International Journal of Computer Integrated Manufacturing

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tcim20

Exploiting a combined process mining approach to enhance the discovery and analysis of support processes in manufacturing

Giovanni Lugaresi, Antonio Davide Ciappina, Monica Rossi & Andrea Matta

Exploiting a combined process mining approach to enhance the discovery and analysis of support processes in manufacturing

Abstract

Recently, production plants have become very complex environments, in which the final output is the result of the favourable interplay of several processes. Successful management of such domains strictly depends on the ability to grasp the current disposition of both the physical and the managerial processes. To achieve this goal, the use of structured data-based methods has proven to be very effective. Yet, literature lacks successful applications, especially regarding *production support processes* (e.g., order acquisition, procure-to-pay, product development), which are directly connected to the overall system performance. This work proposes an approach to enabling automated mapping and controlling of production support processes starting from execution data recorded by IT systems. The limitations of existing methodologies are addressed by exploiting the combined application of two process mining algorithms: heuristic and inductive miner. This approach's managerial implications are described within the application to a real case study, in which the main phases of the procure-to-pay process (P2P) of a manufacturing company are identified and analysed automatically. The proposed approach proves its effectiveness in the context of application. The numerical results demonstrate that process mining can effectively bring tangible benefits in terms of viable improvements not only to physical production processes, but also to information flows and production support processes that are highly crucial for guaranteeing the prosperity of an enterprise, yet extremely hard to manage and control.

Keywords: Data-driven; Process Discovery; Process Mining; Procurement; P2P; Process Mapping; Combined Process Mining Approaches.

1 Introduction

In the last decade, one of the most discussed topics in literature has been understanding how data analytics tools can support managers in making reliable and effective decisions to improve companies' overall performance. The exponential growth of the amount of digital information stored by IT systems defines a promising area for value creation and frontier research in manufacturing. In order to gain competitive advantages, manufacturing companies are trying to improve their business processes by using objective insights and information enclosed in datasets generated within a production environment (Kozjek et al., 2020). Most of the contributions regarding data-based support for manufacturing industries have focused on the enhancement of specific activities rather than focusing on how to improve the end-to-end operational process (Erevelles et al., 2016; Waller & Fawcett, 2013). When the aim is to analyse and enhance business processes with a data driven and a holistic approach, a novel set of techniques provided by process mining (PM) can be exploited (Van Der Aalst, 2016). Process mining (PM) is a new discipline which aims to derive useful insights by investigating operational process execution logs. This discipline can be divided into three main areas: (1) *process discovery*, which aims to automatically recognize a process model that is able to describe the process behaviour observed in the execution log (Van Der Aalst, 2016); (2) *conformance checking*, which aims to analyse the relationship between the process that has been recorded in the operational execution log and the intended process behaviour as described in a process model (Carmona et al., 2018); (3) *enhancement*, which refers to a group of techniques that take an event log and an existing model as input, and produce an enhanced model as output (Van Der Aalst, 2016). PM can be considered as a fundamental support to industrial managers in better understanding their internal processes, in order to take critical decisions based on objective performance (Thiede et al., 2018).

In manufacturing companies, most managerial processes are strictly related to the production facilities. Referring to Figure 1, physical processes which are performed to produce goods interact with several steps of the overlying managerial procedures and these connections are critical for maintaining the desired production levels.

A delay in a managerial step is likely to cause a temporary blockage of productive operations, at the production site itself, at supplier warehouses, or at a customer's plant. *Production support processes*, which do not deal primarily with tangible entities such as information about supplies or production orders are strictly related with plant operations. If they are not executed correctly, they could represent a bottleneck for the company's entire value chain. Thus, it is essential that support processes be performed with the maximum efficiency and effectiveness. This paper focuses on these kinds of processes, such as procure-to-pay (P2P), quote-to-order, as well as product development. Discovery and analysis of their flow is essential for achieving efficient managerial capabilities. Traditionally for production support processes, studies which should assess the real process status and should detect how the process is executed are performed by means of interviews, workshops, or by looking at normative documents. Thus, they usually lead to the development of not fully reliable process models, on which the analysis is based on (Dempsey et al., 2016).

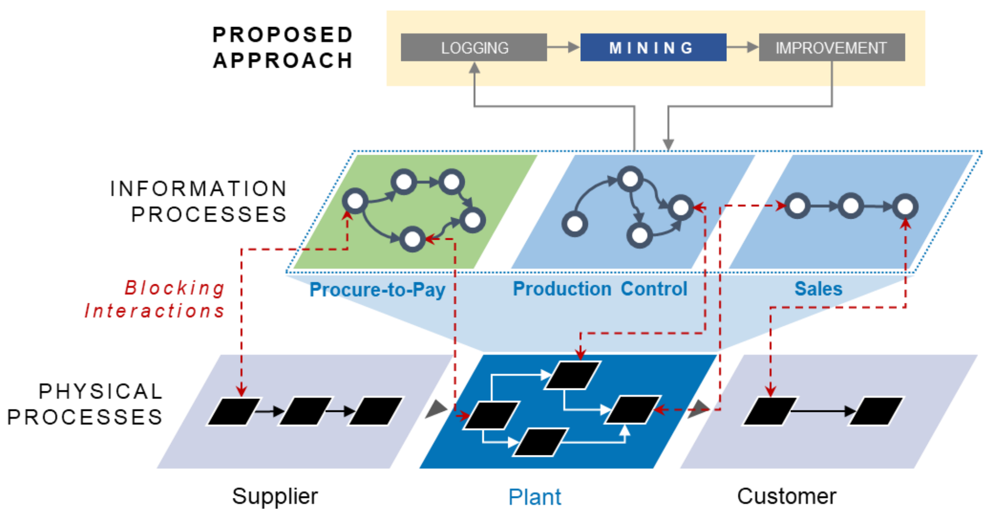

Figure 1. Proposed approach in relation to the interactions between a manufacturing company's processes.

Following these considerations, this paper aims to exploit process mining as an automated and objective tool to support executives in taking decisions, to control and improve support processes that relates directly to production performance. A PM based approach is proposed and shown in Figure 1. The goal is to exploit the event data stored by the company's information system: the support processes can be discovered and monitored automatically. Starting from such insights, production managers can develop and implement corrective actions, and improve managerial procedures. **Several PM approaches can deal with the discovery of an operational process, and there are successful implementations in the manufacturing domain (Stefanovic et al., 2021). However, most approaches focus on generating models that highlight the overall process view or the average behaviour, while assigning minor importance to accenting critical points. To overcome this issue, the proposed approach exploits the combined application of different PM algorithms, so to *guarantee the discovery of an accurate process model.***

This paper also intends to highlight how a combined PM-based approach can be used by managers to develop data-driven decisions to efficiently improving their internal business processes. To understand the practical implication of the proposed approach, this paper offers a case study focused on the purchasing process of standard items at a manufacturing company. Improvement ideas on how to correct anomalous process behaviours are identified from the analysis of the automatically generated purchasing process map. Specifically, the P2P process is analysed as an application problem for the proposed approach. P2P can be considered as a filter between the company and its suppliers since it manages the flow of incoming goods. If this process is not managed properly, the production related activities can suffer negatively. For instance, inefficient procurement related activities or not fully reliable suppliers can lead to delays in receiving the goods with consequences in the overall production schedule. In addition, since procurement process performance strongly impacts the overall company cycle time (Bag et al., 2020), it is important to continuously detect unnecessary or repetitive activities that could be avoided or automated, to make the process fast and flexible. **In this paper, the P2P problem is taken as a reference, which can prove to be an**

essential process for guaranteeing smooth management of production. Several attempts are used to discuss the role of process mining within the P2P process. (Jans et al., 2011) explored how process mining can assist in detecting fraudulent behaviour in the procurement process. In (Swinnen et al., 2012), a P2P event log was used to validate a semi-automatic process deviation analysis method based on PM. Data on goods and services procurement activities were used in (Pane et al., 2021) to evaluate the efficiency of a process discovery algorithm. (Mahendrawathi et al., 2017) performed a post-implementation review of an Enterprise Resource Planning (ERP) system by analysing the recorded data on a procurement process.

This paper continues as follows: section 2 lists the main contributions from literature that deal with process mining in manufacturing industries; section 3 shows the problem of implementing process mining in a P2P process; section 4 presents the proposed approach to detecting valuable insights from a process execution log; section 5 applies the approach to a relevant case study; section 6 highlights the main findings and conclusions.

2 Related Literature

Several contributions highlight the advantages of exploiting data to enhance manufacturing system planning and executions phases (Harding et al., 2005). (Belhadi et al., 2019) have reviewed the most important contributions on big data analytics for manufacturing processes. The authors classified existing approaches based on the main goals of the contributions reviewed: (1) descriptive analytics, (2) inquisitive analytics (i.e., root cause analysis), (3) predictive analytics, and (4) prescriptive analytics. Process Mining (PM) represents a promising field of research in manufacturing given the process perspective of the data analyses it provides (Stefanovic et al., 2021). Nevertheless, even recent reviews give limited space to PM, proving that its application as a support to production planning and control activities remains scarce.

This section lists the principal contributions from literature regarding the application of process mining techniques in manufacturing. Since PM is agnostic in relation to the application field, it has been applied to a plethora of datasets for several purposes. The contributions can be classified based on two main mining aims: (1) papers that use PM to retrieve the movement of physical objects, or information which can be linked to the work-pieces (e.g., production levels); (2) contributions that exploit PM to analyse all the business processes surrounding production environments. Paragraphs 2.1 and 2.2 include a selection of the contributions in the former group, which are classified in relation to the modelling objective: indeed, the model derived by PM can be used either (1) for *as-is analysis* of the underlying process, or (2) for the *to-be analysis*, for instance using PM to populate a discrete event simulation model.

2.1 Material-based Mining of Manufacturing Applications

The simplest yet most effective way to use PM analysing material progressions is to visualise the flows from a time perspective. Data can be analysed from various levels, such as part identifiers, activities, and resources. Aggregated indicators can also be retrieved (e.g., flow times) and displayed over graph models, allowing easy detection of bottlenecks and critical points (Sitova & Pecerska, 2020).

2.1.1 As-is Analysis (Material-based)

Given that most structured event logs contain temporal information (i.e., timestamps of the activities), several contributions exploit it to relate performance measurements such as lead time and completion times (Choueiri et al., 2020). (Intayoad & Becker, 2018) analysed event-logs from manufacturing companies and used PM to extract contextual information of processed orders, namely (1) the number of operations competing for the same assets, and (2) in case of delays, the lead-time of the finished jobs. (Park et al., 2014) developed a method for joint data extraction, clustering, and performance evaluation for a shipbuilding manufacturing system. The approach exploits process mining to extract the sequence of operations, then clustering to find similarities between production sequences. In this way, the performance of each cluster can be analysed independently using Data Envelopment Analysis (Cooper et al., 2000). The results of clustering and operation analyses can be used for planning purposes, such as scheduling or optimal resource allocation. (Lee et al., 2014) developed an approach using RFID to continuously update the parameters of a production system and exploit them to detect anomalies using a recursive algorithm that includes fuzzy association rule mining. The aim is to use PM to find a set of rules that estimate both the quantitative values of the process parameters and their interconnection with final product quality.

PM can also be used to develop probabilistic models such as Bayesian networks. The role of process mining is to feed the probabilistic models of each activity with the time series that represent the durations (Kurscheidt et al., 2015). This integration makes it possible to obtain the occurrence probability of all the operations and the process completion time (Ruschel et al., 2021).

2.1.2 To-be Analysis (Material-based Model Generation)

Recent contributions focus on the use of datasets from real manufacturing systems to construct a digital model that can reflect the system's structure and parameters, such as discrete event simulation (Rozinat et al., 2009). A digital model provides the capability to explore unobserved behaviours and perform what-if analyses. The generation of a digital model consists of several steps. The structure of the physical system can be retrieved from datasets such as event logs (Lugaresi & Matta, 2021), robot code (Farooqui et al., 2019), or directly from programmable logic controllers (Popovics & Monostori, 2013). Then, specific parameters of the system can be retrieved from event records. For instance, inter arrival times can be estimated by considering the queueing of parts at the entrance to the system (Martin et al., 2015). (Pourbafrani et al., 2020) investigated the relation between system features of use for developing a system dynamics simulation model (i.e., arrival rates and waiting times). Finally, specific system behaviour can also be inferred from data, such as control policies (Milde & Reinhart, 2019), causes of process delays (Ferreira & Vasilyev, 2015), or batching of operations (N Martin et al., 2017).

2.2 Information-based Mining of Manufacturing Applications

Process Mining can be used effectively to derive relations between information types in a structured system. This information can be either related to the physical processes themselves, or to the surrounding processes, such as production planning, procurement, and sales. In production systems, process mining can assist with both understanding the dynamics and the development of optimised procedures.

Typically, event logs are not available in the required format in information systems (Fleig et al., 2018). For instance, customer orders are typically stored in CRM systems, while design steps are recorded using PLM tools (Schuh et al., 2020). (Schuh et al., 2020) proposed a UML-based data model to describe the data in a manufacturing environment in such a way that an appropriate model can be discovered through process mining. A map of different data sources within a modern information system is provided, which covers all the steps from an initial customer order to order fulfilment.

2.2.1 As-is Analysis (Information-based)

Once a dataset of traces is created, it is possible to study the interrelations between them at a system level. For instance, the combination of events in a system may give rise to bottlenecks in other locations. This can be done by integrating process mining and correlation analysis to identify events that are causally related one another (Toosinezhad et al., 2020).

The addition of attributes to the log makes it possible to mine additional perspectives. For instance, the yield of a particular process (e.g., semiconductor manufacturing) can be recorded in the log and used to derive high-yield paths in the production flow (Cho et al., 2021). (Dörgo et al., 2018) uses event logs from a coke refinery plant to retrieve the set of actions that operators perform frequently in similar situations. The goal is to exploit multi-temporal sequence mining to infer the causal relationship between alarms and supervisors' actions, together with the effects of these actions. Another attribute of interest is cost. Indeed, if each event record refers to an activity, the hourly cost of performing the same is likely to be known. As a result, the total cost of production can be estimated with the summation of the cost of the activities included in a process (Tu & Song, 2016).

The availability of data from several instances in a system also enables investigating the quality perspective of a production process. (Dogan & Gurcan, 2018) claim that the combination of process mining with traditional Statistical Process Control (SPC) techniques makes it possible to make effective decisions for quality problems. In general, the underlying research question is to find what combination of process steps distinguishes the parts in which a strategy (e.g., parameter setting) is successful, from the parts in which the goal is not reached. (Meyer et al., 2014) combined process mining with control theory and proposed an iterative approach to enhancing treatment strategies by predicting and preventing. During runtime, the deviations between the time sequence streams of parts in a model can be used to determine concept drifts, that is, deviations from the nominal process behaviour (Stertz et al., 2020). When a reference model is available, the quality assessment can be performed using conformance checking methods. (Paszkiewicz, 2013) identified the production management processes that can be assessed using a conformance checking procedure: logical conformance to the formal model, respect of the imposed production policy (e.g., first-in-first-out), quality assurance, performance indicators (e.g., maximum allowed system time), condition-based rules (e.g., re-work), and workload distribution. (Saraeian & Shirazi, 2020) developed a conformance checking module which compares real and expected characteristics of an additive manufacturing process with the goal of preventing cyber-attacks and network intrusions.

2.2.2 To-be Analysis (Information-based Model Generation)

(Jo et al., 2014) defined the scope of an intelligent tool that can view shop-floor data in real time, recognise, and resolve issues at production planning level. For these capabilities, four main modules are required: (1) data visualisation, to allow quick identification of issues in the system; (2) process mining, to quickly gather a model of the system; (3) methods for selecting and evaluating alternatives, and (4) a simulation environment to verify the proposed solutions before implementation.

Another process that can be analysed using process mining is worker’s movements. The supervision of manual operation is a very resource-demanding task. Hence, the combination of process mining and activity recognition can potentially increase efficiency and effectiveness (Mannhardt et al., 2018). The scope is to use data as input to discovery approaches that can reveal knowledge about operators. Process mining can also be used together with a big data analysis procedure to structure the knowledge hidden in more irregular data sources, such as the text of e-mails (Yang et al., 2014). The generation of a model enables prediction capabilities: given the state of execution of a process, knowing how the situation might evolve allows the supervisors to take the proper corrective actions (Ferilli & Angelastro, 2019).

2.3 Process Mining as a BPM decision support tool in Manufacturing

This paper focuses on discovering models related to business processes that support a manufacturing environment, hence directly determining its achievable performance. With this scope, (Flath & Stein, 2018) proposed a toolbox for developing predictive models exploiting machine learning algorithms and the manufacturing process mapped using process mining techniques. PM is used as a support tool to develop the model, to filter the non-relevant features and to extract process patterns. (Ortmeier et al., 2021) discussed how process mining could support life cycle assessment activities in manufacturing, for instance, to identify process deviations and interruptions. (Dakic et al., 2019) analysed the potential of two principal process mining software tools and developed a methodology for implementing process mining techniques in a case study. The goal was to gather a process map in the form of a directly-follow plotted graph.

The successful implementation of PM techniques in several fields feed the intuition that PM can support the management of production related processes. Further, most PM applications typically rely solely on one PM algorithm. As a result, the user is forced to compromise between specific algorithm performance measurements. For instance, inductive algorithms show good results in terms of scalability, fitness to the log, and precision, while heuristic algorithms can discover more process patterns. To cover this shortcoming, this paper introduces the possibility of using more than one PM algorithm within the same analysis. Specifically, it introduces a hybrid methodology that combines inductive and heuristic mining algorithms, making it possible to exploit the advantages of each mining algorithm while keeping computation times reasonably low. In addition, as far as the authors know, the existing contributions on PM application to P2P process relate to non-manufacturing companies, and there are no analyses on the benefits of using such approach to a production support process.

3 Application Problem

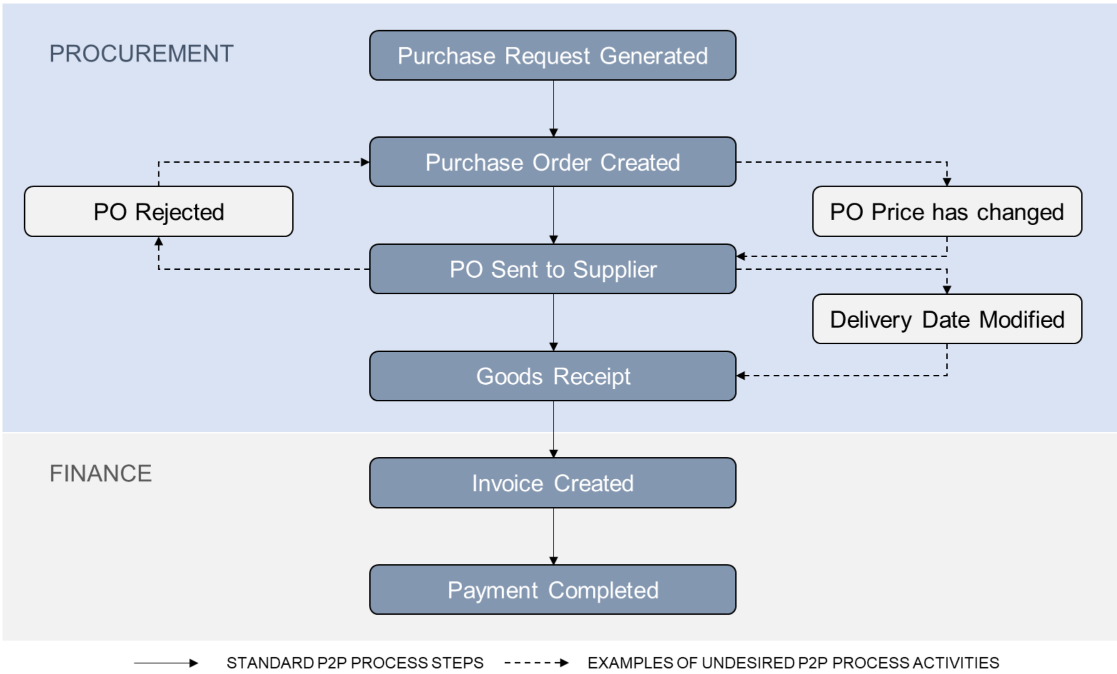

Figure 2. Example of a P2P process.

Our research interest is directed at developing a PM-based approach to discovering and analysing a production support process. As an application problem, a manufacturing company's procure-to-pay (P2P) process was selected. This section describes the process and introduces the problem statement.

3.1 The Procure-to-pay Process

The P2P process can be summarised in the steps depicted in Figure 2. P2P starts when a company employee or department needs to purchase a given product or service. In such a case, a purchase request is generated. The request may also represent an order generated automatically by the MRP system, especially if the purchased items are consumed regularly. The purchase request is then translated into a purchase order (PO), which is then transmitted to the supplier. If the supplier approves it, then the goods are shipped and delivered to the customer. The process ends with the generation of the invoice document and with its payment. Additional undesirable activities may also occur. If the purchase request is generated with the wrong attributes, some order attributes may need to be changed at PO creation before sending it to the supplier. For instance, the “PO price changed” activity in Figure 2, refers to the fact that the purchase order price has been modified before sending the PO to the supplier. Similarly, it can occur that the order is sent to the supplier with wrong parameters and reworking activities are needed to be performed before receiving the goods (i.e., “PO rejected” and “Delivery Date Modified” activities in Figure 2). The performance of the procurement process strongly impacts the company’s overall cycle time (Bag et al., 2020). Hence, the procurement process needs to be managed very closely since its low efficiency and effectiveness can have a detrimental effect on the cost, time and quality production process Key Performance Indicators (KPIs) (Bag et al., 2020). Performance indicators such as the supplier’s deliver reliability and capabilities, the throughput times between different activities, the detection of unnecessary activities and reworking ratios, need to be measured very carefully. Inefficient activities related to procurement or unreliable suppliers can lead to delays in receiving the goods with costly consequences in the overall production schedule. Instead, efficient supply processes mean the right quantity of goods delivered at the right time to guarantee coordination of the production activities.

3.2 Problem Statement

Monitoring and discovering the flow of a P2P process can be challenging, due to the potentially high number of rework activities. Figure 2 shows some practical examples of inefficient procurement-related activities that may occur. The “PO Price Changed” activity usually slows down the process and negatively affects the process cost since additional manual effort is involved. Similarly, the “Delivery Date Modified” activity can be considered a rework inefficiency since the change of the delivery date can negatively affect the production schedule. Further, its occurrence frequency can be used as a supplier reliability indicator. “PO rejected” indicates that a supplier has not accepted the PO. In this case, time and effort have been wasted for the creation of the rejected PO.

Given the importance of the P2P support process, it is fundamental to be able to detect the undesired process variants that negatively affect the company's productivity, and to identify repetitive activities that could be avoided or automated. The discovery of process deviations can help in detecting supplier-related inefficiencies, to ensure order fulfilment, and to optimise the company's working capital.

Despite the aforementioned necessities, it is common that process models, developed using a traditional process analysis approach, are based on information that comes from subjective opinions or from old documents, rather than fact backed up by data. Starting from an untruthful or obsolete process model, the detection of sources of inefficiency, deviations from the normal execution of the process, and the evaluation of the real process KPIs may lead to misleading results. To avoid the above-mentioned inefficiencies, data-driven model generation approaches can be exploited to support the discovering and monitoring phase of a business process analysis.

4 Proposed Approach

This work proposes a systematic approach which aims to support companies in automatically mapping production support processes, such as the P2P process, starting from operational event data recorded in IT systems. It provides a systematic methodology that depends on the combination of two process mining algorithms: Heuristic Miner (Weijters & Ribeiro, 2011) and Inductive Miner (Leemans et al., 2015). This approach assumes that, during execution of the P2P process, process activities event data, that represents well-defined steps in the process, can be recorded by an IT system.

4.1 Overview

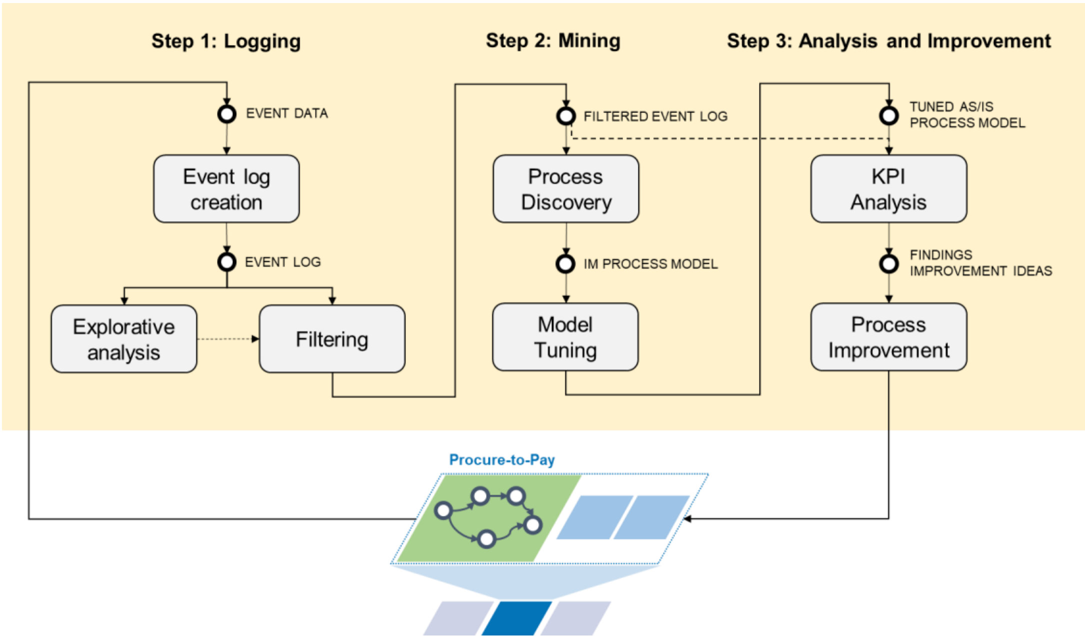

Figure 3. Proposed Approach in a nutshell.

The approach is illustrated in Figure 3. It is composed of three main steps:

- Step 1: *Logging* refers to the tasks that need to be performed to obtain an event log that could be used with process mining discovery algorithms.
- Step 2: *Mining* refers to the process discovery and tuning activities which, starting from the event log produced in Step 1, make it possible to discover the AS-IS process model with high quality characteristics.
- Step 3: *Analysis and Improvement* then describes how to analyse the process and how to improve it by using the process model discovered in step 2.

4.2 Step 1: Logging

The starting point of this approach describes how to construct an event log that is suitable for the mining activities. An event log L is a collection of event-data and represents the set of all the possible sequences of activities observed during the different process executions. It is composed of different cases that represent unique process executions. Each case is composed of a trace or sequence of *events* that happened at a certain time performed by an entity that could be physical or abstract (e.g., software system) (Van Der Aalst, 2016). Table 1 shows an example of an event log.

Table 1. Extract from a generic event log.

CASE ID	**TIMESTAMP**	**ACTIVITY**	**EMPLOYEE**	**ATTRIBUTE 1**	**ATTRIBUTE 2**	**ATTRIBUTE 3**
				Supplier	*Order Amount*	*Item Description*
1	6/3/19 7:41 6/3/19 7:41 8/3/19 15:21 13/3/19 10:45	Register request Examine thoroughly Decide Pay request	John	B&G Brothers	4290.00 €	A513 TUBULAR STEEL
2	6/3/19 7:42 8/3/19 15:21 13/3/19 10:45	Register request Decide Pay request	John	B&G Brothers	5070.00 €	BEARING 15MM
3	6/3/19 7:46 8/3/19 15:21 13/3/19 10:45	Register request Decide Pay request	John	B&G Brothers	1410.00 €	BEARING 25MM
4	30/4/19 17:04 30/4/19 17:28 30/4/19 17:28 3/5/19 10:51 3/5/19 10:52 16/5/19 16:01	Register request Approve Examine thoroughly Decide Approve Pay request	Rose John	Jackson 1960	440.50 €	PUMP XGTR

Information systems such as ERP, which record process execution transactions, do not usually provide structured datasets with the characteristics of an event log. Recorded process event data is usually stored over several tables and aggregating it in the shape of a suitable event log requires additional effort. Logging consists of three main sub-steps: (1) event log creation, (2) explorative

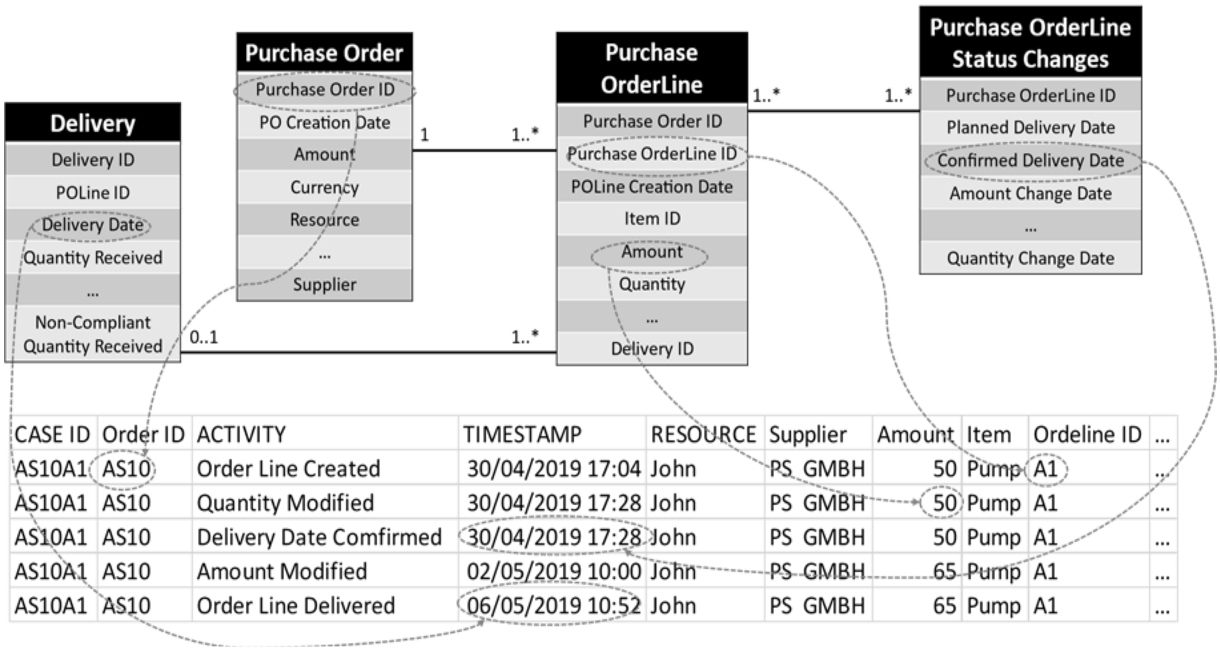

Figure 4. Example of a class diagram, showing the relations between the procurement process tables, and the relation with a case of the extracted event log

analysis, and (3) filtering.

Event log creation

To create an event log, the first activity is the selection of which sub-set of the data collected by the IT systems must be included. A business process is composed of different process executions, also called process instances and to obtain useful and interesting results about the analysed process, it is important to select the entity that better identifies the different process executions and relate the process recorded event data to them. For instance, in the P2P process, orders are usually composed of different lines which represent a purchased good or service as also shown in the example in Figure 4. From a practical perspective, to have a detailed picture of the analysed process, this work suggests considering the purchased order line as the case identifier, since even items purchased within the same order may have different process executions.

Once the process identifier is selected, it is fundamental to figure out where event data related to it is scattered within the different companies' databases. Figure 4 shows an example of a class diagram of P2P data tables and how they are related within an event log case. Each table collects data regarding activities performed during the procurement process. The order table collects data regarding activities performed at the order level, every order has a unique identifier, and it is usually made up of different items. The order line table collects event-data about individual items that make up the order, while each line is characterised by an identifier (ID) and holds information about the ordered product. The order line table is linked with the table regarding the changes made to each of the order line attributes, and with the tables that refers to delivery related event-data. Having understood the different relationships between the tables, collecting the event data means that the event log can be extracted automatically by creating a query, where each row represents a unique recorded event associated with a particular case, and the column represents the attribute associated with such event. In the example shown in figure 4, the case identifier was created ad hoc as the combination of the Order ID and the Order Line ID. After extraction, it may occur that the extracted event data holds missing or corrupted values, thus post-processing activities should be performed to repair the event log (Suriadi et al., 2017).

Explorative Analysis

Before starting the Mining activities, an exploratory analysis of the process event log can be performed. This activity is not mandatory but can be very useful to detect important insights that could be considered in the next approach to the activities. From a basic statistic perspective, it is possible to have an overview of all the individual cases and events. A schematic as-is status of the analysed process can be provided by looking at the mean length of a case (i.e., mean number of events), the most frequent starting and ending activities, the occurrence frequency per trace of each activity and the number of orders in progress, deleted and cancelled. Concurrently with basic statistics, the event log can be analysed from a visual perspective, for example representing it by means of the so-called dotted chart that presents a general overview of the process (Song & Aalst, 2007).

Filtering of not-completed cases

The next activity of this step is to filter out not completed cases from the extracted event log. When dealing with not completed traces, discovery techniques have difficulties with understanding the starting and the ending process activities. Indeed, referring to the P2P process, cases regarding orders in progress, deleted or cancelled should be filtered out before starting Step 2 to improve the algorithms' efficiency.

4.3 Step 2: Mining

In this step, we describe how to automatically discover an as-is process model, starting from a process execution event log. For this purpose, the following approach proposes the combination of two process mining discovery algorithms: Heuristic Miner (Weijters & Ribeiro, 2011) and Inductive Miner (Leemans et al., 2015). This step is composed of two mains phases, as illustrated in Figure 3: (1) process discovery and (2) process model tuning.

Process discovery

The first activity in the Mining step, concerns the discovery of a *process model, M,* which "collects graphically inter-related events, activities and decision points that involve a number of actors and objects" (Dumas et al., 2018). Process discovery consists of mapping an event log $L \in B(\mathcal{A})$ onto a process model *M*, which can describe the behaviour recorded in *L,* through a function $\gamma(L) = \mathrm{M}$. For this phase, the inductive miner (IM) which provides a better trade-off between scalability, fitness, generalisation, and precision (Leemans et al., 2015) was chosen as the process discovery algorithm. The inductive algorithm logic is illustrated briefly in the appendix, and the reader may refer to (Leemans et al., 2015) for theoretical details.

By applying the chosen discovery algorithm to the filtered event log generated in the previous step, it is possible to automatically discover a process model with Petri net notation. The algorithm discovers the model by deriving it from a process tree (Leemans et al., 2015) that by definition is always *sound* (i.e. a process model that is fully replicable). Indeed, the discovered model is sound too and thus it does not present any potential deadlock situations.(Van Der Aalst, 2016). By using the inductive algorithm, it is possible to filter infrequent behaviour from the most frequent ones. Indeed, by setting different frequency thresholds, different process models can be discovered starting from the same event log. Considering only the most frequent behaviours and filtering the less frequent ones is suggested. The value of specific thresholds strictly depends on the application, although from empirical experiments a threshold of 80% of paths has provided satisfactory results. Hereinafter, for the sake of simplicity, we refer to the model discovered by the inductive mining algorithm as the IM process model.

Process model tuning

The quality of the IM process model can be characterized by some metrics. Considering all the process executions recorded in the event log and all the process behaviours allowed by a process model as two different traces sets, $\tau(L)$ and $\tau(M)$, the qualities dimensions of a process model can be measured by analysing the commonalities and differences between the two sets. Typically, *fitness* (φ) and *precision* (π) are two metrics used to check the quality of a process model. *Fitness* is defined as "the fraction of behaviour of the log that is also allowed by the model" (Carmona et al., 2018):

$$\varphi = \frac{|\tau(L) \cap \tau(M)|}{|\tau(L)|} \qquad (1)$$

$\varphi = 1$ if all the event log traces from beginning to end can be replayed by the model. *Precision* is defined as "the counterpart of fitness and can be calculated by looking at the fraction of the model behaviour that is covered in the log" (Carmona et al., 2018):

$$\pi = \frac{|\tau(L) \cap \tau(M)|}{|\tau(M)|} \qquad (2)$$

$\pi = 1$ means that all the behaviours allowed by the model have been recorded by the event log. When dealing with a complex event log which is composed of numerous process variants, the inductive algorithm usually discovers a process model characterised by several parallel composition operators that could allow for many behaviours not observed in the log. Consequently, the IM model generated from a complex event log is usually characterised by a high value of fitness, while its precision value is rarely as high. This situation is typically called underfitting and to improve its precision, the approach described in the Tuning sub-step is proposed and is described briefly in Figure 5. The tuning activities take the IM process model and the filtered event log extracted in step 1 as input. Thanks to its soundness property, the IM process model can be used as a skeleton of the tuned process model while the heuristic algorithm (Weijters & Ribeiro, 2011) is used to improve the model's accuracy, focused on precision.

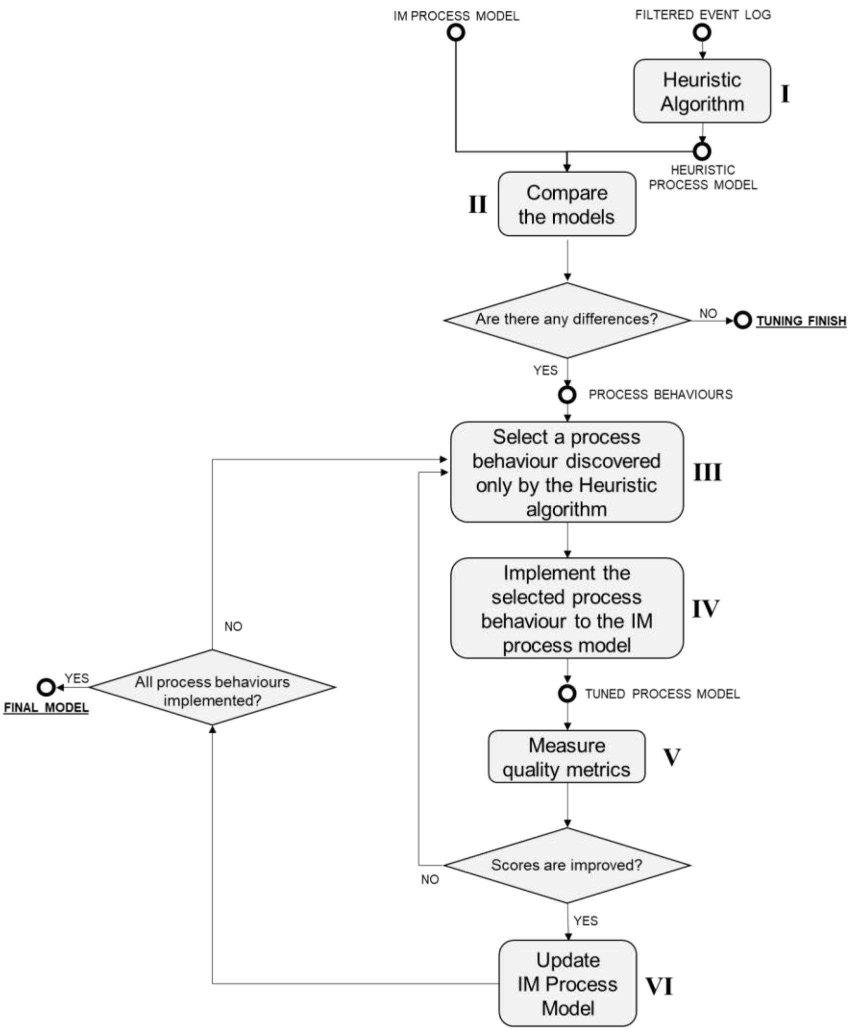

Figure 5. Process model tuning.

Starting from the filtered event log, the first activity in the tuning phase is to discover a new process model using the heuristic algorithm (Weijters & Ribeiro, 2011) (I). This algorithm is usually able to discover more complex process behaviours than the inductive algorithm since frequencies of events and sequencies are considered with more importance. However, for

complex event logs composed of numerous traces and process variants, this algorithm has the limitation that discovering a sound process model is not assured. Hence, for complex event logs this algorithm could discover not fully replicable models (Van Der Aalst, 2016).

Once a model is discovered using the heuristic algorithm, it must be compared with the IM process model (II). If the two models are different, it is possible to select a process behaviour such as an AND/OR split or joints or a path, previously neglected by the inductive miner (III) and add it to the IM process model (IV). After this activity, the quality metrics (i.e., fitness and precision) of the new process model need to be computed and compared with the previous ones (V). If the process model quality improves, it is possible to update the IM process model with the selected process behaviours (VI) and repeat the iterations, otherwise the split or joint is discharged. The iteration ends when it is not possible to improve the process model quality with the addition of the heuristic miner discoveries. The result is a tuned process model that can describe what was observed in the event log with a high degree of accuracy.

4.4 Step 3: Analysis and Improvement

Starting from the tuned process model, the process flow can be monitored to check if it is performed as desired or not. Indicators such as the sojourn time between activities can be immediately computed to analyse the throughput time.

As qualitatively described in Figure 3, by aligning the filtered event log traces with the discovered process model, it is possible not only to compute the process quality metrics and to detect whether there are discrepancies between the process model and the event log, but some process performance indicators can be immediately measured from both control flow and time perspectives. The results of the alignments can provide indicators on how many traces are replayed through certain paths of the process model, to immediately detect the most and less frequent paths of the process.

The alignments results and the discovered process model could be used as the starting point for developing improvement actions, that can fix anomalous behaviours detected with PM if present or that can enhance the analysed process performance levels. Thus, the findings discovered from the process analysis should be used as a reference for targeted corrective or improvement actions, that are driven from objective data and not from subjective opinions typical of traditional process analysis approaches. After the improvement of the process, the proposed approach can be applied again to check whether the process improvement actions have been correctly implemented.

5 Case Study

The proposed approach was applied to a real case study to map and to monitor the production support process used by a multinational manufacturer of steam management systems and peristaltic pumps. The company has bases in more than 60 countries all over the world and the manufacturing plant of the Italian division oversees the production of valves and heat exchangers. The process analysed is the purchasing process for standard items, which is a sub-process of the P2P process. In 2018 alone, more than 8,000 purchase orders were generated and recorded by the ERP system, with a total sum of almost € 25 M in products and services ordered by the procurement department.

The primary objective was to discover the purchasing process for standard items. For this case study, the open source ProM toolkit (Van Der Aalst et al., 2009) was used as the process mining software, and Matlab as a data processing tool. For the sake of simplicity activities are labelled with letters, and the activity description can be derived from Table 2.

5.1 Step1: Logging

Table 1. Single-letter label activity codification and basic statistics of 2019 purchase order lines.

Activity	Single-letter label	Traces in which occurs	Relative frequency
Creation of an order	a	14 568	100,0%
Change of purchase price	b	4 705	32,3%
Change of purchase quantity	c	1 886	12,9%
Change of planned receipt date	d	11 221	77,0%
Planned receipt date confirmation	e	12 809	87,9%

Change of confirmed planned receipt date	f	1 689	11,6%
Order partially received	g	700	4,8%
Order totally received	h	14 568	100,0%

To create the process event-log, the purchase order (PO) line was selected as the case identifier and the event data was pre-processed and aggregated into an event log. Following the logic behind the processing of PO lines by the purchasing department, it was possible to understand the ERP architecture and where to find data related to the selected process instance. Data was scattered over 4 different tables such as those shown in figure 4. All the event-data related to PO lines of standard items issued from January 2017 until November 2019 was extracted from the company ERP. Since the extracted data was at a too detailed level of granularity, a pre-processing activity was needed. Repetitive event data was filtered out, and some events were aggregated to obtain an event log with a lower level of granularity. For the event log creation sub-step, Matlab was used as a supporting tool.

The extracted event log held information about the 8 different activities shown in Table 2, and it was composed of 68,903 cases for a total of 291,176 events.

For the first part of the analysis performed, only event-data concerning the year 2019 was used since the main aim was to discover an updated process model. Data related to the years 2017 and 2018 was not considered in the mapping phase, but it was useful for checking whether the process drifted through the years during the analysis and improvement step.

Once the event log had been created, an explorative analysis was performed to detect general process insights from visual and basic statistic perspectives. Through the computation of the relative

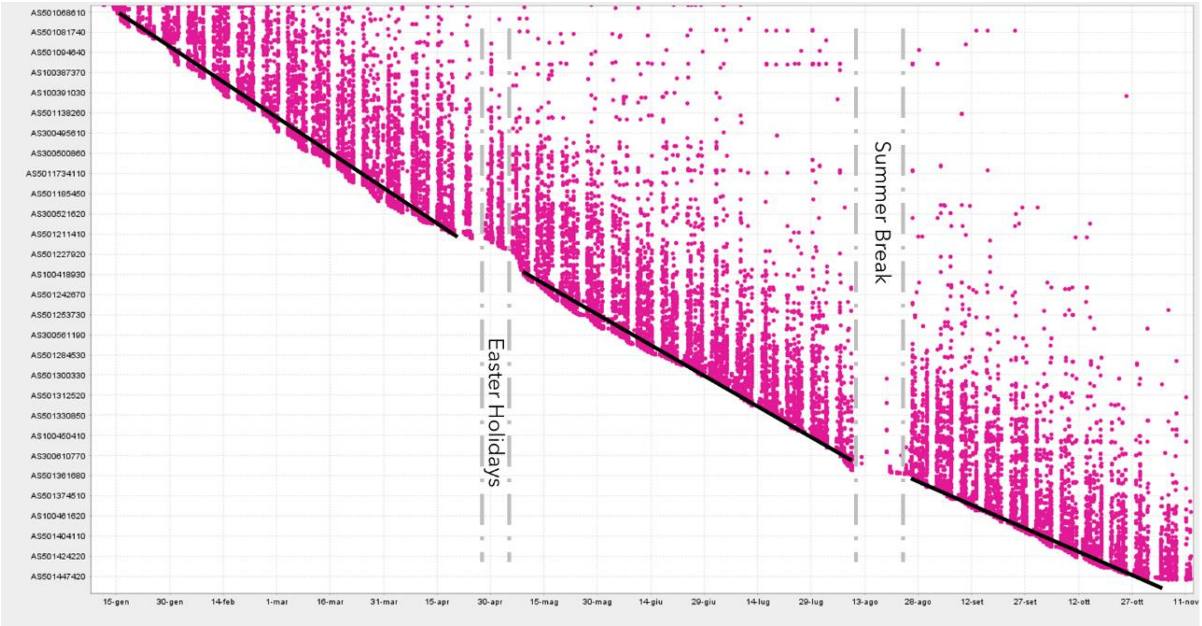

Figure 6. Dotted chart of 2019 complete standard items purchase orders.

frequencies of the different process activities, which are summarized in Table 2, it was possible to immediately discover some anomalous findings. For example, reworking activities such as activity *b* and *d* occurred with a not negligible frequency and activity *e*, which from a normative perspective is mandatory, did not occur in all the cases. **For a visual analysis, the dotted chart was plotted using the Prom plugin called "Project log on Dotted Chart", and the result is illustrated in Figure 6.**

Before the discovery activities, not completed cases were filtered out from the extracted event log. Deleted or cancelled purchase orders accounted for almost 10% of those extracted; this results in a waste of time and money by the procurement department.

5.2 Step 2: Mining

The filtered event log was used as input for the inductive discovery algorithm which was implemented using the "Mine with Inductive visual Miner" Prom plug-in. The output was the as-is purchasing process model depicted in Figure 8 which describes the purchasing process as follows. A purchase order line is created in the ERP system so activity *a* is recorded, then it can be subjected to different activities that can be performed concurrently or in a predefined sequence. Activities *c* and *b* can be performed in parallel and without any predefined order within activities *d, e,* and *f* that, by contrast, must be performed in a predefined sequence. Activities *b, c* and *d* represent activities performed to change some of the purchase order line attributes, namely the order price, the order quantity and the planned delivery date computed by the MRP respectively. Activities *e* and *f* refer to the confirmation and the change of the delivery date given by suppliers respectively. The process ends when the order is delivered. If the order is delivered partially activity *g* is recorded, that means some purchase order compliances are detected. Otherwise, the mapped process can be considered fully completed when the purchase order line is delivered with no issues, thus activity *h* is recorded. The discovered Petri net is characterised by several parallel activities and its quality indicators, fitness, and precision, computed using the "Multi perspective process explorer" Prom plug-in, were 0.9 and 0.8 respectively. This means that 90% of the recorded events can be correctly replayed by the process model. By contrast, due to its relatively low precision the model allows for some process behaviour items (i.e., 20 %) not recorded in the event log.

The IM process model suggests that the order price and the quantity change (activities *c* and *b*) were often performed after confirmation of the delivery date (activity *e*). At the same time activity

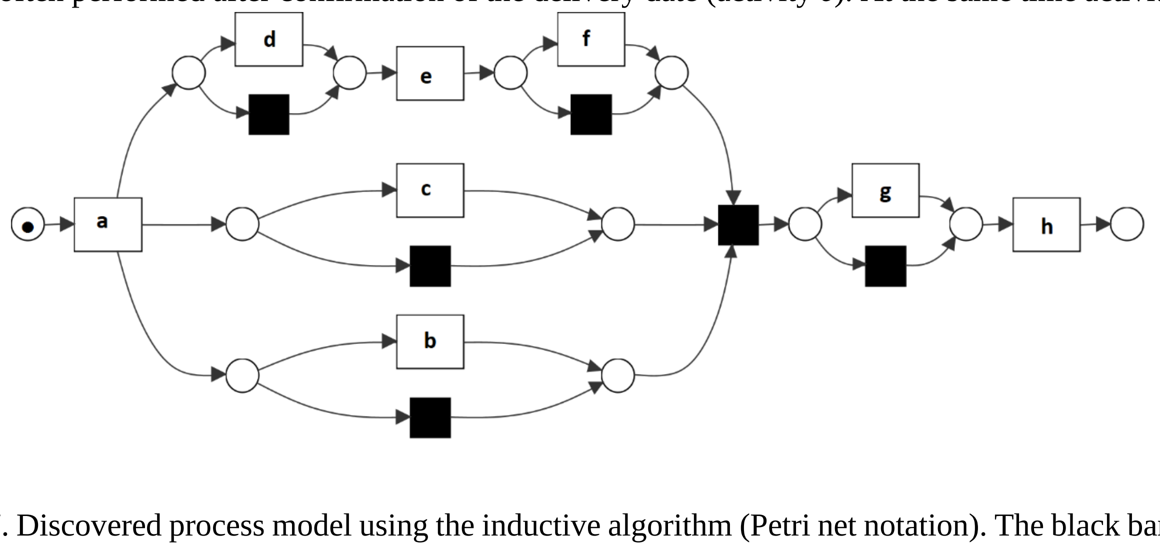

Figure 7. Discovered process model using the inductive algorithm (Petri net notation). The black bars represent silent transitions which make it possible to either skip process activities or to enable other transitions.

e, as shown in Table 2, is not present in all the traces and by contrast in the model in figure 7 it cannot be skipped. All these anomalies suggested that the real process is underfitted by the IM process, and through the process model tuning activities it was possible to improve the process model's accuracy.

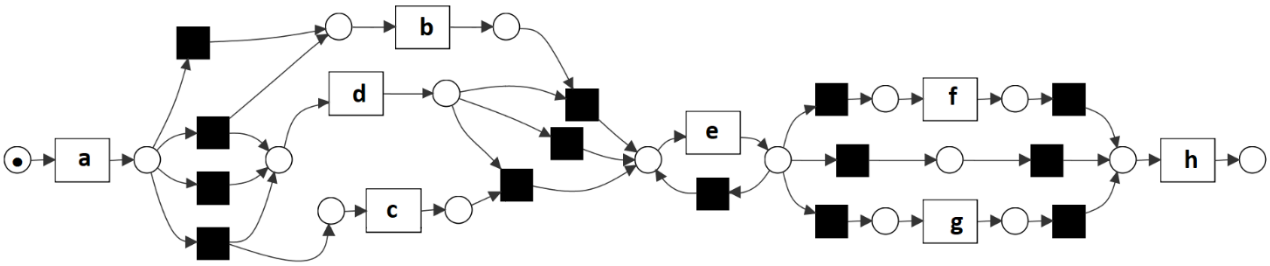

Figure 8. Discovered process model using the heuristic algorithm, Petri net notation.

Thanks to the process model tuning activities, using the heuristic algorithm implemented using the "interactive data-aware heuristic miner" Prom plug-in made it possible to detect some process relationships that were neglected by the inductive miner. The model discovered using the heuristic algorithm is shown in Figure 8 while the directly followed measures and dependency relations used by the heuristic algorithm are presented in Tables 3 and 4, respectively. The Heuristic model was found to be unsound because the proper completion property[1] was not satisfied. Compared to the IM process model, the heuristic algorithm mapped activity b and c according to the following relations $e \not> b$; $e \not> c$ and $b\#c$, which means that activity *e* was usually not followed by neither *b* nor *c*, and activities *b* and *c* were not present in the same trace. Another difference that was discovered is a short unit redo loop for activity *e*, that means activity *e* can occur more than once per trace. The last important difference was that the heuristic algorithm did not detect any relevant direct relation between activities *f* and *g*, so *f* and *g* are mapped as they cannot occur in the same case.

[1] **Proper completion property** (Petri nets): for any marking, if the sink place is marked, all other places must be empty.

Table 2. Frequency of the directly following relationship in the case study event log. For instance, activity *b* followed activity *a* in 2848 events.

$>_L$	a	b	c	d	e	f	g	h
a	0	2 848	1 266	8 596	1 360	0	30	468
b	0	77	229	1 850	1 990	53	24	641
c	0	219	34	1 008	453	13	6	260
d	0	1 508	373	100	9 092	0	12	269
e	0	250	105	0	1 923	1 629	533	10684
f	0	8	46	0	0	65	95	1 546
g	0	0	0	0	0	0	7	700
h	0	0	0	0	0	0	0	0

Following the tuning step, the discovered splits and joints were added iteratively to the IM process model. We discovered that by mapping activities *b*, *c*, and *d* in parallel but before activity *e* and *f* and by adding a short unit loop for activity *e*, the IM process model qualities were positively affected. In addition, a path that allows for skipping activities *e* and *f* was added to the IM process model. This process relationship was suggested by looking at the directly following measures and dependency relationships.

Table 3. Dependency measures between the activities of the case study event log. The dependency measure is defined in Appendix A. If the measure between two activities is close to 1, it means there is a strong positive dependency between *a* and *b*, i.e., *a* is often the cause of *b*.

$\Rightarrow_L$	a	b	c	d	e	f	g	h
a	0,00	1,00	1,00	1,00	1,00	0,00	0,97	1,00
b	-1,00	0,99	0,02	0,10	0,78	0,73	0,96	1,00
c	-1,00	-0,02	0,97	0,46	0,72	-0,55	0,86	1,00
d	-1,00	-0,10	-0,46	0,99	1,00	0,00	0,92	1,00
e	-0,97	-0,78	-0,72	-1,00	1,00	1,00	1,00	1,00
f	-1,00	-0,73	0,31	0,00	-1,00	0,98	0,99	1,00
g	-0,99	-0,96	-1,00	-0,92	-1,00	-0,99	0,88	1,00
h	-0,98	-1,00	-1,00	-1,00	-0,99	-1,00	-1,00	0,00

The final as-is process model is shown in Figure 9 and is characterized by a fitness φ = 97% and a precision π = 94%, which highlight the fact that, compared to the original IM process model, the tuned model fits the process behaviours observed in the event log better.

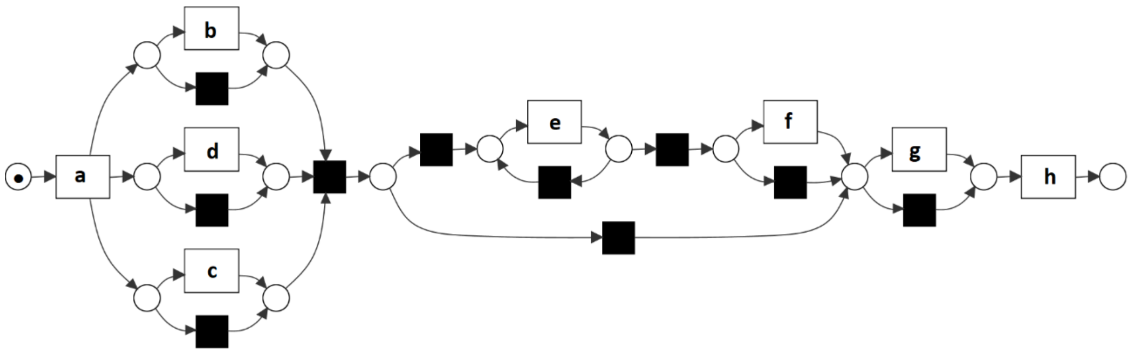

Figure 9. Tuned as-is Process Model

5.3 Step 3: Analysis and Improvement

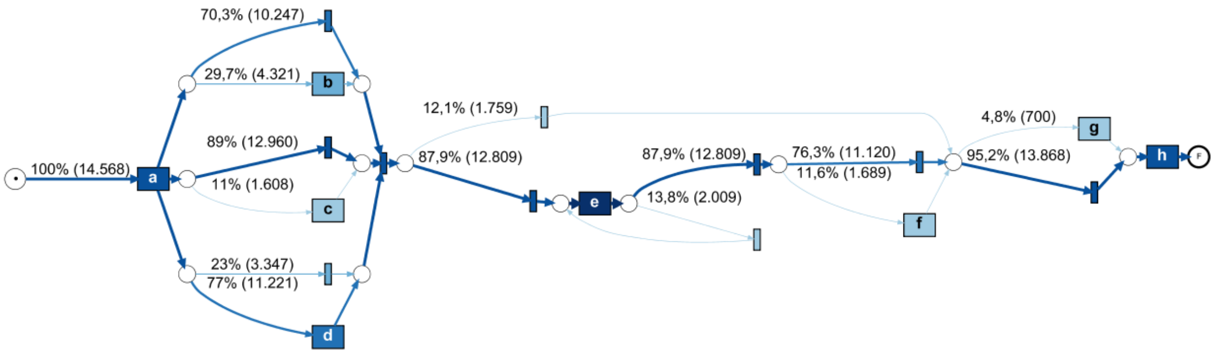

Figure 10. Alignments of 2019 Event log with the tuned process model.

For the sake of completeness, the event log traces were aligned with the tuned process model and the process was analysed from a control flow perspective. For this purpose, the "Multi perspective process explorer" Prom plug-in (Mannhardt et al., 2015) was used. The model in Figure 10 represents the tuned process model enriched with additional insights dispensed by the alignment.

Each arc of the net has values that represent how many traces were replayed through that path, in terms of absolute and relative frequency. The thickness of the arc is proportional to the path frequency, which is useful for immediately detecting the most frequent path given by the sequence (*a, d, e, h*). The most interesting discoveries from the analysis activity are summarised below.

By aligning the event log cases regarding 2017 and 2018 purchase orders and comparing the results with the 2019 case, it was possible to detect that the relative occurrence frequency of *d* (change of planned receipt date) increased over the years. Activity *d* is a reworking activity hence, it should not have been performed with high frequency. By discussing this with the purchasing manager, it was possible to understand that this issue was due to two main reasons. First, the delivery lead times uploaded in the ERP were not completely correct for some items. Second, in 2019 it was not possible to transform the planned purchase orders generated by the MRP into POs at the right time, due to a lack of resources. Thus, purchase orders were created with delays.

Activity *b* (change of purchase price) was experienced with a relatively high occurrence frequency which was discovered to be due to not updated price lists for some suppliers, indeed in many observed cases a change of the PO lines’ price was needed especially for certain suppliers.

Activity *e* (planned receipt date confirmation) was skipped in more than 10% of cases, even though from an internal normative perspective it was mandatory because it refers to the time instance when the supplier confirms the delivery date of the PO. When activity *e* is skipped it is hard for the planning department to schedule production since there is no official confirmation about when the purchased items will be available at the warehouse. This means that delays in the production phase could occur or could not be managed effectively. Through a process variant analysis, it was discovered that skipping of activity *e* was due to two main reasons: 1) a lack of collaboration by some suppliers, that means some suppliers rarely confirm the delivery date; 2) a non- efficient or effective expediting phase performed by some buyers.

Table 5. Discovered purchasing process undesired behaviours and proposed related corrective actions.

Undesired Behaviour	Reason	Corrective Action
Occurrence frequency of *b* was too high.	Not all the price lists were updated, so MRP was working with wrong parameters.	Price lists should be constantly updated in the system, and attention should be focused on suppliers with missing updated price lists.
Occurrence frequency of *d* was too high.	Delivery lead time was not fully reliable, and many planned purchase orders were not fulfilled at the right time.	Estimation of the delivery lead time should be updated, and an additional person should be added in the procurement department.
Activity *e* was skipped too many times.	Some buyers did not effectively perform the expediting phase. At the same time, some suppliers were not collaborative.	Expediting phase of some buyers should be encouraged.
Activity *e* was performed more than once per trace.	It was discovered that this behaviour is related only to some buyers. This affects the reliability index of some suppliers.	A discussion with those buyers should be held. The supplier reliability index should be reviewed.

Another anomalous behaviour refers to the repetition of activity *e* that was recorded for some cases. From a normative perspective, when the supplier changes the delivery date confirmed previously, instead of performing activity *e* again, activity *f* should be performed. The repetition of activity *e* had a negative impact on the supplier reliability index. The analysed company evaluated suppliers not only in terms of their OTTR (On Time To Request) index, but also on how many times the supplier modifies the purchase order delivery date, i.e., how many times activity *f* is performed over the delivered orders. Indeed, if activity *e* is repeated instead of performing activity *f*, the supplier reliability index is not reliable anymore. Process mining assisted in discovering that this anomalous behaviour was related only to some buyers, since the occurrence frequency of activity *e* repetition changes significantly between the different buyers.

In conclusion, the analysis was presented to the purchase managers and some improvement ideas, that are summarised in table 3, were suggested. The ideas were developed thanks to the evidence-based insights discovered through process mining.

6 Conclusion

This study shows how process mining techniques can be successfully applied to a real production support process. Starting from raw event data collected by the ERP system, an approach based on inductive and heuristic algorithms is proposed to better analyse a P2P process. The approach showed how discovery and conformance checking techniques can be applied to analyse the real behaviour of a P2P process, with the aim of supporting managers' decisions. Following the proposed approach, the as-is process model of a real purchasing process was discovered automatically, and interesting insights were detected. The discovered insights led to improvement ideas that could improve production performance. The case study presented showed how starting from raw event data recorded by the company ERP it was possible to develop improvement ideas to enhance the efficiency and effectiveness of the purchasing process. Anomalous resource behaviours were detected, such as being possible to correctly evaluate some suppliers using objective data. The proposed techniques proved to be very effective from a time and objectivity perspective and they can be applied to any kind of business process thanks to their high scalability. This approach could be successfully implemented to map, monitor, and analyse other production support processes as illustrated in Figure 1, both for those that provide input to manufacturing systems and the ones that receive input from the production environment. Thus, for future contributions this approach can be applied to other production support processes. Almost all industrial companies use ERP to support their internal business process. Introducing process mining as a new technology into a company to enhance their production process is something that can be done with limited investment. The behaviour of different employees can be monitored in a detailed way, to check whether they perform their respective tasks as desired or not. Even with all the previous considerations, process mining alone cannot fully substitute traditional process discovery and monitoring approaches. Rather, it can provide process insights and discover process deviations that can be the starting point for deeper and targeted research questions when conducting workshops or interviews.

Despite the advances and potentialities, the proposed approach is still affected by some limitations. Some of the approach's activities are still characterised by human involvement. The authors propose that future work looks at automating the comparison of the heuristic and inductive algorithms results, to speed up the discovery of the discrepancies and the analogies of the two models.

Another limitation is that the P2P process has been the only production support process analysed. In the future, this approach should be tested on different kinds of processes, such as product design and development. In addition, in this work as a case study only one source of data (i.e., ERP) was used and in the future, it will be relevant to develop a framework to guide the user on how to combine event data collected by sources other than the ERP, such as common IT systems and PLM or CRM or novel digital tools, as task management tools.

Bibliography

Bag, S., Dhamija, P., Gupta, S., & Sivarajah, U. (2020). Examining the role of procurement 4.0 towards remanufacturing operations and circular economy. *Production Planning and Control*, *0*(0), 1–16. https://doi.org/10.1080/09537287.2020.1817602

Belhadi, A., Zkik, K., Cherrafi, A., Yusof, S. M., & El fezazi, S. (2019). Understanding Big Data Analytics for Manufacturing Processes: Insights from Literature Review and Multiple Case Studies. *Computers & Industrial Engineering*, *137*, 106099. https://doi.org/https://doi.org/10.1016/j.cie.2019.106099

Buijs, J. C. A. M., La Rosa, M., Reijers, H. A., Dongen, B. F., & Aalst, W. (2013). Improving Business Process Models Using Observed Behavior. *Lecture Notes in Business Information Processing*, *162*, 44–59. https://doi.org/10.1007/978-3-642-40919-6_3

Carmona, J., van Dongen, B., Solti, A., & Weidlich, M. (2018). *Conformance Checking*. Springer.

Cho, M., Park, G., Song, M., Lee, J., Lee, B., & Kum, E. (2021). Discovery of Resource-Oriented Transition Systems for Yield Enhancement in Semiconductor Manufacturing. *IEEE Transactions on Semiconductor Manufacturing*, *34*(1), 17–24. https://doi.org/10.1109/TSM.2020.3045686

Choueiri, A. C., Sato, D. M. V., Scalabrin, E. E., & Santos, E. A. P. (2020). An extended model for remaining time prediction in manufacturing systems using process mining. *Journal of*

Manufacturing Systems, *56*, 188–201. https://doi.org/10.1016/j.jmsy.2020.06.003

Cooper, W. W., Seiford, L. M., & Tone, K. (2000). Data envelopment analysis. *Handbook on Data Envelopment Analysis*, 1–40.

Dakic, D., Sladojevic, S., Lolic, T., & Stefanovic, D. (2019). Process mining possibilities and challenges: A case study. *SISY 2019 - IEEE 17th International Symposium on Intelligent Systems and Informatics, Proceedings*, 161–166. https://doi.org/10.1109/SISY47553.2019.9111591

Dogan, O., & Gurcan, O. F. (2018). Data perspective of lean six sigma in industry 4.0 era: A guide to improve quality. *Proceedings of the International Conference on Industrial Engineering and Operations Management*, *2018*(JUL), 943–953.

Dörgo, G., Varga, K., Haragovics, M., Szabó, T., & Abonyi, J. (2018). Towards operator 4.0, increasing production efficiency and reducing operator workload by process mining of alarm data. *Chemical Engineering Transactions*, *70*, 829–834. https://doi.org/10.3303/CET1870139

Dumas, M., La Rosa, M., Mendling, J., & Reijers, H. A. (2018). *Fundamentals of Business Process Management*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-56509-4

Erevelles, S., Fukawa, N., & Swayne, L. (2016). Big Data consumer analytics and the transformation of marketing. *Journal of Business Research*, *69*(2), 897–904. https://doi.org/10.1016/j.jbusres.2015.07.001

Farooqui, A., Bengtsson, K., Falkman, P., & Fabian, M. (2019). From factory floor to process models: A data gathering approach to generate, transform, and visualize manufacturing processes. *CIRP Journal of Manufacturing Science and Technology*, *24*, 6–16. https://doi.org/10.1016/j.cirpj.2018.12.002

Ferilli, S., & Angelastro, S. (2019). Activity prediction in process mining using the WoMan framework. *Journal of Intelligent Information Systems*, *53*(1), 93–112. https://doi.org/10.1007/s10844-019-00543-2

Ferreira, D. R., & Vasilyev, E. (2015). Using logical decision trees to discover the cause of process delays from event logs. *Computers in Industry*, *70*, 194–207. https://doi.org/10.1016/j.compind.2015.02.009

Flath, C. M., & Stein, N. (2018). Towards a data science toolbox for industrial analytics applications. *Computers in Industry*, *94*, 16–25. https://doi.org/10.1016/j.compind.2017.09.003

Fleig, C., Augenstein, D., & Maedche, A. (2018). Process mining for business process standardization in ERP implementation projects – An SAP S/4 HANA case study from manufacturing. *CEUR Workshop Proceedings*, *2196*, 149–155.

Harding, J. A., Shahbaz, M., Srinivas, & Kusiak, A. (2005). Data Mining in Manufacturing: A Review. *Journal of Manufacturing Science and Engineering*, *128*(4), 969–976. https://doi.org/10.1115/1.2194554

Intayoad, W., & Becker, T. (2018). Exploring the Relationship between Business Processes and Contextual Information in Manufacturing and Logistics Based on Event Logs. *Procedia CIRP*, *72*, 557–562. https://doi.org/10.1016/j.procir.2018.03.220

Jans, M., Van Der Werf, J. M., Lybaert, N., & Vanhoof, K. (2011). A business process mining application for internal transaction fraud mitigation. *Expert Systems with Applications*, *38*(10), 13351–13359. https://doi.org/10.1016/j.eswa.2011.04.159

Jo, H., Noh, S. D., & Cho, Y. (2014). An agile operations management system for green factory.

International Journal of Precision Engineering and Manufacturing - Green Technology, *1*(2), 131–143. https://doi.org/10.1007/s40684-014-0018-z

Kozjek, D., Vrabič, R., Rihtaršič, B., Lavrač, N., & Butala, P. (2020). Advancing manufacturing systems with big-data analytics: A conceptual framework. *International Journal of Computer Integrated Manufacturing*, *33*(2), 169–188. https://doi.org/10.1080/0951192X.2020.1718765

Kurscheidt, R. J., Santos, E. A. P., de FR Loures, E., Pecora Jr, J. E., & Cestari, J. M. A. P. (2015). A Methodology for Discovering Bayesian Networks Based on Process Mining. *IIE Annual Conference. Proceedings*, 2322.

Lee, C. K. H., Ho, G. T. S., Choy, K. L., & Pang, G. K. H. (2014). A RFID-based recursive process mining system for quality assurance in the garment industry. *International Journal of Production Research*, *52*(14), 4216–4238. https://doi.org/10.1080/00207543.2013.869632

Leemans, S. J. J., Fahland, D., & van der Aalst, W. M. P. (2015). Scalable process discovery with guarantees. In *Enterprise, Business-Process and Information Systems Modeling* (pp. 85–101). Springer.

Lugaresi, G., & Matta, A. (2021). Automated manufacturing system discovery and digital twin generation. *Journal of Manufacturing Systems*, *59*(December 2020), 51–66. https://doi.org/10.1016/j.jmsy.2021.01.005

Mahendrawathi, E. R., Zayin, S. O., & Pamungkas, F. J. (2017). ERP Post Implementation Review with Process Mining: A Case of Procurement Process. *Procedia Computer Science*, *124*, 216–223. https://doi.org/10.1016/j.procs.2017.12.149

Mannhardt, F., Bovo, R., Oliveira, M. F., & Julier, S. (2018). A taxonomy for combining activity recognition and process discovery in industrial environments. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Vol. 11315 LNCS*. https://doi.org/10.1007/978-3-030-03496-2_10

Mannhardt, F., de Leoni, M., & Reijers, H. A. (2015). *The multi-perspective process explorer*.

Martin, N, Swennen, M., Depaire, B., Jans, M., Caris, A., & Vanhoof, K. (2017). Retrieving batch organisation of work insights from event logs. *Decision Support Systems*, *100*, 119–128. https://doi.org/10.1016/j.dss.2017.02.012

Martin, Niels, Depaire, B., & Caris, A. (2015). The use of process mining in business process simulation model construction structuring the field. *Business and Information Systems Engineering*, *58*(1), 73–87. https://doi.org/10.1007/s12599-015-0410-4

Meyer, G., Adomavicius, G., Johnson, P. E., Elidrisi, M., Rush, W. A., Sperl-Hillen, J. A. M., & O'Connor, P. J. (2014). A machine learning approach to improving dynamic decision making. *Information Systems Research*, *25*(2), 239–263. https://doi.org/10.1287/isre.2014.0513

Milde, M., & Reinhart, G. (2019). Automated Model Development and Parametrization of Material Flow Simulations. *Proceedings - Winter Simulation Conference*, *2019-Decem*, 2166–2177. https://doi.org/10.1109/WSC40007.2019.9004702

Ortmeier, C., Henningsen, N., Langer, A., Reiswich, A., Karl, A., & Herrmann, C. (2021). Framework for the integration of Process Mining into Life Cycle Assessment. *Procedia CIRP*, *98*, 163–168. https://doi.org/10.1016/j.procir.2021.01.024

Pane, S. F., Awangga, R. M., Siregar, M. A. H., & Majesty, D. (2021). Mapping log data activity using heuristic miner algorithm in manufacture and logistics company. *Telkomnika (Telecommunication Computing Electronics and Control)*, *19*(3), 781–791.

https://doi.org/10.12928/TELKOMNIKA.v19i3.18153

Park, J., Lee, D., & Zhu, J. (2014). An integrated approach for ship block manufacturing process performance evaluation: Case from a Korean shipbuilding company. *International Journal of Production Economics*, *156*, 214–222. https://doi.org/10.1016/j.ijpe.2014.06.012

Paszkiewicz, Z. (2013). Process mining techniques in conformance testing of inventory processes: An industrial application. In *Lecture Notes in Business Information Processing* (Vol. 160, pp. 302–313). https://doi.org/10.1007/978-3-642-41687-3_28

Pourbafrani, M., van Zelst, S. J., & van der Aalst, W. M. P. (2020). Supporting automatic system dynamics model generation for simulation in the context of process mining. *Lecture Notes in Business Information Processing*, *389 LNBIP*, 249–263. https://doi.org/10.1007/978-3-030-53337-3_19

Rozinat, A., Mans, R. S., Song, M., & van der Aalst, W. M. P. (2009). Discovering simulation models. *Information Systems*, *34*(3), 305–327. https://doi.org/10.1016/j.is.2008.09.002

Ruschel, E., Rocha Loures, E. D. F., & Santos, E. A. P. (2021). Performance analysis and time prediction in manufacturing systems. *Computers and Industrial Engineering*, *151*. https://doi.org/10.1016/j.cie.2020.106972

Saraeian, S., & Shirazi, B. (2020). Process mining-based anomaly detection of additive manufacturing process activities using a game theory modeling approach. *Computers and Industrial Engineering*, *146*. https://doi.org/10.1016/j.cie.2020.106584

Schuh, G., Gutzlaff, A., Cremer, S., Schmitz, S., & Ayati, A. (2020). A data model to apply process mining in end-to-end order processing processes of manufacturing companies. *IEEE International Conference on Industrial Engineering and Engineering Management*, *2020-Decem*, 151–155. https://doi.org/10.1109/IEEM45057.2020.9309946

Sitova, I., & Pecerska, J. (2020). Process Data Analysis Using Visual Analytics and Process Mining Techniques. *2020 61st International Scientific Conference on Information Technology and Management Science of Riga Technical University, ITMS 2020 - Proceedings*. https://doi.org/10.1109/ITMS51158.2020.9259296

Song, M., & Aalst, W. (2007). Supporting process mining by showing events at a glance. *Proceedings of 17th Annual Workshop on Information Technologies and Systems (WITS 2007)*, 139–145.

Stefanovic, D., Dakic, D., Stevanov, B., Lolic, T., & Marjanovic, U. (2021). PROCESS MINING IN THE MANUFACTURING CONTEXT: REVIEW AND RECOMMENDATIONS. *International Journal of Industrial Engineering: Theory, Applications, and Practice*, *28*(4). https://doi.org/10.23055/ijietap.2021.28.4.7287

Stertz, F., Rinderle-Ma, S., & Mangler, J. (2020). Analyzing process concept drifts based on sensor event streams during runtime. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Vol. 12168 LNCS*. https://doi.org/10.1007/978-3-030-58666-9_12

Suriadi, S., Andrews, R., ter Hofstede, A. H. M., & Wynn, M. T. (2017). Event log imperfection patterns for process mining: Towards a systematic approach to cleaning event logs. *Information Systems*, *64*(September 2016), 132–150. https://doi.org/10.1016/j.is.2016.07.011

Swinnen, J., Jans, M. J., & Vanhoof, K. (2012). *A Process Deviation Analysis – A Case Study*. 87–98.

Thiede, M., Fuerstenau, D., & Bezerra Barquet, A. P. (2018). How is process mining technology used by organizations? A systematic literature review of empirical studies. *Business Process Management Journal*, *24*(4), 900–922. https://doi.org/10.1108/BPMJ-06-2017-0148

Toosinezhad, Z., Fahland, D., Köroglu, O., & Van Der Aalst, W. M. P. (2020). Detecting system-level behavior leading to dynamic bottlenecks. *Proceedings - 2020 2nd International Conference on Process Mining, ICPM 2020*, 17–24. https://doi.org/10.1109/ICPM49681.2020.00014

Tu, T. B. H., & Song, M. (2016). Analysis and prediction cost of manufacturing process based on process mining. *ICIMSA 2016 - 2016 3rd International Conference on Industrial Engineering, Management Science and Applications*. https://doi.org/10.1109/ICIMSA.2016.7503993

Van Der Aalst, W. (2016). Data science in action. In *Process Mining*. Springer.

Van Der Aalst, W., Dongen, B., Günther, C., Rozinat, A., Verbeek, E., & Weijters, A. (2009). ProM: The Process Mining Toolkit. *Allergy*.

Waller, M. A., & Fawcett, S. E. (2013). Big data, predictive analytics, and theory development in the era of a maker movement supply chain. *Journal of Business Logistics*, *34*(4), 249–252. https://doi.org/10.1111/jbl.12024

Weijters, A., & Ribeiro, J. T. S. (2011). Flexible heuristics miner (FHM). *2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*, 310–317.

Yang, H., Park, M., Cho, M., Song, M., & Kim, S. (2014). A system architecture for manufacturing process analysis based on big data and process mining techniques. *Proceedings - 2014 IEEE International Conference on Big Data, IEEE Big Data 2014*, 1024–1029. https://doi.org/10.1109/BigData.2014.7004336

Appendix A

"Cut definition:

Let L be an event log with corresponding directly-follows graph $G(L) = \left(A_L, \rightarrow_L, A_L^{start}, A_L^{end}\right)$. Let $n \geq 1$. An n-th cut of G(L) is a partition of A_L into pairwise disjoint sets $A_1, A_2, \ldots, A_n$: $A_L = \cup_{i \in 1,\ldots,n} A_i$ and $A_i \cap A_j = \emptyset$ for $i \neq j$. Let $a \mapsto_L^+ b$ denote that a directed edge chain (path) exists from *a* to *b* in G Notation is $(\bigoplus, A_1, A_2, \ldots, A_n)$ with $\bigoplus \in (\rightarrow, \times, \wedge, \circlearrowleft)$. For each type of operator $(\rightarrow, \times, \wedge, \circlearrowleft)$ specific conditions apply:

- An *exclusive choice cut* of G(L) is a cut $(\times, A_1, A_2, \ldots, A_n)$ such that

  -$\forall_{i,j \in \{1,\ldots,n\}} \forall_{a \in A_i} \forall_{b \in A_j}\ i \neq j \ \Rightarrow a \nrightarrow_L b$.

- A *sequence cut* of G(L) is a cut $(\rightarrow, A_1, A_2, \ldots, A_n)$ such that

  -$\forall_{i,j \in \{1,\ldots,n\}} \forall_{a \in A_i} \forall_{b \in A_j}\ i < j \Rightarrow (a \mapsto_L^+ b \wedge b \not\mapsto_L^+ a)$.

- A *parallel cut* of G(L) is a cut $(\wedge, A_1, A_2, \ldots, A_n)$ such that

  - $\forall_{i\in\{1,\ldots,n\}}\ A_i \cap A_L^{start} = \emptyset\ \wedge A_i \cap A_L^{end} = \emptyset$,

  - $\forall_{i,j\in\{1,\ldots,n\}}\forall_{a\in A_i}\forall_{b\in A_j}\ i \neq j\ \Rightarrow a \rightarrow_L b.$

- A *redo-loop cut* of G(L) is a cut $(\wedge, A_1, A_2, \ldots, A_n)$ such that

  - $n \geq 2,$

  - $A_L^{start} \cup A_L^{end} \subseteq\ A_1,$

  - $\{\mathrm{a} \in A_1 | \exists_{\mathrm{i}\in\{2,\ldots,n\}}\exists_{\mathrm{b}\in A_i} a \rightarrow_L b\} \subseteq\ A_L^{end}$,

  -$\{\mathrm{a} \in A_1 | \exists_{\mathrm{i}\in\{2,\ldots,n\}}\exists_{\mathrm{b}\in A_i} b \rightarrow_L a\} \subseteq\ A_L^{start}$,

  -$\forall_{i,j\in\{2,\ldots,n\}}\forall_{a\in A_i}\forall_{b\in A_j}\ i \neq j\ \Rightarrow a \nrightarrow_L b,$

  - $\forall_{i\in\{2,\ldots,n\}}\forall_{b\in A_i}\exists_{a\in A_L^{end}} a \rightarrow_L b \Rightarrow \forall_{a'\in A_L^{end}} a' \rightarrow_L b$ and

  - $\forall_{i\in\{2,\ldots,n\}}\forall_{b\in A_i}\exists_{a\in A_L^{start}} b \rightarrow_L a \Rightarrow \forall_{a'\in A_L^{end}} b \rightarrow_L a'$."(Leemans et al., 2015)

Inductive algorithm steps:

I. **Construction of a Directly Follows Graph (DFG)**: A DFG is a graph in which nodes correspond to activities recorded in the event log and directed arcs correspond to directly-follows relationships. In this step, given an event log $L \in B(\mathcal{A})$, the corresponding directly follows graph $G(L) = \left(A_L, \rightarrow_L, A_L^{start}, A_L^{end}\right)$ is constructed using the method defined in (Van Der Aalst, 2016), where: A_L is the set of activities in L; $\rightarrow_L$ is the directly-follow relationship, A_L^{start} is the set of start activities; A_L^{end} is the set of end activities.

II. **Process relationships detection:** By iteratively *cutting* the directly follows graph, the original log is split into smaller sub-logs until they correspond to single activities and so process relationships can be detected. The process relationships detected are described formally by the operators $\bigoplus \in (\rightarrow, \times, \wedge, \circlearrowleft)$. $\rightarrow(A_i, A_j)$ means that in all the process traces that include the activity in A_i and at least one element of A_j, $A_i\ is\ always\ followed\ by\ A_j$. $\times(A_i, A_j)$ means that in a process trace $A_i\ is\ never\ followed\ by\ A_j$. $\wedge(A_i, A_j)$ means that in a process trace

A_i *can be followed by* A_j and vice versa. $\circlearrowleft (A_i, A_j)$ means that in a process trace A_i *can be followed by* A_j multiple times. In this step, the algorithm searches for specific process relationships by partitioning the directly follows graph into pairwise disjoint sets $A_1, A_2, \ldots, A_n$: $A_L = \cup_{i \in 1,\ldots,n} A_i$ and $A_i \cap A_j = \emptyset$ for $i \neq j$. The partitioning is done through iterative *cuts* of the directly follows graph, until the disjoint sets correspond to single activities.

III. **Construction of the process tree**: The original event log is mapped into a process tree, thanks to the operators discovered in step II. A *process tree* is a compact abstract representation of a block structured Petri net with a single start place and a single end place: a rooted tree in which leaves are labelled with activities and all other nodes are labelled with operators (Leemans et al., 2015).

IV. **Construction of the Petri net:** The process tree is converted into a Petri net (Buijs et al., 2013). A node of the Petri net can be a place (circle) or a transition (bar). The transitions represent process activities, and they are enabled only if all its input places hold a token. When they are performed, they generate a token in its output places. The black bars represent silent transitions which allow process activities to be skipped or some transitions to be enabled.

Process Tree definition:

"Let $\mathcal{A}$ be a definite set of activities, with $\tau \notin \mathcal{A}$ representing a silent activity. $\bigoplus \in (\rightarrow, \times, \wedge, \circlearrowleft)$ is the set of process tree operator discovered in step II. A process tree is recursively defined as follows:

- if $a \in A \cup \{\tau\}$, then Q=a is a process tree;
- if $Q_1, Q_2, \ldots, Q_n$ are process trees where $n \geq 1$, and $\bigoplus \in (\rightarrow, \times, \wedge, \circlearrowleft)$, then $Q = \bigoplus (Q_1, Q_2, \ldots, Q_n)$ is a process tree;
- if $Q_1, Q_2, \ldots, Q_n$ are process trees where $n \geq 2$, then $Q = \circlearrowleft (Q_1, Q_2, \ldots, Q_n)$ is a process tree.

The nodes of a process tree, both operator and activity nodes, are denoted as *N(Q).*(Leemans et al., 2015)

Heuristic algorithm description:

The heuristic algorithm is based on the computation of the directly-follow and dependency measures, and it is illustrated below. Considering an event log $L \in B(\mathcal{A})$ and two process activities $a, b \in \mathcal{A}$, the directly-follow measure $|a >_L b|$ represents how many times an activity *a* is directly followed by an activity *b* in all the traces that compose L. The dependency measure $|a \Rightarrow_L b|$ is the value of the dependency relationship between *a* and *b*, it is a value between -1 and 1. If *a* is often the cause of *b* the dependency measure will be a value close to 1, by contrast if *b* is often the cause of *a* its value will be close -1 (Van Der Aalst, 2016).

$$|a \Rightarrow_L b| = \begin{cases} \dfrac{|a >_L b| - |b >_L a|}{|a >_L b| + |b >_L a| + 1} & if \quad a \neq b \\ \dfrac{|a >_L a|}{|a >_L a| + 1} & if \quad a = b \end{cases} \tag{3}$$

By the combination of the directly-follow measures$|a >_L b|$ and dependency relations $|a \Rightarrow_L b|$, the heuristic algorithm can learn process splits and joints that are mapped graphically into a process model.