the component \bar{C} occurs.

4 Application scenario: oil refinery

The proposed approach described in Section 3 is applied to a real-life case study concerning an 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 sub-plants 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) The third one, instead, is composed of *thermal cracking*, *visbreaking*, and *hydro-desulfurization* sub-plants for the production of fuel oil and bitumen.

Table 2 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 of a

Table 2 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%

maintenance policy. In addition, the high percentage of components’ breakages implies a high cost due to the reduced production capacity of the sub-plant. And, this implicitly confirms the need of implementing 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, it is necessary that the flow along this sub-plant runs smoothly. The data provided by the maintenance department of the refinery plant refer to the period from January 2001 to December 2003 and they are organized in two different databases. The former is referred to the crude oil circulating in the sub-plant and 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 in 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 (SID) 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).

Table 3 Excerpt of the integration between the two databases. The date of the blockage, the blockage category, the intervention date on the component, the component name, and the corresponding failing item are reported

BD	BC	ID	Component	Item
06/03/2001	NS	09/03/2001	Coupling	TUB1000
06/03/2001	NS	01/04/2001	Controller	LC1060A
06/03/2001	NS	20/03/2001	Piping	P1015A
06/03/2001	NS
06/03/2001	NS	29/03/2001	Valve	Valv1000

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, for each activity, it stores information about the component and the date in which the maintenance has been performed. The maintenance date is equal to or later than one of the component’s breakage. In this work, we assume that it is exactly equal to the date in which the component’s breakage occurs. During the monitored period, several blockages occurred: 21 NS blockages (103 h), 37 SID blockages (122 h), and 8 days of ShD (192 h). Moreover, 767 components required maintenance activities. In order to apply the solution proposed in Section 3, the two databases have been properly integrated by joining data regarding the blockages of the sub-plant and the components’ breakage occurred after the blockage in a defined time interval. Table 3 shows an example of the integrated database where the first two columns report the date in which the blockage occurs (BD) and its category (BC), respectively. The remaining columns refer to the maintenance activities performed on a given component. In order to extract the ARs, data reported in Table 3 are re-arranged as presented in Table 4. 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

Table 4 Excerpt of the input dataset for RapidMiner. The information reported regard the blockage date, the blockage category, the time interval, and a list of all the components monitored in the sub-plant

BD	BC	ΔT	Coupling	Controller	Valve	...
06/03/2001	NS	1 month	False	True	True	...
06/03/2001	NS	1 week	True	False	False	...
06/03/2001	NS	2 weeks	False	False	True	...
08/04/2001	SID	1 month	True	False	False	...
08/04/2001	SID	1 week	False	False	False	...
08/04/2001	SID	2 weeks	False	True	False	...

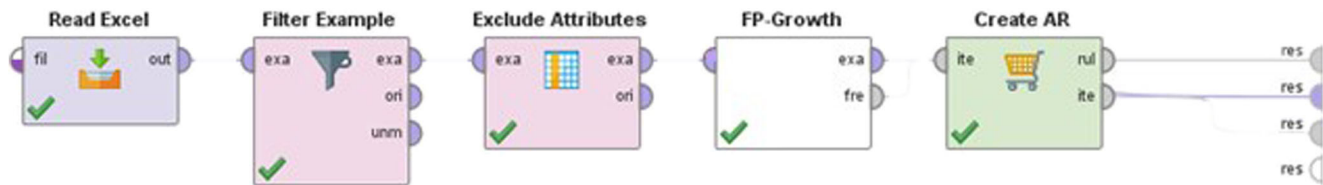


Fig. 2 View of the process implemented in RapidMiner

interval, then the value assigned is *true*, *false* otherwise. For example, the blockage occurred on April 8, 2001 is a SID: a maintenance activity is performed on the component *coupling* in a 1-month time interval. *Alarm* and *impeller* are not maintained after this blockage. According to the maintenance policy described in the previous section, when a component breakage occurs, the ARs having the broken component as their body are extracted. To this end, we use the tools provided by RapidMiner (www.rapidminer.com), a widely applied data-mining platform. In particular, Fig. 2 describes the whole process. Firstly, the integrated dataset (as represented in Table 4) 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 which 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. 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

5 Numerical results

This section describes the numerical results obtained by applying the solution approach detailed in Section 3 to the case study reported in Section 4. In the following experiments, we focus attention on the component requiring a lot 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 SID, ShD, and NS blockages. According to the privacy policy adopted by the refinery, we cannot report details about the total budget, the repair times, and the costs of the components. However, in the following experiments, we use reasonable estimated values for them.

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

5.1 Analysis on the component controller

The first example presented regards the breakage of the *controller*, since it resulted 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 T_{max} is 350 min. Finally, α is initially set to 1. Firstly, all the ARs of interest are individuated as described in Section 3. 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 body equal to *controller* are selected. In Table 5, 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 breakage 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 *pipng* (all highlighted in italics in Table 5), so that both the total repair time and the budget constraints can be respected. In this way, it can be obtained a 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 min 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 simpler way for detecting the most convenience set of components to maintenance 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 time and budget constraints.

Table 5 Association rules having support greater than min_{sup} and “controller” as body

Body	Head	Confidence	RC_{Head}	T_{Head}
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

Table 6 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

In this way, the components *coupling*, *lighting*, and *liquid level* are selected for maintenance, with a 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 a 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 have an impact on the components’ breakages. For instance, Table 6 contains the ARs related to a SID blockage. Comparing Tables 6 and 5, 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 SID blockages are the majority. The only exception is the component *Belt*, whose

support is higher than the min_{sup} only in the case of a SID. When a ShD is considered (see Table 7), 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 a ShD blockage has the highest impact on production. However, this kind of blockage is the rarest, so the related rules have a low significance.

5.1.1 Sensitivity analysis on α parameter

We carry out also a scenario analysis to study the sensitivity of the solution varying the α parameter. In particular, we

Table 7 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

define a range for it between 0.50 and 1.50 and we test different cases using an incremental step of 0.05. In Fig. 3, the values of the objective function (1) are reported as the parameter α increases and all kinds of blockages are considered. 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 $\alpha = 0.50$ ($\alpha T_{\text{max}} = 175$ min), *piping*, *liquid level*, and *lighting* are selected for maintenance planning, but the total confidence decreases of about 53% (0.914). On the contrary, increasing

the available time of the 50% ($\alpha = 1.50$, $\alpha T_{\text{max}} = 525$ min) leads to a 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 5), have not been selected since they violate the total repair time constraint.

In Table 8, for each scenario, we report the corresponding α , the total repair time (TRT) of the selected components. The third column, instead, 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 is willing to pay for increasing the total time available for maintenance.

5.1.2 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 €1000. If B ranges from €1500 to €2000, instead, the total confidence is higher (1.379) and the components selected are *coupling*, *lighting*, and *piping*. Remarkably, allowing a budget higher than €2500 is not useful since the optimal solution found remains the same: *ammeter*,

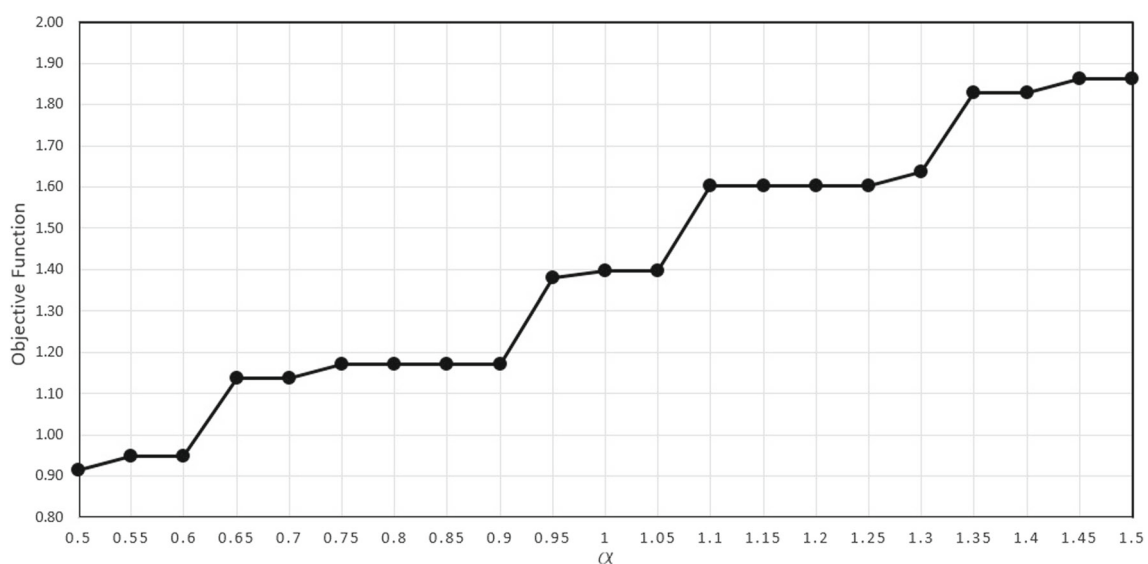
**Fig. 3** Values of the objective function (1) for different α

Table 8 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

drainer, lighting, liquid level, and piping are the selected components, while the total confidence is 1.397. Indeed, above this value, the constraint (2) becomes tighter than the constraint (3), making any variation on the budget irrelevant.

5.1.3 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, SID and ShD) and varying the α parameter. Indeed, we properly filtered the dataset in order to extract only the ARs related to each blockage category and consider the corresponding confidence values to solve the model. Figure 4 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 a SID 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 SID blockages. Indeed, ShD blockages rarely occur and after them, the number of components that have broken within ΔT is less than the one after SID blockages. A further consequence of this is obtaining ARs with very quantized confidence values (see for instance Table 7). This leads to the piecewise linear trend of the objective function in the case of ShD blockages (see Fig. 4). A similar trend is reported also in the case of NS blockages, but 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, the most of ARs involve only a component (e.g., the *controller*) and there are very few rules involving two components within ΔT after a NS blockage. Hence, these rules are characterized by very low confidence values. In particular, in the 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

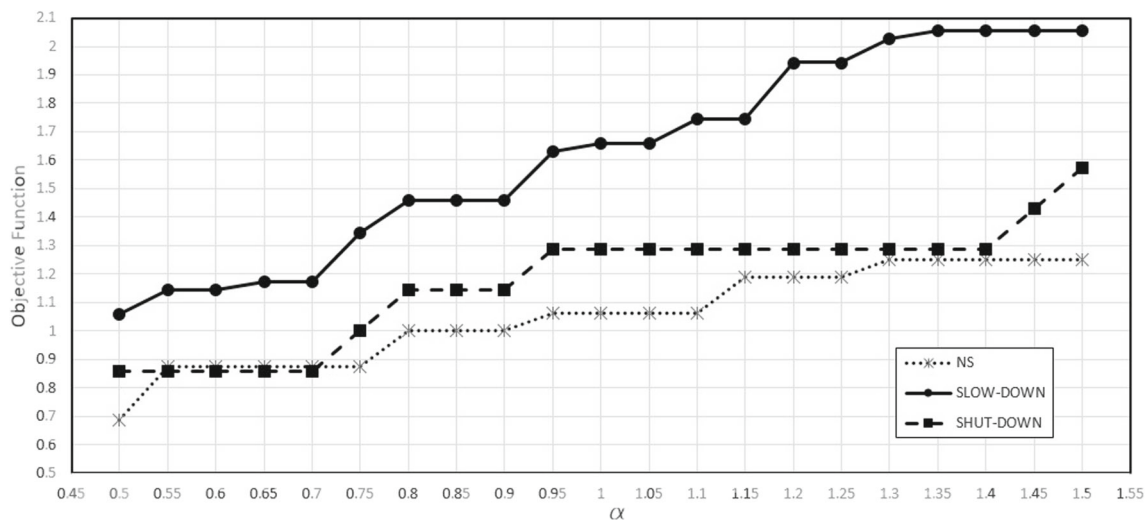


Fig. 4 Comparison between objective function values (1) for different α , discriminating the blockage category

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 a confidence by far higher (i.e., 0.714) than the aforementioned ones, but its repair time exceeds T_{\max} . Thus, increasing the repair time of the 40% does not lead any improvement on the total confidence.

5.2 Scalability analysis

As already remarked in Section 4, the proposed methodology does not require higher computational times (i.e., 1 min 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 $10 \cdot 2^{i-1}$ 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 being 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.

5.3 Discussion

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 components' breakage and 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, as presented in Section 5.1, 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 (Section 3.4) and the fact that we are focusing attention on a sub-plant at 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 (Section 3.3) 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, although, in any case, it remains reasonable (Section 5.2).

6 Conclusions and future work

The components maintenance is a critical issue in all industrial fields, specifically in the case of continuous processes since the occurrence of an event may have an influence on the rest. For this reason, in this work, an association rule (ARs)-based maintenance optimization procedure is proposed and tested on a real-life oil refinery sub-plant. In particular, it integrates the potentialities of both AR mining and

mathematical optimization. The former technique, indeed, allows extracting the existing relationships among data, while the latter, through the definition of an integer linear programming formulation, aims at selecting the most critical components to maintenance. Being able to anticipate the need for maintenance activities is the key aspect for limiting production flow interruptions and, thus, productivity losses. A wider control on components' breakages can be translated into maintenance cost savings. The proposed procedure extracts all ARs considering the maintenance activities executed within a given time interval from all the past sub-plant blockages. Then, the system is monitored and as soon as a component breaks, an ILP model is solved for selecting the optimal set of components to maintain, respecting budget and time constraints. The analysis is carried out differentiating among the three categories of blockages considered in the refinery object of the case study. The results remark that, depending on the category of blockage, different optimal sets of components are selected. In addition, a scenario analysis carried out varying the time devoted to maintenance planning allows studying the sensitivity of the solution found. For these reasons, the results obtained through the implementation of the maintenance procedure provide a valuable decision support system: indeed, the optimal set of components to maintain is presented and, through the scenario analysis, modifications on the time range devoted to maintenance planning can be taken into consideration.

Further research directions worthy of investigation could firstly concern the application of the proposed predictive optimization-based methodology to the whole refinery plant, including, this way, dependency relationships among sub-plants and, hence, a larger number of components. Regarding the optimization model, additional constraints could be introduced for taking into account both the costs due to the operators used for maintenance and their hour availability. In addition, multi-objective programming can be also used for modeling the situation in which one wants to maximize the plant's reliability and to minimize the blockage costs due to failure, simultaneously. Moreover, due to the nature of the problem addressed, stochastic programming could be also used for taking into account some aspects that cannot be known in advance. In addition, due to the complexity of the problem, meta-heuristics and/or matheuristics could be also defined to efficiently solve it. Finally, the introduction of an architecture aimed at collecting and analyzing streaming data would improve the significance of the procedure developed. Interesting application field may regard other kinds of processes, for example water recovery and purification systems, with the aim of reducing waste-water [10]. In addition, a further development may regard the inclusion in the model of performance indicators regarding the emissions, as recommended by Accorsi et al. [1], in proper contexts.

References

1. Accorsi R, Bortolini M, Gamberi M, Manzini R, Pilati F (2017) Multi-objective warehouse building design to optimize the cycle time, total cost, and carbon footprint. *Int J Adv Manuf Tech* 92(1–4):839–854. <https://doi.org/10.1007/s00170-017-0157-9>
2. Agard B, Kusiak A (2004) Data mining for subassembly selection. *J Manuf Sci E-T ASME* 126(3):627–631. <https://doi.org/10.1115/1.1763182>
3. Alkhamis TM, Yellen J (1995) Refinery units maintenance scheduling using integer programming. *Appl Math Model* 19(9):543–549. [https://doi.org/10.1016/0307-904X\(95\)00032-F](https://doi.org/10.1016/0307-904X(95)00032-F)
4. Allaoui H, Artiba A (2004) Integrating simulation and optimization to schedule a hybrid flow shop with maintenance constraints. *Comput Ind Eng* 47(4):431–450. <https://doi.org/10.1016/j.cie.2004.09.002>
5. Alrabghi A, Tiwari A (2015) State of the art in simulation-based optimisation for maintenance systems. *Comput Ind Eng* 82:167–182. <https://doi.org/10.1016/j.cie.2014.12.022>
6. Antomarioni S, Bevilacqua M, Potena D, Diamantini C (2019) Defining a data-driven maintenance policy an application to an oil refinery plant. *Int J Qual Reliab Manag.* <https://doi.org/10.1108/IJQRM-01-2018-0012>
7. Bertolini M, Bevilacqua M (2006) A combined goal programming-AHP approach to maintenance selection problem. *Reliab Eng Syst Safe* 91(7):839–848. <https://doi.org/10.1016/j.res.2005.08.006>
8. Bevilacqua M, Ciarapica FE (2018) Human factor risk management in the process industry: a case study. *Reliab Eng Syst Safe* 169:149–159. <https://doi.org/10.1016/j.res.2017.08.013>
9. Bortolini M, Ferrari E, Gamberi M, Pilati F, Faccio M (2017) Assembly system design in the industry 4.0 era: a general framework. *IFAC-PapersOnLine* 50(1):5700–5705. <https://doi.org/10.1016/j.ifacol.2017.08.1121>
10. Bortolini M, Gamberi M, Mora C, Pilati F, Regattieri A (2017) Design, prototyping, and assessment of a wastewater closed-loop recovery and purification system. *Sustainability* 9(11):1938. <https://doi.org/10.3390/su9111938>
11. Buddhakulsomsiri J, Siradeghyan Y, Zakarian A, Li X (2006) Association rule-generation algorithm for mining automotive warranty data. *Int J Prod Res* 44(14):2749–2770. <https://doi.org/10.1080/00207540600564633>
12. Cassidy CR, Bowden RO, Liew L, Pohl EA (2000) Combining preventive maintenance and statistical process control: a preliminary investigation. *Iie Trans* 32(6):471–478. <https://doi.org/10.1080/07408170008963924>
13. Chalabi N, Dahane M, Beldjilali B, Neki A (2016) Optimisation of preventive maintenance grouping strategy for multi-component series systems: particle swarm based approach. *Comput Ind Eng* 102:440–451. <https://doi.org/10.1016/j.cie.2016.04.018>
14. Chen MC (2003) Configuration of cellular manufacturing systems using association rule induction. *Int J Prod Res* 41(2):381–395. <https://doi.org/10.1080/0020754021000024184>
15. Chen WC, Tseng SS, Wang CY (2004) A novel manufacturing defect detection method using data mining approach. In: *International conference on industrial, engineering and other applications of applied intelligent systems*. Springer, pp 77–86. https://doi.org/10.1007/978-3-540-24677-0_9
16. Choudhary AK, Harding JA, Tiwari MK (2009) Data mining in manufacturing: a review based on the kind of knowledge. *J Intell Manuf* 20(5):501. <https://doi.org/10.1007/s10845-008-0145-x>
17. Da Cunha C, Agard B, Kusiak A (2006) Data mining for improvement of product quality. *Int J Prod Res* 44(18–19):4027–4041. <https://doi.org/10.1080/00207540600678904>

18. Dekker R (1996) Applications of maintenance optimization models: a review and analysis. *Reliab Eng Syst Safe* 51(3):229–240. [https://doi.org/10.1016/0951-8320\(95\)00076-3](https://doi.org/10.1016/0951-8320(95)00076-3)
19. Diamantini C, Potena D, Storti E (2013) A virtual mart for knowledge discovery in databases. *Inform Syst Front* 15(3):447–463. <https://doi.org/10.1007/s10796-012-9399-0>
20. Ding SH, Kamaruddin S (2015) Maintenance policy optimization-literature review and directions. *Int J Adv Manuf Tech* 76(5–8):1263–1283. <https://doi.org/10.1007/s00170-014-6341-2>
21. Djatna T, Alitu IM (2015) An application of association rule mining in total productive maintenance strategy: an analysis and modelling in wooden door manufacturing industry. *Procedia Manufacturing* 4:336–343. <https://doi.org/10.1016/j.promfg.2015.11.049>
22. Fayyad U, Piatetsky-Shapiro G, Smyth P (1996) From data mining to knowledge discovery in databases. *AI mag* 17(3):37. <https://doi.org/10.1609/aimag.v17i3.1230>
23. Gharbi A, Kenné JP (2005) Maintenance scheduling and production control of multiple-machine manufacturing systems. *Comput Ind Eng* 48(4):693–707. <https://doi.org/10.1016/j.cie.2004.12.007>
24. Goel HD, Grievink J, Weijnen MP (2003) Integrated optimal reliable design, production, and maintenance planning for multipurpose process plants. *Comput Chem Eng* 27(11):1543–1555. [https://doi.org/10.1016/S0098-1354\(03\)00090-5](https://doi.org/10.1016/S0098-1354(03)00090-5)
25. Hadjaissa B, Ameur K, Essounbouli N et al (2016) Bi-objective optimization of maintenance scheduling for power systems. *Int J Adv Manuf Tech* 85(5–8):1361–1372. <https://doi.org/10.1007/s00170-015-8053-7>
26. Han J, Pei J, Yin Y (2000) Mining frequent patterns without candidate generation. *SIGMOD Rec* 29(2):1–12. <https://doi.org/10.1145/335191.335372>
27. Harding J, Shahbaz M, Kusiak A et al (2006) Data mining in manufacturing: a review. *J Manuf Sci E-T ASME* 128(4):969–976. <https://doi.org/10.1115/1.2194554>
28. Ilgin MA, Tunali S (2007) Joint optimization of spare parts inventory and maintenance policies using genetic algorithms. *Int J Adv Manuf Tech* 34(5–6):594–604. <https://doi.org/10.1007/s00170-006-0618-z>
29. Irawan CA, Ouelhadj D, Jones D, Stålhane M, Sperstad IB (2017) Optimisation of maintenance routing and scheduling for offshore wind farms. *Eur J Oper Res* 256(1):76–89. <https://doi.org/10.1016/j.ejor.2016.05.059>
30. Kamsu-Foguem B, Rigal F, Mauget F (2013) Mining association rules for the quality improvement of the production process. *Expert Syst Appl* 40(4):1034–1045. <https://doi.org/10.1016/j.eswa.2012.08.039>
31. Kenné JP, Boukas E, Gharbi A (2003) Control of production and corrective maintenance rates in a multiple-machine, multiple-product manufacturing system. *Math Comput Model* 38(3–4):351–365. [https://doi.org/10.1016/S0895-7177\(03\)90093-8](https://doi.org/10.1016/S0895-7177(03)90093-8)
32. Laggoune R, Chateaneuf A, Aissani D (2010) Impact of few failure data on the opportunistic replacement policy for multi-component systems. *Reliab Eng Syst Safe* 95(2):108–119. <https://doi.org/10.1016/j.ress.2009.08.007>
33. Lee HT, Yang DL, Yang SJ (2013) Multi-machine scheduling with deterioration effects and maintenance activities for minimizing the total earliness and tardiness costs. *Int J Adv Manuf Tech* 66(1–4):547–554. <https://doi.org/10.1007/s00170-012-4348-0>
34. Marseguerra M, Zio E, Podofilini L (2002) Condition-based maintenance optimization by means of genetic algorithms and monte carlo simulation. *Reliab Eng Syst Safe* 77(2):151–165. [https://doi.org/10.1016/S0951-8320\(02\)00043-1](https://doi.org/10.1016/S0951-8320(02)00043-1)
35. Mokhtari H, Mozdgir A, Kamal Abadi IN (2012) A reliability/availability approach to joint production and maintenance scheduling with multiple preventive maintenance services. *Int J Prod Res* 50(20):5906–5925. <https://doi.org/10.1080/00207543.2011.637092>
36. Nedic N, Prsic D, Dubonjic L, Stojanovic V, Djordjevic V (2014) Optimal cascade hydraulic control for a parallel robot platform by PSO. *Int Journal Adv Manuf Tech* 72(5–8):1085–1098. <https://doi.org/10.1007/s00170-014-5735-5>
37. Nedic N, Stojanovic V, Djordjevic V (2015) Optimal control of hydraulically driven parallel robot platform based on firefly algorithm. *Nonlinear Dynam* 82(3):1457–1473. <https://doi.org/10.1007/s11071-015-2252-5>
38. Nedić N, Pršić D, Fragassa C, Stojanović V, Pavlović A (2017) Simulation of hydraulic check valve for forestry equipment. *Int J Heavy Veh Syst* 24(3):260–276. <https://doi.org/10.1504/IJHVS.2017.084875>
39. Olafsson S, Li X, Wu S (2008) Operations research and data mining. *Eur J Oper Res* 187(3):1429–1448. <https://doi.org/10.1016/j.ejor.2006.09.023>
40. Pardalos PM, Hansen P (2008) Data mining and mathematical programming, vol 45. American Mathematical Soc. <https://doi.org/10.1023/A:1009735908398>
41. Martínez-de Pisón FJ, Sanz A, Martínez-de Pisón E, Jiménez E, Conti D (2012) Mining association rules from time series to explain failures in a hot-dip galvanizing steel line. *Comput Ind Eng* 63(1):22–36. <https://doi.org/10.1016/j.cie.2012.01.013>
42. Pistikopoulos EN, Vassiliadis CG, Papageorgiou LG (2000) Process design for maintainability: an optimization approach. *Comput Chem Eng* 24(2–7):203–208. [https://doi.org/10.1016/S0098-1354\(00\)00514-7](https://doi.org/10.1016/S0098-1354(00)00514-7)
43. Rezg N, Xie X, Mati Y (2004) Joint optimization of preventive maintenance and inventory control in a production line using simulation. *Int J Prod Res* 42(10):2029–2046. <https://doi.org/10.1080/00207540310001638235>
44. Rygielski C, Wang JC, Yen DC (2002) Data mining techniques for customer relationship management. *Technol Soc* 24(4):483–502. <https://doi.org/10.1088/1742-6596/910/1/012021>
45. Sarker R, Haque A (2000) Optimization of maintenance and spare provisioning policy using simulation. *Appl Math Model* 24(10):751–760. [https://doi.org/10.1016/S0307-904X\(00\)00011-1](https://doi.org/10.1016/S0307-904X(00)00011-1)
46. Shafiee M (2017) Maintenance optimization and inspection planning of wind energy assets: models, methods and strategies. *Reliab Eng Syst Safe*. <https://doi.org/10.1016/j.ress.2017.10.025>
47. Nedic N (2016) A nature inspired parameter tuning approach to cascade control for hydraulically driven parallel robot platform. *J Optimiz Theory App* 168(1):332–347. <https://doi.org/10.1007/s10957-015-0706-z>
48. Stojanovic V, Nedic N, Prsic D, Dubonjic L, Djordjevic V (2016) Application of cuckoo search algorithm to constrained control problem of a parallel robot platform. *Int J Adv Manuf Tech* 87(9–12):2497–2507. <https://doi.org/10.1007/s00170-016-8627-z>
49. Tagaras G (1988) An integrated cost model for the joint optimization of process control and maintenance. *J Oper Res Soc* 39(8):757–766. <https://doi.org/10.1057/jors.1988.131>
50. Vilarinho S, Lopes I, Oliveira JA (2017) Preventive maintenance decisions through maintenance optimization models: a case study. *Procedia Manufacturing* 11:1170–1177. <https://doi.org/10.1016/j.promfg.2017.07.241>
51. Wang K (2007) Applying data mining to manufacturing: the nature and implications. *J Intell Manuf* 18(4):487–495. <https://doi.org/10.1007/s10845-007-0053-5>
52. Wang S, Liu M (2015) Multi-objective optimization of parallel machine scheduling integrated with multi-resources preventive maintenance planning. *J Manuf Syst* 37:182–192. <https://doi.org/10.1016/j.jmsy.2015.07.002>
53. Wang Z, Shao X, Zhang G, Zhu H (2005) Integration of variable precision rough set and fuzzy clustering: an application to knowledge acquisition for manufacturing process planning.

- In: International workshop on rough sets, fuzzy sets, data mining, and granular-soft computing. Springer, pp 585–593. https://doi.org/10.1007/11548706_62
54. Xia T, Xi L, Lee J, Zhou X (2011) Optimal CBPM policy considering maintenance effects and environmental condition. Int J Adv Manuf Tech 56(9–12):1181–1193. <https://doi.org/10.1007/s00170-011-3235-4>
55. Xia T, Xi L, Zhou X, Lee J (2013) Condition-based maintenance for intelligent monitored series system with independent machine failure modes. Int J Prod Res 51(15):4585–4596. <https://doi.org/10.1080/00207543.2013.775524>
56. Zhang S, Wu X (2011) Fundamentals of association rules in data mining and knowledge discovery. Wires Data Min Knowl 1(2):97–116. <https://doi.org/10.1002/widm.10>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.