UNIVERSIDAD DE SEVILLA

DOCTORAL THESIS

Improving data preparation for the application of process mining

Author:
Belén Ramos Gutiérrez

Advised by:
Dra. Mª Teresa Gómez López
Dra. Antonia Mª Reina Quintero

A thesis submitted in fulfillment of the requirements
for the degree of Doctor in Computer Engineering

in the

IDEA Research Group
Department of Computer Languages and Systems
Escuela Técnica Superior de Ingeniería Informática

December, 2022

UNIVERSIDAD DE SEVILLA

TESIS DOCTORAL

Improving data preparation for the application of process mining

Autor:
Belén Ramos Gutiérrez

Dirigida por:
Dra. Mª Teresa Gómez López
Dra. Antonia Mª Reina Quintero

Memoria que presenta para optar al título de
Doctor en Informática

realizada en

IDEA Research Group
Departamento de Lenguajes y Sistemas Informáticos
Escuela Técnica Superior de Ingeniería Informática

Diciembre de 2022

To all the magnificent and empowered women in STEM.

This doctoral thesis has been funded by the European Commission with project 0754_-CIU3A_5_E, the Spanish Ministry of Science and Innovation through project PID2020-112540_-RB-C44, the Spanish Ministry of Science, Innovation and Universities by project RTI2018-094283-B-C33, and the Spanish Ministry of Economy and Competitiveness for project TIN2015-63502-C3-2-R. Funding has also been received from the Department of Economy, Knowledge, Business and University of the Regional Government of Andalusia, through projects US-1381375 and P20_01224. Finally, it has also received economic support thanks to knowledge transfer projects funded by Airbus Defence and Space S.A.

Comisión
Europea

Interreg
España - Portugal
CIU3A
Fondo Europeo de Desarrollo Regional
UNIÓN EUROPEA

GOBIERNO
DE ESPAÑA
MINISTERIO
DE CIENCIA
E INNOVACIÓN

GOBIERNO
DE ESPAÑA
MINISTERIO
DE CIENCIA, INNOVACIÓN
Y UNIVERSIDADES

GOBIERNO
DE ESPAÑA
MINISTERIO
DE ECONOMÍA, INDUSTRIA
Y COMPETITIVIDAD

Junta de Andalucía
Consejería de Economía, Conocimiento,
Empresas y Universidad

AIRBUS
DEFENCE & SPACE

UNIVERSIDAD DE SEVILLA

Abstract

Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

Doctor in Computer Engineering

Improving data preparation for the application of process mining

by Belén Ramos Gutiérrez

Immersed in what is already known as the fourth industrial revolution, automation and data exchange are taking on a particularly relevant role in complex environments, such as industrial manufacturing environments or logistics. This digitisation and transition to the Industry 4.0 paradigm is causing experts to start analysing business processes from other perspectives. Consequently, where management and business intelligence used to dominate, process mining appears as a link, trying to build a bridge between both disciplines to unite and improve them. This new perspective on process analysis helps to improve strategic decision making and competitive capabilities. Process mining brings together data and process perspectives in a single discipline that covers the entire spectrum of process management. Through process mining, and based on observations of their actual operations, organisations can understand the state of their operations, detect deviations, and improve their performance based on what they observe. In this way, process mining is an ally, occupying a large part of current academic and industrial research.

However, although this discipline is receiving more and more attention, it presents severe application problems when it is implemented in real environments. The variety of input data in terms of form, content, semantics, and levels of abstraction makes the execution of process mining tasks in industry an iterative, tedious, and manual process, requiring multidisciplinary experts with extensive knowledge of the domain, process management, and data processing. Currently, although there are numerous academic proposals, there are no industrial solutions capable of automating these tasks. For this reason, in this thesis by compendium we address the problem of improving business processes in complex environments thanks to the study of the state-of-the-art and a set of proposals that improve relevant aspects in the life cycle of processes, from the creation of logs, log preparation, process quality assessment, and improvement of business processes.

Firstly, for this thesis, a systematic study of the literature was carried out in order to gain an in-depth knowledge of the state-of-the-art in this field, as well as the different challenges faced by this discipline. This in-depth analysis has allowed us to detect a number of challenges that have not been addressed or received insufficient attention, of which three have been selected and presented as the objectives of this thesis.

The first challenge is related to the assessment of the quality of input data, known as event logs, since the requeriment of the application of techniques for improving the event log must be based on the level of quality of the initial data, which is why this thesis presents a methodology and a set of metrics that support the expert in selecting which technique to apply to the data according to the quality estimation at each moment, another challenge obtained as a result of our analysis of the literature. Likewise, the use of a set of metrics to evaluate the quality of the resulting process models is also proposed, with the aim of assessing whether improvement in the quality of the input data has a direct impact on the final results.

The second challenge identified is the need to improve the input data used in the analysis of business processes. As in any data-driven discipline, the quality of the results strongly depends on the quality of the input data, so the second challenge to be addressed is the improvement of the preparation of event logs. The contribution in this area is the application of natural language processing techniques to relabel activities from textual descriptions of process activities, as well as the application of clustering techniques to help simplify the results, generating more understandable models from a human point of view.

Finally, the third challenge detected is related to the process optimisation, so we contribute with an approach for the optimisation of resources associated with business processes, which, through the inclusion of decision-making in the creation of flexible processes, enables significant cost reductions. Furthermore, all the proposals made in this thesis are validated and designed in collaboration with experts from different fields of industry and have been evaluated through real case studies in public and private projects in collaboration with the aeronautical industry and the logistics sector.

UNIVERSIDAD DE SEVILLA

Resumen

Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

Doctor en Informática

Improving data preparation for the application of process mining

por Belén Ramos Gutiérrez

Inmersos en lo que ya se conoce como la cuarta revolución industrial, la automatización y el intercambio de datos están tomando un papel especialmente relevante en entornos complejos, como los entornos de fabricación industrial o la logística. Esta digitalización y transición al paradigma de industria 4.0 está provocando que los expertos comiencen a analizar los procesos de negocio desde otras perspectivas. En consecuencia, donde dominaban la gestión y la inteligencia de negocio, la minería de procesos aparece como enlace, tratando de construir un puente entre ambas disciplinas para unirlas y mejorarlas. Esta nueva perspectiva para analizar procesos ayuda a mejorar la toma de decisiones estratégicas y sus capacidades competitivas. La minería de procesos permite aunar en una sola disciplina la perspectiva de los datos y de los procesos, cubriendo todo el espectro de gestión de procesos. Gracias a la minería de procesos, y basándose en observaciones de su operativa real, las organizaciones pueden conocer el estado de sus operaciones, detectar desviaciones y mejorar su funcionamiento en base a lo observado. De esta forma, el la minería de procesos se constituye como un aliado, ocupando buena parte de la investigación académica e industrial actual.

Sin embargo, aunque esta disciplina cada vez recibe más atención, presenta serios problemas de aplicación cuando se quiere implantar en entornos reales. La variedad de los datos de entrada en forma, contenido, semántica y niveles de abstracción provoca que la ejecución de tareas de minería de procesos en la industria se convierta en un proceso iterativo, tedioso y manual, que requiere de expertos multidisciplinares con amplios conocimientos del dominio, la gestión de procesos y el tratamiento de los datos. En la actualidad, aunque se encuentran numerosas propuestas académicas, no existen soluciones industriales capaces de automatizar estas tareas. Por este motivo, en esta tesis por compendio abordamos la problemática de la mejora de los procesos de negocio en entornos complejos gracias al estudio del estado del arte, y a un conjunto de propuestas que mejoran aspectos de relevancia en el ciclo de vida de los procesos, desde la creación de logs, la preparación de logs, la evaluación de calidad de procesos y la mejora de procesos de negocio.

En primer lugar, para esta tesis se realiza un estudio sistemático de la literatura que permite conocer en profundidad el estado del arte en esta materia, así como los diferentes retos a los que se enfrenta esta disciplina. Gracias a este análisis profundo, se han identificado un conjunto de retos aún no abordados, o abordados con menos intensidad, de los que se han seleccionado tres, los cuales se han presentado como objetivos de esta tesis.

El primer reto identificado está relacionado con la evaluación de la calidad de los datos de entrada, los conocidos como log de eventos. Es necesario la aplicación de técnicas basadas en la medida y evaluación de la calidad de los datos iniciales, es por esto que en esta tesis se presentan de manera conjunta una metodología y un conjunto de métricas que dan soporte al experto a la hora de seleccionar qué técnica aplicar a los datos en función de la estimación de la calidad en cada momento, reto poco estudiado según el previo análisis de la literatura. Así mismo, se propone también el uso de un conjunto de métricas para evaluar la calidad de los modelos de proceso resultantes, con el objetivo de evaluar si la mejora de calidad de los datos de entrada tiene un impacto directo en los resultados finales.

El segundo reto identificado, es la necesidad de mejorar los datos de entrada utilizados en los análisis de los procesos de negocio, conocidos como log de eventos. Como en toda disciplina orientada a datos, la calidad de los resultados depende fuertemente de la calidad de los datos de entrada, por este motivo, el primer reto a abordar es la mejora en la preparación de los logs de eventos. En este ámbito, esta tesis propone la aplicación de técnicas de procesamiento del lenguaje natural para re-etiquetar actividades procedentes de descripciones textuales de las actividades del proceso, así como la aplicación de técnicas de clustering para ayudar a simplificar los resultados, generando modelos más entendibles desde un punto de vista humano.

Finalmente, el tercer reto detectado se centra en la creación de procesos de negocios optimizados, por lo que presentamos una propuesta para la optimización de recursos asociados a procesos de negocio, que, a través de la inclusión de la toma de decisiones en la creación de procesos flexibles, permite reducir costes de manera significativa. Todas las propuestas realizadas en esta tesis están validadas y diseñadas de manera conjunta con expertos de diferentes ámbitos de la industria y han sido evaluadas a través de casos de estudio reales en proyectos públicos y privados en colaboración con la industria aeronaútica y el entorno logístico.

Acknowledgements

First of all, I would like to thank my thesis supervisor, Mayte. Thank you for being a magnificent example to follow, for your tireless way of working, and for your understanding when I needed it. Since the beginning, I have been learning from you and I am certain that I could not have asked for a better supervisor or someone who could have inspired me as much.

To my supervisor, Toñi. My first female reference when I started degree studies. Thank you for your unique way of teaching, which helped me not to give up in the first years, and also for everything I have learnt from you at this stage of research.

I must also thank Ángel Varela for his support and help whenever I needed it. You are the colleague everyone would like to have around.

I would also like to thank all my colleagues in the Research Hub, from who I have the opportunity to learn something new every day and with who it is always a pleasure to work, as well as all of the researchers who have allowed me to work with them on the creation of this thesis, specially to Luisa Parody.

I cannot forget my family, the driving force of my life. None of this would have been possible without the unconditional love of my parents and brother and their wonderful way of teaching me the importance of effort and perseverance. My grandmothers, two extraordinary women that I love with all my heart. And my grandparents, two angels who always take care of me and who I hope that wherever they are, they are proud of their granddaughter.

To my other half, Alejandro. Thank you for your understanding and unconditional support, for believing in me when I do not and for accompanying me on this journey. We always said we could do it together and since then we have not stopped trying. I cannot wait to achieve new challenges with you.

Contents

Chapter 1

Introduction

This chapter introduces this thesis, and to do so is structured as follows: Firstly, Section 1.1 describes the context in which this thesis is framed and motivates why this work is necessary. Secondly, Section 1.2 introduces the background needed to understand the rest of the dissertation. Next, the objectives of our work are described in Section 1.3. After that, the research methodology and resources used to develop this thesis are detailed in Section 1.4. Finally, the chapter is concluded with a roadmap of the document.

1.1 Context and Motivation

The aim of this section is to contextualise the thesis, presenting the research subject, the problematic issues it presents and introducing some relevant concepts. In addition, the main objectives to be covered in this thesis will also be outlined.

1.1.1 Context

Today, technology is advancing at a vertiginous speed, offering progress in all fields, allowing for the optimisation of resources and methodologies. This process, known as digital transformation [134], helps us apply digital capabilities to improve the processes, efficiency, and competitiveness of organisations. Digital transformation involves horizontal and vertical changes, leading not only to improvements at the operational level, but also at the cultural, organisational, and strategic levels. This transformation, which is no longer an option, but a necessity, has generated new needs in the storing and processing of information, leading to the rise of technical areas such as data science, data analytics, and data mining. However, in most cases, it is not sufficient to store and analyse data because this information and analysis must be aligned with the organisation's business processes.

The current trend to automate the processes in the industry has promoted the known as Industry 4.0 paradigm. Automating processes also produces a large number of data events that can be analysed to improve organisations from a strategic point of view [5]. In fact, Roblek et al. [103] point out that it would be advisable to understand the concept of Industry 4.0 as the digitisation of business processes instead of the point of view of robots and automation. To this end, a growing number of organisations are using Business Process Management Systems (BPMS) to support their operational processes and analyse them from an "AS IS" and "TO BE" perspective [5]. Unfortunately, these models are often disconnected from the events generated by the different systems, making the analysis of the results complex and often dependent on the nature, format, and granularity of these complex events.

Combining process models with event data brings about new forms of process-focused analysis. In addition, data generated by human processes or generated by ad hoc processes can also be used in the analysis. In this regard, disciplines such as Business Process Management (BPM) and Business Intelligence (BI) are receiving increasing attention, but do not fully meet the expectations of experts. To address these gaps, the process mining discipline analyses the event data produced to discover, monitor, and improve real processes [15]. The increasing automation of business processes has provoked the creation of an enormous number of events whose management requires to face new challenges, known as Complex Event Processing (CEP). The term complex event processing [75] refers to a group of technologies used to discover relationships between single events to infer the existence of relevant information, such as timing, causality, membership, or the existence of behavioural patterns or correlation. CEP is used to monitor, guide, and optimise real business process models, such as logistics or assembly line processes.

Process mining is a mature area with more than two decades [17]. Although there has certainly been significant progress and a proliferation of proposals from the scientific community in this area, most of these proposals focus on highly controlled experiments oriented towards the academic world, where some areas of practical application stand out, such as healthcare or studies based on data from ERP-type information systems [43]. Even today, it is still a challenge to face real-life cases of successful automated process mining applications in complex and distributed environments [81], whose challenges must still be addressed and solved.

IEEE Task Force on Process Mining [15] and other studies [101, 118] have posed the definition and discovery of complex processes as the main challenge in this discipline, highlighting the special relevance of this issue in very complex environments, such as logistics or supply chains, where other business processes are needed to support the entire activity [9].

In-depth studies of the state of the art establish logistics and manufacturing as two of the most active fields of contribution in recent years [109]. Furthermore, the authors emphasise the difficulties of these application areas due to three intrinsic characteristics of the context: *(i)* problems in defining process instances and tasks for the macro-processes specific to the area; *(ii)* the level of aggregation of data that are not stored as an event, making them difficult to process by the usual process mining tools, and; *(iii)* the participation of different parties involved in processes. These challenges reveal the need to adapt current process mining techniques to complex environments.

Process mining uses event logs, where each event refers to a processing activity and is related to a process instance. In these logs, which represent the primary material for process mining, different techniques can be applied: *(i)* generate process models based on the actual behaviour observed; *(ii)* check the alignment among events and processes, and; *(iii)* identify process improvements based on what has been observed.

As a consequence, obtaining the event logs is a key task and supposes tackling some difficulties that have produced a wide field of study. One of the main challenges on this topic lies in the nature of processes and events: *Although processes usually follow a top-down strategy, events* *follow a bottom-up one*. This is why it is important to establish a correct level of abstraction to deal with the conflicts that arise in complex environments, where, typically, information is stored neither process-oriented nor event-oriented.

In these complex environments, processes are usually collaborative and distributed, which implies that there are cooperative and coordinated tasks that need to be discovered and managed. At this point, problems related to the levels of abstraction must be faced, as it is important not only to know that information of different nature is involved (processes and events), but also that processes and events from different sources play an important role. This is particularly relevant because most of the proposed solutions found in the literature so far work with events coming from the same repository, which is often neither easy nor possible. This is a major challenge, as not only are the processes, their definition, and discovery complex but so will the preprocessing of the event logs for this purpose. Under these conditions, aspects such as defining what will be considered as an instance, deciding what level of granularity to use to define each task, or estimating what impact the addition of information resources has on the very definition of the execution trace acquire high importance.

In addition, all these discovery tasks must be performed on the basis of **event data quality**. Otherwise, the results obtained will not be satisfactory, resulting in excessively simple process models, which do not help to understand the organisation's real-life operations, or, on the contrary, excessively complex models, known as spaghetti models [6], which are impossible to handle from a human point of view.

Although there are many approaches in the literature to measure the quality of the discovered processes [5, 84] (mostly based on graph metrics), an automated way to measure the usability of processes without the involvement of the expert has not yet been established. In this way, discovery tasks become a highly manual process, where the person in charge must have notions of data science and process mining and must know the business domain thoroughly. Since this profile is very infrequent, it would be advisable to get an automated solution capable of inferring the level of usability of a process model given a series of preliminary domain conditions. This is yet another challenge to face.

In relation to this challenge, there is another unresolved area of study: the relationship between the input data and the quality of the discovered processes [22]. Process discovery tasks in real-life environments are usually very time-consuming and resource-intensive. In some cases, after a long waiting time, when the model is finally obtained, it becomes useless. For this reason, a new challenge arises, as it would be desirable to be able to obtain an a priori evaluation of the input data, which helps to know the expected quality and usability before discovering the process models. This would help the expert decide how the data can be pre-processed before developing the discovery process.

The next subsection (Section 1.1.2) provides an in-depth look at the different areas of improvement in data preparation tasks for the application of process mining.

1.1.2 Motivation

During the development of this thesis, we have relied on real application cases to detect the real needs of the industry to analyse the information available and try to apply process mining techniques to optimise their business. Thus, we have collaborated with organisations in the aeronautical (assembly lines) and the port-logistics (logistics and supply chains) sectors. Thanks to this transfer of knowledge and collaborative work, we have been able to identify some of the major problems that these large industries face on a daily basis.

Several proposals can be found to improve process mining algorithms, but before and after applying those techniques, some issues must also be tackled. This thesis focusses on the associated tasks of process discovery, both upstream tasks (such as the pre-processing of the input event log, the log data quality, the transformation and integration) and downstream tasks (related to the quality and usability of the discovered processes).

The tasks preceding process mining are particularly relevant when data come from complex scenarios, from distributed sources, and have different structures, formats, and granularity. In these contexts, it is important to identify the different data sources, establish integration mechanisms between them, transform and standardise the data to be able to work in a homogeneous way, and carefully analyse them to determine which of them will be useful, what amount and type of information are available, and establish how far it will be possible to go in the analysis with these input data.

In addition, process mining tasks require data to be provided in a very specific format known as eXtensible Event Stream (XES) [16]. Unfortunately, systems rarely generate their event traces in a format compatible with XES and ready to be used for discovery tasks; as a consequence, these event traces in XES format must be built from business data. However, this requires extensive knowledge of the business, as the construction of the log will determine the possible discovery of subsequent processes. Therefore, decisions such as what to extract and from where to extract will have a major influence on the subsequent analysis. In addition, many of the tools designed as a middleware between the information system and the process mining tools require extensive knowledge of the database management systems used in the organisation, which requires that the expert have extensive knowledge of the business, the technologies that support it, the structure of the information stored, and the format of the event log. For these reasons, this thesis aims to provide a framework that helps experts in the identification, extraction, integration, and transformation business data to be used as events capable of generating process models that reflect the real operations of organisations.

The results of discovery tasks are significantly influenced by the quality of the input data, which requires the use of data quality measurement and assessment techniques to ensure the quality of the discovered models. Both the logistics and manufacturing environments integrate data coming from various sources whose quality cannot be ensured. Thus, the measurement and evaluation of event data quality must be done in a context-specific way, depending on the data usage scenario. In this thesis, we also seek to improve the way in which event data quality is assessed and measured.

Additionally, once the quality of the input data has been computed, enhancements can be made to the input data to improve the quality indicators. In this thesis, we work on different techniques related to simplification, as well as enrichment from textual data, whose requirements were derived from the real data analysed.

This whole process always ends up in a process discovery task. However, which algorithm to choose and how to parameterise it is not a trivial task either. The difficulty of the discovery task will entirely depend on the complexity of the environment in which the events are generated, and the considerations of the users about how the process models must be. The discovery of federated [3], distributed, and choreographed processes will require additional steps. In this thesis, we address this problem in two real environments of a highly complex nature.

Process mining techniques allow the generation of process models based on the actual behaviour observed [4] and allows discovering the business process from different points of view. However, it is not very well established what quality metrics to use or how to compare and measure the quality and complexity of the processes discovered after the discovery task. Currently, process mining tools are focused on discovery and data analysis, but do not allow measuring the quality and complexity of the discovered processes.

In an equivalent way as with input data, the quality and usability of the resulting process models must be measurable and assessable. Only in this way will it be possible to establish empirical and automated methods to improve it. In this thesis, we work to identify relationships between the quality of the input data and the quality models and establish useful metrics and indicators of quality and usability.

To integrate every step mentioned in a single solution, we propose a framework to support the whole process mining pipeline oriented towards the data log evaluation.

This thesis was developed as part of the IDEA Research Group [1]. This group has proven experience in the application of business process management, process mining, data preparation, and governance. The need to use techniques based on process mining (and consequently data preparation for discovery tasks) derives from research national, regional, and local projects, as well as from collaborative projects with private organisations. The following is the list of research projects in which this thesis plays a crucial role:

- *CIU3A: Centro de Innovación Universitario de Andalucía, Alentejo y Algarve Interreg Europa*[2] (0754_CIU3A_5_E). This project is carried out in collaboration with Portel. It aims to improve the modelling and simulation of business processes in the industrial systems of the Seville port-logistics environment.

- *Aether-US*[3] (PID2020-112540RB-C44). This project aims to improve the way data is managed by following a context-aware basis to support business processes.

- *ECLIPSE*[4] (RTI2018-094283-B-C33). This project focusses on improving data quality and data security in business processes.

[1] IDEA Research Group: https://www.idea.us.es/

[2] CIU3A: https://investigacion.us.es/sisius/sis_proyecto.php?idproy=31997

[3] Aether-US: https://investigacion.us.es/sisius/sis_proyecto.php?idproy=33855

[4] ECLIPSE: https://investigacion.us.es/sisius/sis_proyecto.php?idproy=29615

- *METAMORFOSIS*[5] (US-1381375). This project aims to support digital transformation processes by improving data management, business processes, and security governance.
- *COPERNICA*[6] (P20_01224). This project aims to improve business processes through data governance.
- *Sequoia-US*[7] (TIN2015-63502-C3-2-R). This project aims to improve business process management and improve data-oriented decision-making.
- *Clean Sky 2*[8] (P036-19/E08). This project is carried out in collaboration with Airbus. Its objective is to improve the diagnostic of business processes in the aircraft assembly process.
- *Troubleshooting para Sistemas Ciberfísicos*[9] (P033-19/E08). This project is carried out in collaboration with Airbus. Its objective is to improve the diagnostic of business processes in the aircraft assembly process.

This thesis is framed on the data management, data preparation, and process discovery part of each of these projects. The purpose is to facilitate decision-making, discovery, diagnosis, and optimisation of business processes.

1.2 Background

The main research areas that are closely related to this thesis are outlined in this section. The necessary fundamentals related to business processes are introduced in Section 1.2.1. Then, Section 1.2.2 dives into the concepts related to process mining. In this section, we also dive into essential aspects of this thesis such as event logs (Subsection 1.2.2.1), event log preparation (Subsection 1.2.2.2), event log quality (Subsection 1.2.2.3), process discovery (Subsection 1.2.2.4) and the assessment of quality of the discovered processes (Subsection 1.2.2.5).

1.2.1 Business Processes

Business process management comprises a set of methods to model, observe, and improve the activities carried out in an organisation to achieve its strategic objectives according to a set of constraints that govern its behaviour [56]. According to [130], a business process is made up of a set of activities belonging to one or more organisations. These activities are carried out in a coordinated manner to accomplish a business purpose. Therefore, modelling these processes is the first step in understanding how activities performed by organisations are related, both those that are executed manually and those that are performed with the support

[5] METAMORFOSIS: `https://investigacion.us.es/sisius/sis_proyecto.php?idproy=33186`

[6] COPERNICA: `https://investigacion.us.es/sisius/sis_proyecto.php?idproy=33720`

[7] Sequoia-US: `https://investigacion.us.es/sisius/sis_proyecto.php?idproy=26974`

[8] Clean Sky 2: `https://investigacion.us.es/sisius/sis_proyecto.php?idproy=31890`

[9] Troubleshooting para Sistemas Ciberfísicos: `https://investigacion.us.es/sisius/sis_proyecto.php?idproy=31892`

of an information system. To this end, business process management brings together techniques and methods to support the business process life cycle [48]. Management of the life cycle of business processes comprises different phases [21]: identification, discovery, analysis, redesign, implementation, and monitoring and control. Different imperative and declarative languages can be used to model business processes at the different stages of their life cycle. The language focused on expressing a set of guidelines to control how components of a process can be integrated is known as the business process modelling language [48]. One of the most widely used business process modelling languages is BPMN [89], although others are also very common in the literature, such as Petri nets [63].

1.2.2 Process Mining

Process mining involves a set of tools and methods that provide critical information for BPM and accelerate the implementation of complex IT solutions. The goal of process mining is to provide techniques and tools to discover processes, control, data, and organisational and social structures from event logs [12]. It is based on process, model-based approaches combined with data mining, bringing otherwise static processes to life, using, in a process context, the massive data that are generated every day. This enables management trends related to business process improvements in competitive and rapidly changing environments, such as Six Sigma [112], TQM [74], CPI [104], and CPM [54], to benefit from it [5, 7, 9].

Figure 1.1 shows the illustration traditionally used in the literature to explain process mining. It shows the three main tasks executed in a process mining context, what kind of information each process mining consumes, and what kind of output it produces. These three main tasks are as follows:

- **Process Discovery** [4] uses an event log that recovers the evidence of process executions and generates a process model that represents the operation of the organisation based on those observations. Figure 1.1 shows that process discovery works by consuming the digital footprints generated in information systems during the execution of business activities and generates as output a process model that represents how that process occurs in the real world.

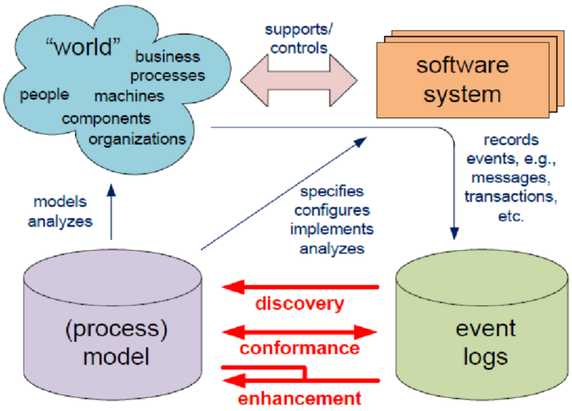


FIGURE 1.1: Classical illustration of process mining [5]

- **Conformance Checking** [105] could be considered as a diagnostic task, where the input is made up of the expected model of the process and a log of events that represent the observations of the process. Therefore, with the observed and expected results, it is possible to study the adjustment or deviations of the expected operation compared to what actually occurred. Figure 1.1, shows how the conformance-checking technique involves the process model and the event logs. The comparison of these two models allows experts to identify areas for improvement and change.

- **Process Enhancement** [7] improves an existing process model by using information about the observations of the process underlying the event log. While conformance checking is in charge of aligning the model and the observations, process enhancement is about extending and improving the reference process model so that it is better adapted to the observed operation of the organisation. This task is executed when problems such as bottlenecks, insufficient service levels, or excessive production times are detected. Figure 1.1 depicts an iterative behaviour by representing an extension and improvement of the process model. The enhancement is obtained thanks to the insight gained from the discovery and conformance tasks.

Process mining allows organisations to extract valuable knowledge that enables them to gain strategic advantages over their competitors [62]. In a general way, process mining benefits organisations thanks to: understanding better how processes are actually carried out; improving existing processes; improving productivity and reducing costs; and comparing processes and their effects. Additionally, it is relevant to highlight that process mining does not perform an off-line analysis, but it can also be beneficial for those organisations that need information about their processes in real-time, since, in the case of Process Discovery, it can also be used as a predictive tool [47, 25].

Once process mining has been introduced in general terms, the following subsections introduce more in depth how events are specified in event logs, how they can be prepared, how to assess their quality, how they are involved in process discovery, and, finally, how the quality of the discovered process models can be assessed.

1.2.2.1 Event Logs

Event logs are the cornerstone of process mining, and in this context, they must satisfy the following requirements: *(i)* each event has a unique *identifier*, *(ii)* each log contains information representing a single process, *(iii)* each event contained in the log is related to a single instance of the process (also called *case*) and *(iv)* the events are related to an activity of the process.

When talking about processes, the existence of a life cycle is assumed, which implies a temporal order that explains a logical sequence/concurrence of activities. From this, the need arises not only to have a set of events but also that they can be ordered sequentially/concurrently and temporally so that each event must be associated with a time stamp. Thus, event logs are made up of event traces, which represent the execution of an instance/case of the process. A trace contains a unique sequence of events that occur in an ordered manner over time. In addition, events may have other attributes that help to provide more information on the process or allow analysis from different perspectives. The most common extra attributes are *costs, resources or location.*

For any element contained in an event log, [5] defines the following cardinalities:

- Each process can have one or more activities.
- Each activity belongs to only one process.
- Each process instance (case) represents the execution of a single process type.
- Each activity instance represents the execution of a single type of activity.
- Each activity instance belongs to a single process instance (case).
- There can be one or more instances for each (activity-case) tuple.
- Each event belongs to a single case and activity instance.
- For an activity instance, there may be multiple events.
- Each process instance attribute belongs to a single instance and must have a name and a value.
- Each process event attribute belongs to that single event and must have a name and a value.

XES (eXtensible Event Stream) is the de facto standard for defining event logs. The XES components were conceptualised in the metamodel specified by Gunter in [60, 122] (Figure 1.2), and an instance of this metamodel in XML is shown in Listing 1.1. This listing shows an example of an XES file. This file includes a trace (lines 8-39). This trace is made up of a set of events (lines 10-16, 17-23, 24-30, 31-37). Each event has a name, a timestamp, a cost, a resource (a person) and an id to be indexed.

As can be seen in the listing, the log, trace and event elements only define the structure of the document, i.e. they do not contain any information themselves. In any log in XES format, there are two types of global attributes: traces and events. Traces must have a name, which is represented with the *concept:name* attribute (see line 11 of listing 1.1). Events must have a name, and a timestamp (see attributes *concept:name* and *time:timestamp* in the listing 1.1). Furthermore, events may contain attributes of type *org:resource* and others, which may be domain-specific. In this way, XES covers all perspectives to be provided by process mining, through the use of the key extensions: *concept, time and organisational.* The *concept* extension (line 4) is used to give an identifier (name) to traces and events. In the case of traces, it is usually the identifier of the process instance, while for events, it is typically the name of the activity being executed in that event. The extension *time* (line 7) defines the time of occurrence of events in a timestamp format, which allows both date and time to be taken into account. The *organisational* extension (line 5) can have three types of attributes: *source, role, group.* The attribute *source* refers to the resource that triggered or executed the event, while the attributes *role* and *group* refer to the capabilities of the resource that triggered or executed the action in

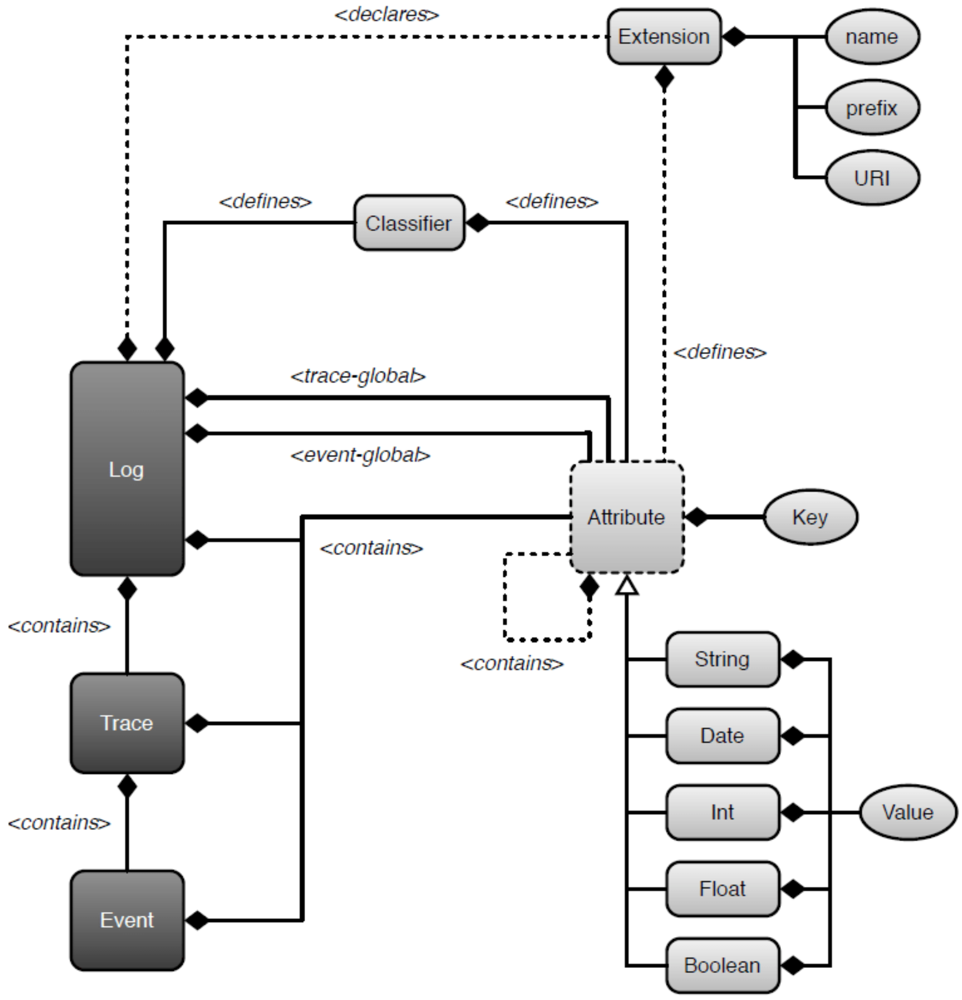


FIGURE 1.2: XES Metamodel [60]

terms of permissions, policies, position, and roles played in the organisation. Finally, in addition to the extensions shown in the listing 1.1, two additional extensions are possible: *life-cycle, semantic*. *Life-cycle* is used when the attribute defines a traditional transactional event model, while the *semantic* attribute is the basis of SA-MXML, which aggregates ontology concepts.

Listing 1.1: An example of a XES file [26]

```
<?xml version='1.0' encoding='UTF-8'?>
<log xes.version="1849-2016">
  <string key="origin" value="csv"/>
  <extension name="Concept" prefix="concept" uri="http://www.xes-standard.org/concept.
      xesext"/>
  <extension name="Organizational" prefix="org" uri="http://www.xes-standard.org/org.
      xesext"/>
  <extension name="Cost" prefix="cost" uri="http://www.xes-standard.org/cost.xesext"/>
  <extension name="Time" prefix="time" uri="http://www.xes-standard.org/time.xesext"/>
  <trace>
    <string key="concept:name" value="1"/>
    <event>
      <string key="concept:name" value="register request"/>
      <date key="time:timestamp" value="2010-12-30T11:02:00.000+01:00"/>
      <int key="cost:total" value="50"/>
      <string key="org:resource" value="Pete"/>
      <int key="@@index" value="14"/>
    </event>
    <event>
      <string key="concept:name" value="examine thoroughly"/>
      <date key="time:timestamp" value="2010-12-31T10:06:00.000+01:00"/>
      <int key="cost:total" value="400"/>
      <string key="org:resource" value="Sue"/>
      <int key="@@index" value="15"/>
    </event>
    <event>
      <string key="concept:name" value="check ticket"/>
      <date key="time:timestamp" value="2011-01-05T15:12:00.000+01:00"/>
      <int key="cost:total" value="100"/>
      <string key="org:resource" value="Mike"/>
      <int key="@@index" value="16"/>
    </event>
    <event>
      <string key="concept:name" value="decide"/>
      <date key="time:timestamp" value="2011-01-06T11:18:00.000+01:00"/>
      <int key="cost:total" value="200"/>
      <string key="org:resource" value="Sara"/>
      <int key="@@index" value="17"/>
    </event>
    ...
  </trace>
</log>
```

1.2.2.2 Event Log Preparation

The preparation of event logs could be understood from the point of view of log construction and log improvement. In some cases, construction does not occur with only an extraction job, but integration, cleaning, normalisation, and improvement tasks need to be carried out. In the same way, when the results obtained are not as desired, further work is needed to improve the event log, with the expectation of also improving the resulting discovered model. This section introduces the different solutions existing in the literature for log preparation, both from the point of view of construction and improvement.

As the quality of the results obtained in process mining is highly dependent on the logs used, a significant amount of academic research is done in the area of event log preparation. These proposals are based on the fact that all events that occur in a moderately digitised organisation leave a trace in the databases that support the operation of the business, either explicitly or implicitly. This leads to numerous approaches for the construction of event logs from relational and non-relational datasources.

Integrating information in an event log from relational and non-relational databases of organisational information systems is a significant challenge in itself. This difficulty is further increased if we consider that the information is not always stored in the same source. The combination of process-related data from different sources is still an unsolved challenge in process mining environments today. There are different approaches for integrating information from relational and non-relational databases into an event log. However, all proposals lack automation, and, as a consequence, the data combination tasks are carried out completely manually. An example of this is shown in [96], whose objective is to build a process based on the records left by software developers' activities in software repositories. These repositories are distributed on different platforms, and the combination of their data is done manually or with ad-hoc solutions oriented towards the identification of users based on meta-data. In [80, 36] the need to determine the possible source of events and a way to combine all the information is highlighted. In fact, as a simplification of the problem and a challenge to be tackled in the future, the assumption of completeness and existence of the event log from different sources is mentioned. The author of [32] focusses on the need to combine events that occur in sensors such as oscilloscopes and gyroscopes of different devices. To combine these data, clustering techniques are used to help grouping the called similar events. After the clustering of events, the intervention of a domain expert is needed to help identify and correlate elements to support the manual construction of an event log that combines the different sources.

The approaches whose focus is cleaning the event logs are mostly concerned with outliers, noise treatment, and detection of infrequent behaviour. The problem of detection of outliers has been addressed in [88] by using auto-encoders, a type of deep neural network capable of reconstructing its own input and learning the complex relationships between values and attributes in an event log. However, it is not always recommended to remove all outliers from the event log [113], and statistical measures to evaluate run-time outliers are proposed to determine which are candidates to be removed.

Regarding noise detection, it should be noted that to choose the process that best represents the most common behaviour, infrequent patterns in process discovery are often treated as noise [76] and deleted from traces of the log [108]. In relation to the detection of infrequent behaviour, there are two types of approach: those that apply filtering based on frequency [40] and others based on the discovery of a chaotic set of activities that occur frequently [116]. Filtering can be performed in a variety of ways, including *(i)* removing events that do not fit the norm [40, 108], *(ii)* incorporating it into the discovery process [70, 82, 128, 31], *(iii)* filtering traces in an unsupervised [55] or supervised manner [38], and *(iv)* incorporating previous steps for problem clustering, making it easier to distinguish between traces from different points of view or to categorise different types of behaviour [72, 114].

In relation to approaches to normalise event logs, some studies have looked at event log preprocessing to enhance discovery tasks when the real-world logs are written in natural language. This is achieved by automatically identifying and categorising eight different semantic roles in event data, as stated in [102]. Semantic-based methods are used to collect and normalise event log text data in [46]. By identifying incorrect activity labels that result in ambiguity and consistency, several forms of analysis have been performed to improve the labels of activities in a process model [95]. However, in most cases, these ideas begin from event data in natural language with a strong process-oriented structure and a straightforward and precise grammar.

Regarding improvement tasks, they could be classified in two types: tasks whose goal is to enrich the event logs with extra information and those that pretend to simplify the event logs. In relation to event log enrichment, it is worth noting that most of the proposals made in this area are based on the assumption of the prior existence of a log of events that needs to be improved. Sometimes, it is possible to extract very generalised event logs that can provide a very generalised perspective of the process under study. However, within the context in which this thesis is defined, it is very common to have a considerable amount of process-related information, whose presence in the event log can be of great help in understanding what is actually happening. Most academic research on process mining focusses on developing new techniques or improving existing ones, ignoring the limitations that can arise from using a poor quality event log. In this area, there are very few approaches, most of them related to the healthcare domain. As an example, in [80], the authors propose the use of information provided by indoor location systems to solve temporal inconsistencies and inaccuracies, missing or miscorrelated events, missing process instances, missing traces and event attributes, as well as the absence of the resource that executes each action. However, this approach is made only at the conceptual level, identifying needs and potential solutions, without demonstrations. Furthermore, it should be noticed that this work also highlights the need to extract and combine information from the internal location system with the information that supports the operation of the hospital and its translation in a semi-automated way to an XES log. [133] proposes the enrichment of the event log using non-contextual information, namely sentiment analysis based on national news texts. The authors propose that three measures of influence should be added to the event log as a user-defined attribute for each event in the log. Finally, enrichment with context-internal information is proposed in [50]. Other authors first proposed to assess the quality and usability of the event log and later executed an event log preprocessing [115]. Some of the proposed improvements include correcting timestamp errors [39, 45, 53], repairing the identification of cases, or even relabelling activities [106, 107].

Simplification is motivated by the fact that process mining techniques still have some limitations that become very present when applied to real-life case data. Process discovery techniques work very well when the event logs are complete and of good quality, generating well-structured processes. However, in the real world, processes suffer from various deviations and alterations that lead to very chaotic and unstructured processes. This is when Process Discovery techniques begin to present problems in both discovery and visualisation, generating complex models that are very difficult to understand. In [114], the flexibility present in some domains is highlighted, making possible the grouping of traces, dividing them into homogeneous subsets where each subset represents a less complex process. This approach to trace partitioning was also proposed in [57]. Other studies that have identified the need to simplify the processes obtained have also used clustering as a strategy to reduce complexity [29], this time not in the traces themselves, but in the raw data, dividing the users to group them by similarity and generating, in an unspecified way, different logs for each group. Another way to simplify processes has also been addressed, not only at the event log level, but also from the point of view of discovery algorithms, in [61, 11] the use of fuzzy mining algorithms (Fuzzy Miner) is proposed. For the same purpose, the Heuristic Miner discovery algorithm [23], designed to deal with noise and the most successful option when only the "main" observed behaviour in an event log is to be obtained, has sometimes been used. In [59], new Fuzzy Miner approximations and event log-based metrics are used to provide different levels of abstraction when obtaining the model to reduce or increase the complexity of the model according to the parameters indicated by the user.

1.2.2.3 Quality of Event Logs

The concept of data quality becomes particularly relevant when dealing with data extraction from different sources, as these data will use different keys, formats, and even syntaxes and, as a consequence, will have a significant impact on the quality of event logs [30]. In the Process Mining Manifesto [15], four criteria are outlined to help determine the quality of the event logs. When these criteria are met, they are referred to as top-quality logs:

- *They must be reliable*: it must be possible to assume that the recorded events accurately represent real observations and that the attributes of the events are correct.
- *They must be complete*: given a scope, no event can be missing.
- *They must have semantics*: all data must be part of a well-defined semantics where all similar events represent the same thing and no inconsistencies occur, which implies that there can be one or more ontologies.
- *They must be secure*: the privacy and security of the information stored in the logs must be taken into account during the construction of the event logs.

It is important to highlight some concepts and guidelines set out in [2, 5] to ensure that a log has the desired quality and information:

1. All references and names must have clear semantics, meaning the same thing to everyone involved in creating and analysing event data.
2. There should be a structured, hierarchical form of naming attributes which should be agreed on and belong to the domain of the organisation.
3. References and names should be stable and never depend on context or elements such as time, language or region.
4. Attribute values should be as precise as possible, which is especially important in the case of temporal attributes. Furthermore, if the desired level of precision is not available, this should be explicitly stated.
5. If there is uncertainty about the true occurrence of an event, this should also be made explicit, preventing a possible outlier from introducing noise into the final model.
6. Events must be minimally ordered and stored. This can be done implicitly by using timestamp attributes or explicitly by giving them a specific order in the log.
7. Whenever possible, it is a good idea to store transactional information on events.
8. It is recommended to periodically perform automated consistency and correctness tests on the event log.

9. It should be ensured that different models can be compared with each other over time in their different variants, using the same data collection principles.

10. You should not add additional information to serve as a starting point for the analysis. Whenever possible, the log data should be raw.

11. You should not eliminate events and ensure provenance, as it is essential to ensure that the results will be reproducible.

12. You should ensure privacy without losing the correlations between events.

As stated above, a significant issue that has been the subject of research over the past few decades is how to quantify and evaluate the potential to leverage their quality. In addition, event logs appear in new contexts and include new features that further reinforce the need to measure, extend, and adapt different quality dimensions [20, 123], such as completeness, accuracy, and simplicity, to the context of the business process [119].

1.2.2.4 Process Discovery

Process discovery includes a set of techniques and tools that help to automatically discover processes based on the behaviour of a process represented through an event log. The first attempts in this discipline started with [41], where the authors analysed the process data to produce formal models using three methods (Markov, KTail, based on [27] and RNet, starting from [44]). In later years, this study evolved by its own authors, resulting in [42], which presented techniques for discovering patterns of concurrent behaviour from traces of workflow events. However, the results of the proposed methodology were not an explicit representation of the process, but a set of dependency relationships between the different events present in the processes under study. In relation to this need, in the same year [13] was published, as a result of previous work [14, 125, 126, 10], where the process discovery algorithm considered today as the baseline algorithm, known as $alpha$-$algorithm$, appeared for the first time. In this case, the result was an explicit process model in a standardised format, namely Petri Nets [63], which allowed any workflow to be analysed without a priori knowledge of the underlying process. Although this algorithm handled concurrency and produced the intended results, it had some limitations dealing with noise, infrequent or incomplete behaviour, and requires complex routine constructions. Therefore, although it is a good theoretical proposal, it is difficult to use it in practise.

To address these problems, different solutions have emerged for different purposes. These solutions have been classified in the literature in various ways [5, 109, 94]. In this thesis, we focus on the application of Heuristic and Inductive Mining:

Heuristic Miner [127, 129] algorithms differ from the baseline in that they take into account not only the sequence of events, but also the frequency in which they occur, with the aim of building models in which infrequent behaviour does not appear, thus dealing with noise and exceptional cases, which in the $alpha$-algorithm represents a limitation.

Inductive Miner [70] algorithms are ones of the most flexible and scalable techniques, so they are also the most used by the community and are considered the most robust. Inductive Miner algorithms are responsible for discovering models that do not present robustness problems (*soundness*) [8]. These algorithms generate process models that are able to reproduce any trace of the event log (hence, it has full fitness). In addition, since there are no repeated activities, it tends to generate simple and generalised models. However, it can sometimes create inaccurate models (underfitting) and encounter difficulties when dealing with infrequent and incomplete behaviour, as the frequencies of occurrence of activities are not taken into account and the event traces are assumed to be complete, and hence the graph that represnts the process model, also. These limitations were addressed in successive studies [69, 70, 71] and, furthermore, according to [5], are counterbalanced by its important advantages, since it is highly scalable, as it can process, relatively fast, billions of activities with only 2GB of RAM memory, and also, as it is based on the "divide and conquer" paradigm, its computation can be easily distributed using Big Data solutions such as MapReduce.

1.2.2.5 Quality of Process Models

Since automated process discovery based on observed events has been identified as a primary task in this thesis, it is necessary to establish an objective method of assessing the quality and complexity or simplicity and usability of the discovered models. Among many other indicators, there are four essential and standardised metrics for process models discovered through process mining [33]:

- *Fitness*: The ability of the model to reproduce the behaviour observed in the event log.
- *Precision*: The ability of the model to not allow behaviour unrelated to what is observed in the event log, avoiding underfitting.
- *Simplicity*: The discovered model should be as simple as possible.
- *Generalisation*: The model should prevent overfitting to observed event data.

When process discovery is running using real-life event logs, most of the processes are known as *spaghetti* (see Figure 1.3). This means that they are very large and complex models, difficult to interpret at first glance, and contain duplicate activities [1]. In addition to the metrics mentioned above, in this thesis we have taken as a reference the works [34, 85, 120, 86, 28], from which we have used the following quality metrics:

- **Density** [85] is inspired by social network analysis and aims to quantify the complexity of a business model represented by event-driven process chains (EPC) [110]. This metric has a value between zero and one, where a value close to zero means that the process is not very complex and is easily understandable. In [85], it is stated that every model can be considered a graph and therefore the density is defined in a simple way as the number of existing arcs between the number of all possible arcs that could exist for a given set of nodes.

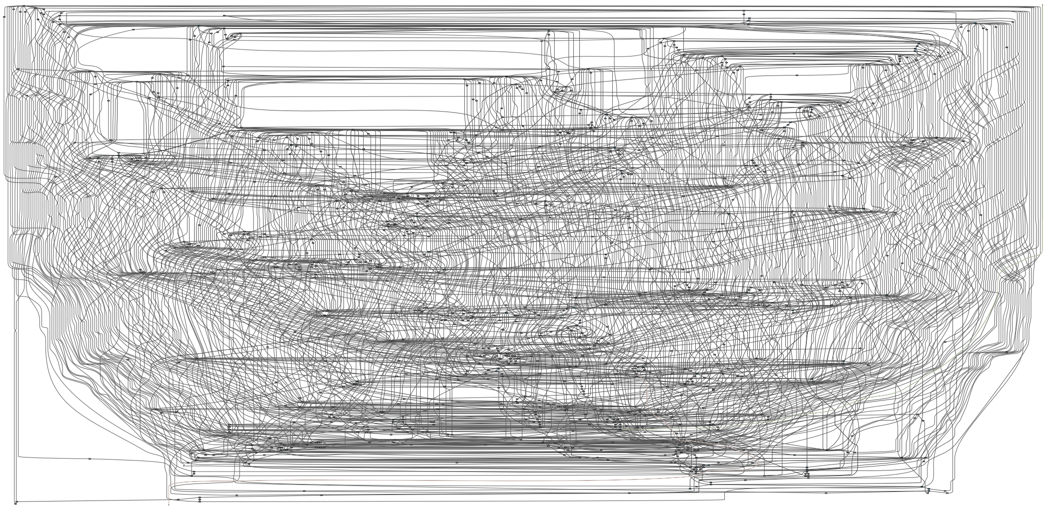


FIGURE 1.3: Example of spaghetti process

- **Cyclomatic Complexity (CC)** [68]: is inspired by [83], which defines software complexity as the number of linearly independent paths through the source code of a programme. There is no agreement in the scientific community on the definition of this metric. Cardoso [34] states that this metric cannot be defined in the context of process models, Gruhn and Laue [58] define CC as the number of binary decisions that are made minus one, and finally Latva [68] defines this metric as arcs minus nodes plus one. Thus, the lower the value of this metric, the lower the level of complexity of the model of the discovered process.
- **Coefficient of Connectivity (CNC)** [85] is proposed as a measure of the complexity of the process, taking as a reference the study by [68], who had based their proposal on [92] and who claims that the connectivity coefficient of a network and, consequently, of a process model could be measured as the ratio of transitions between activities. The higher this measure, the more complex the process model will be.
- **Control Flow Complexity (CFC)** [85] is used to evaluate the complexity introduced into a process by the presence of XOR-split, OR-split, and AND-split structures. The higher the value of CFC, the higher the overall architectural complexity of a process. In [34], it was shown that the CFC metric defined in [35] is proportional to the complexity of the control flow of the process and therefore could be used as a metric for this purpose.
- **Sequentiality (SQ)** [86] is defined as the division of the total number of arcs that connect elements that do not act as gateways by the total number of arcs present in the model. This measure, whose results can range from zero to one, means that a process model is more complex when the result is close to one.
- **Simplicity (S)** [28] is defined as the weighted average of the difference between the number of incoming and outgoing arcs calculated per node in the discovered model

compared to the same measure in the original model. The greater the differences between the mined model and the original model, the greater the complexity of the model.

Finally, it is important to note that these complexity measures help not only to estimate the degree of understanding and simplicity of processes, but also to compare different processes with each other. This allows experts to compare the different versions of process models that are discovered from the same event log and, consequently, from the same observed behaviour. And also to compare it with the process models discovered from event logs where improvements are introduced.

1.3 Objectives

This section introduces the objectives of this thesis. They are related to the enhancement of the phases related to the improvement of the event logs to be managed and measured for later process discovery processes. Thus, given the context in which this thesis is developed, the objectives traced are related to improving the state-of-the-art in relation to the challenges found in process mining in complex contexts such as logistics and manufacturing. Figure 1.4 summarises the four global objectives that we address and are defined as follows:

- **OBJ 1**. This objective is related to an overview analysis of the combination of event data and business process in complex environments, such as logistics and manufacturing contexts. The aim is to detect the main challenges that we have to face. After the analysis, some challenges were found in the literature, three of which are taken up in this thesis.
- **OBJ 2**. The aim of this objective is twofold: to improve the quality of the event data and to promote the automation of data measurement to improve data quality for later process discovery.
- **OBJ 3**. This objective is related to the definition of the abstraction level of the analysed events. The high number and types of the generated events can increase the complexity of the discovery algorithms, and, as a consequence, the discovered processes could be intractable by both humans and tools. It aims at determining the correct abstract level to be useful.
- **OBJ 4**. The last objective is related to the optimisation of business processes. In this case, we propose the optimisation of the resources associated with the process execution based on previous instances. This challenge has been analysed using manufacturing data extracted from a real example.

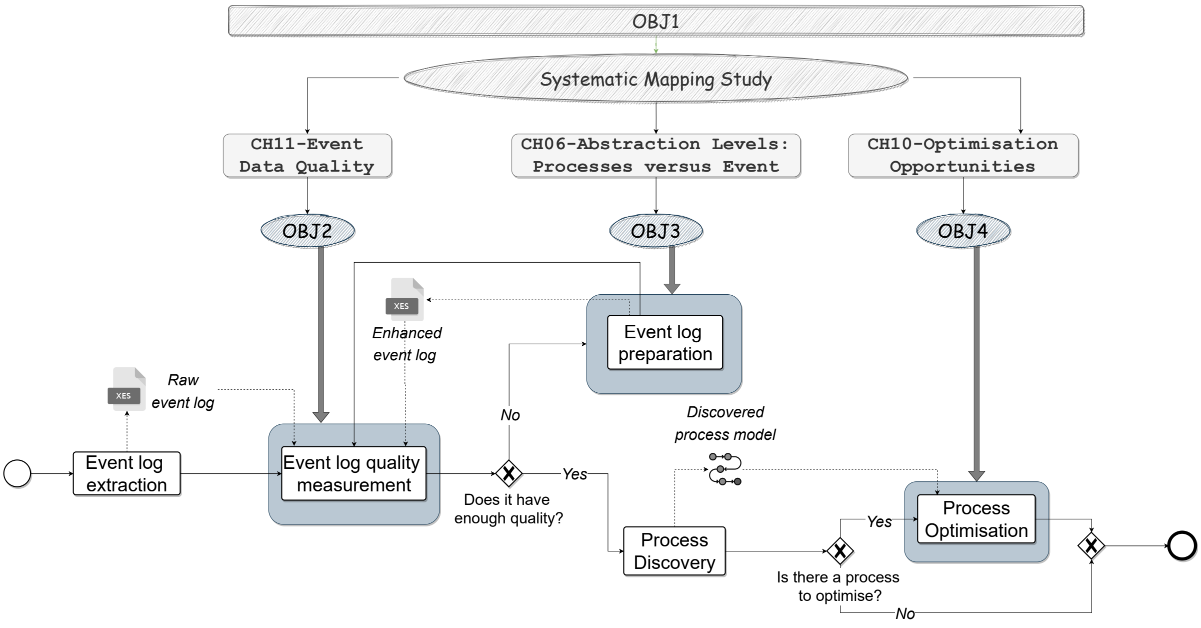


FIGURE 1.4: Thesis overview.

1.3.1 OBJ1: Overview Analysis

Manufacturing and logistics operations have grown in importance as a key component of supply chain management as a result of product exchange and market globalisation. Information and communication technologies (I&T) play a key role in helping supply chain organisations overcome these challenges. Because they can coordinate a variety of elements, including resources and time constraints, logistics processes may be extremely complex in the current globalisation scenario. Additionally, the interaction of numerous entities that may be geographically dispersed results in a large number of intricately connected events. Due to the complexity and heterogeneity of the systems and the types of events that are handled in this context, the need to analyse how business process management (BPM) can provide support has been identified. Furthermore, the particularities of the events generated in these contexts require the use of complex event processing (CEP) techniques and their combination with classical process management [19, 111, 49]. Event-Driven Business Process Management (EDBPM) [18] has therefore emerged as a new discipline to integrate the two, allowing events produced by BPM systems to be analysed concurrently by a CEP and leveraging both disciplines.

This objective examines the state-of-the-art of EDBPM in the context of logistics and manufacturing in order to understand how business processes and complex events are being integrated into these processes. To accomplish this, we conducted a systematic mapping study that groups contributions according to different categories to give a general overview of the research area. The findings of the mapping study are useful for organisations because they could use the analysis to get to know the most advanced frameworks, techniques, and technologies; furthermore, they are also useful for researchers because they highlight areas that need more investigation. This thesis addresses three of the challenges identified in the analysis, which are the basis of the rest of the objectives.

1.3.2 OBJ2: Measurement of Event Log Quality

As mentioned above, the viability of applying process mining techniques is critically dependent on the quality of the event logs. Thus, the techniques mentioned above related to the preparation of event logs must definitely be related to the quality of the event logs. For this reason, we need to empirically and objectively measure the quality of the logs since, depending on the result, different preparation and improvement techniques should be applied. As a result, the evaluation of the quality of the event log is the first and most important step in any subsequent analysis. If incomplete or inaccurate event logs are used with any process mining technique, such as process discovery, the resulting process models will also be incomplete or inaccurate. To address this challenge, it is necessary to define metrics to measure the quality of the event log, as well as a system to assess the quality level in each context.. This is known as fitness per purpose [87], where the quality level must be tailored to the needs. Examples of parameters that should be tailored are: *(i)* determining the typical number of activities per trace; *(ii)* determining the level of noise permitted; and *(iii)* determining the average length of the label of the activities.

In conclusion, the difficulty lies in determining when quality is adequate for a given purpose and what methods to employ to enhance it. In this thesis, we focus on enhancing the event logs' quality through different data pre-processing methods that impact both measurement and evaluation. We propose three elements to evaluate the quality of an event log: *(i)* a set of metrics; *(ii)* a mechanism to describe both the data and the processing rules; and *(iii)* a guide to choose the best improvement methods.

1.3.3 OBJ 3: Event Log Preparation

Process mining can be used by organisations to fully understand, enhance, and identify potential deviations from expected behaviour in their processes. Event logs, which capture the times of the events included in the traces, are used as input data for process discovery. Today, in the new IoT environments, systems tend to generate massive, distributed, heterogeneous, and complex events, resulting in chaotic combinations of elements without a structured schema [116, 73], leading to new complex scenarios because the data must first be transformed before being handled by process mining tools. The ability to extract significant events is one of the difficulties with process mining [5]. In addition, obtaining the event logs produced by the systems is becoming increasingly complex. Because current process mining solutions rely primarily on homogeneous structured data, this adds a layer of complexity that must be managed.

Even when data sources are homogeneous, significant challenges are encountered in the extraction and integration of data for event logs. References between relational tables provide a variety of combinations of traces of events that can be extracted from a relational database. Different traces can be extracted, depending on the attribute that represents the case id, the attribute that represents the execution of an activity, or the method used to obtain the timestamp to define the sequence of events. Therefore, expert knowledge is critical to determining which attributes are relevant to extracting promising business models during the discovery process. Identifying a method to quantify the potential traces that can be recovered is difficult, but it may also help the expert determine whether it is possible to identify a promising process from a potential trace extracted from a relational database.

Once the information needed to build an event log has been extracted, new challenges arise with respect to the quality of the information in these logs. Sometimes, the information in the logs needs to be pre-processed. One of the main subjects of this thesis is the event logs that contain in-depth descriptions of the execution of activities written in natural language. Because textual data are unstructured and may contain errors, process mining approaches become impractical, and this is why a pre-processing phase is required.

The final challenge faced in this objective is related to improving processes that, once discovered, turn out to be spaghetti models. It is frequently necessary to split the traces into smaller ones due to the potential complexity of both logs and discovered processes. This makes the process model easier for the user to understand and follow. In this thesis, we apply and compare various traces clustering techniques with the goal of reducing the complexity of event logs and thus the complexity of the process models discovered.

1.3.4 OBJ4: Process Optimisation

A proper process that minimises time and costs is essential for the daily performance of logistics and manufacturing organisations. This is particularly important for processes whose operations and components are expensive and involve very delicate analyses that can take a long time to complete, causing significant losses and delays. In this objective, we focus on guiding experts in optimising logistics and manufacturing processes to improve their performance. In this thesis, we provide a user-guided solution that helps users receive the information they need to improve decision-making and reduce costs.

Thus, we focus on optimisation of the process, including flexibility. Due to the analysis of real business processes in logistics and manufacturing organisations, we have identified the challenge of improving decision-making to save time and costs. This is made possible by creating flexible processes and decision tables. Since these types of decision do not follow the structure of a classical decision tree, well-known techniques in the field of business process management such as DMN [90] cannot be used, as a consequence, we need to use new techniques that allow the analysis of past events and the use of complex events to guide the execution of the process. In our proposal, optimisation is focused on the optimisation of diagnosis and isolation of incorrect components in an aircraft assembly process.

Finally, it should be made clear that the achievement of this goal usually requires methodologies that are highly dependent on the business context and domain, in which experts must have a high level of knowledge of the business and a high level of participation in the adoption of these new working techniques.

1.4 Research Methodology and Resources

This section presents the research methodology used to carry out the thesis. It begins with an overview of the methodology applied and the tasks involved, followed by details of the application of each stage of the methodology to each scope of the thesis.

1.4.1 Research Methodology

This thesis was developed using the *Design Science Research* (DSR) research methodology [93]. In essence, the goal of this methodology is to build scientific and technological knowledge so that it may be used to address the issues faced by experts in industrial domains to solve their problems. The final aim is to create artefacts that resolve a current issue. This methodology consists of six main activities that are formulated as follows [93]:

- **Activity 1: Identification of the problem**. First, a specific problem must be identified. It needs to be conceptualised and broken down into several independent sub-problems. The value that a potential solution could offer must be justified by the relevance of the problem.
- **Activity 2: Define the objectives of the solution**. The objectives of the solution must be deduced after the problem has been established.
- **Activity 3: Design and development of the solution**. This activity involves developing and designing the artefact or artefacts that will be provided as a solution to the problem. Methodologies, models, or tools can all be considered artefacts.
- **Activity 4: Demonstration**. Experiments must be conducted in realistic scenarios after the solution has been developed to determine how successfully the solution can address the problems identified.
- **Activity 5: Evaluation**. Solutions must be assessed in practical scenarios by addressing specific use cases. All of the solutions in this thesis have been carried out in actual industrial environments.
- **Activity 6: Communication of results**. It is crucial to share the results. The results obtained in this thesis have been disseminated through articles in relevant JCR journals, national and international conferences, industry forums, tool demonstrations, and book chapters in important editorials.

The rest of the section describes the process we used to identify the problem (the first activity). This thesis arises from the need to apply process mining techniques in real environments. From the beginning, it is closely connected to the transfer of knowledge and the application of academic techniques in real-life contexts. In our collaboration with experts from different domains, we observed that, in general, the more complex the processes implemented in organisations, the more problems they faced when they wanted to apply process mining techniques. To understand and improve their own processes, organisations must perform manual and tedious tasks, with a high requirement of dedication and knowledge, which usually they are not willing to assume. For this reason, we identified the need to help organisations automate some of the most costly stages of process mining, so that they could benefit from its advantages without having to expend an unbearable effort. We needed then to research which issues had already been addressed in academia or industry and which areas of process mining still required new and effective solutions. We denote this problem as **P-R**. To reach this milestone, we started by conducting a systematic literature review, which constitutes our solution **S-R**, and helped us gain an in-depth understanding of the state-of-the-art of all stages of the process mining lifecycle in manufacturing and logistics contexts, and identify areas where better solutions were still needed that could address the real problems of organisations. Due to this, we were able to select the following challenges to address.

Our previous work with data from organisations and the literature review helped us establish the following objective: how to measure the quality of the data present in the event logs. So far, we have intuited that the quality of the input data had a strong impact on the final result. However, we needed objective and empirical mechanisms to help us measure different aspects of data quality and usability. This problem has been codified as **P-DQ**. In addition, this would also help us to check if a change in the level of abstraction would produce an improvement of the discovered process model by measuring and assessing the quality of the discovered process before and after the data transformation.

Thanks to the literature review and our collaborative work with organisations, we experienced great difficulties in working with the available data and its event orientation. Thus, a previous pre-processing task was necessary to determine the correct levels of abstraction that these events should have, in order to generate better logs and, consequently, better processes. This problem has been codified as **P-AL**.

Additionally, some processes can be improved in terms of performance, costs, and resources. It is necessary to identify these processes and their points of improvement in order to propose to experts a better decision-making system, which is codified as **P-OP**.

The following sections of this thesis are organised around the three areas for improvement listed below, which are connected to the phases and tasks of process mining that we want to enhance. Each improvement area is referred to as *scope*:

- *Event log quality*: This scope includes problems related to the quality of event data **(P-DQ)**.
- *Event log preparation*: This scope covers issues related to the data preparation stage for event logs (**P-AL)**.
- *Process optimisation*: This scope groups the problems inherent in the process optimisation opportunities. **P-OP**.

1.4.2 Methodology Results

Once the problems have been identified, the rest of the activities specified in the DSR methodology have been undertaken. Figure 1.5 gives a general overview of the results of the activities in the different scopes of this thesis. The rest of the subsection explain the details of each activity focusing on the different scopes:

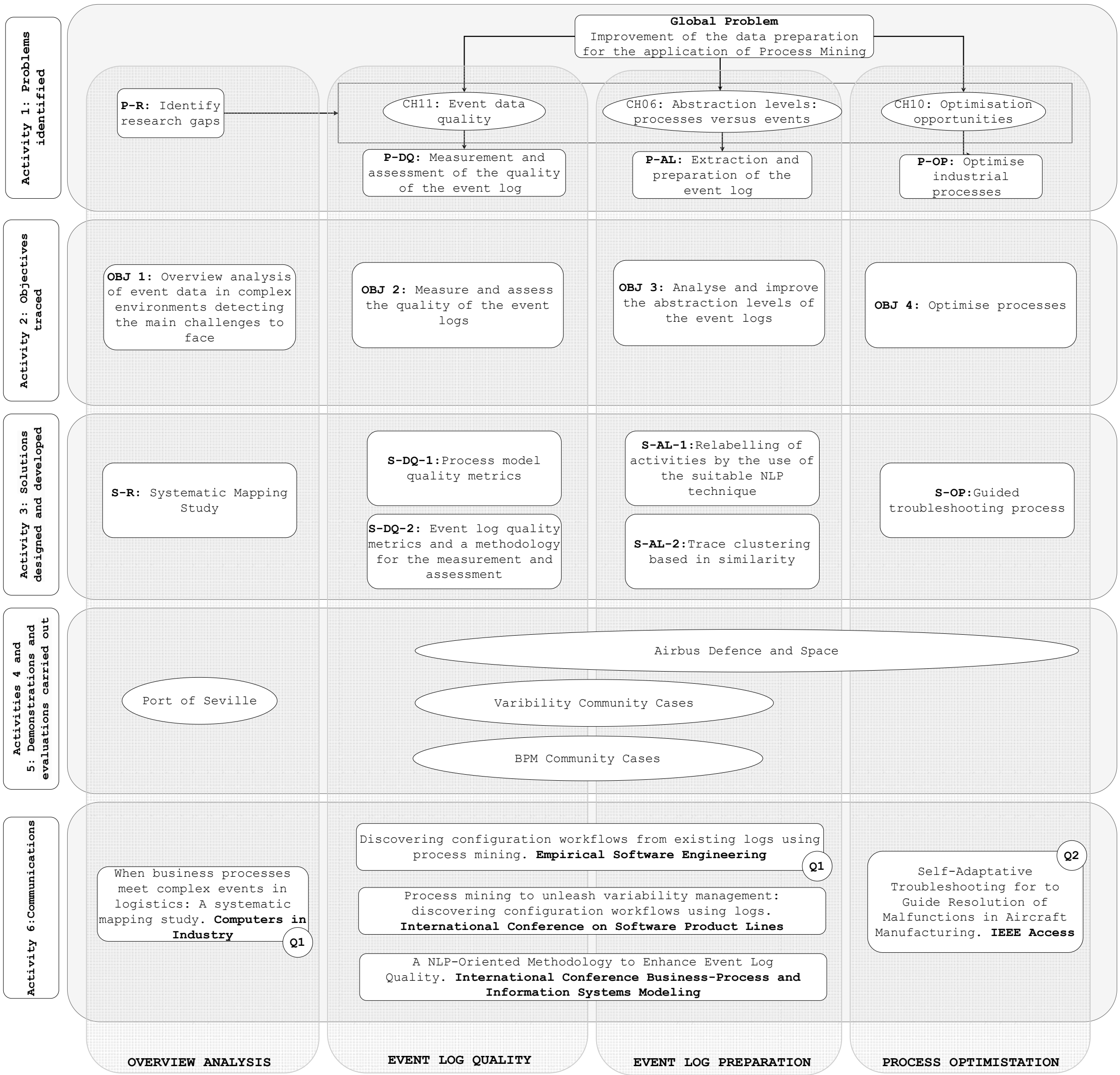

Activity 1: Problems identified
Activity 2: Objectives traced
Activity 3: Solutions designed and developed
Activities 4 and 5: Demonstrations and evaluations carried out
Activity 6:Communications
Global Problem
Improvement of the data preparation for the application of Process Mining
P-R: Identify research gaps
CH11: Event data quality
CH06: Abstraction levels: processes versus events
CH10: Optimisation opportunities
P-DQ: Measurement and assessment of the quality of the event log
P-AL: Extraction and preparation of the event log
P-OP: Optimise industrial processes
OBJ 1: Overview analysis of event data in complex environments detecting the main challenges to face
OBJ 2: Measure and assess the quality of the event logs
OBJ 3: Analyse and improve the abstraction levels of the event logs
OBJ 4: Optimise processes
S-R: Systematic Mapping Study
S-DQ-1:Process model quality metrics
S-DQ-2: Event log quality metrics and a methodology for the measurement and assessment
S-AL-1:Relabelling of activities by the use of the suitable NLP technique
S-AL-2:Trace clustering based in similarity
S-OP:Guided troubleshooting process
Airbus Defence and Space
Port of Seville
Varibility Community Cases
BPM Community Cases
Discovering configuration workflows from existing logs using process mining. Empirical Software Engineering
Q1
When business processes meet complex events in logistics: A systematic mapping study. Computers in Industry
Q1
Process mining to unleash variability management: discovering configuration workflows using logs. International Conference on Software Product Lines
A NLP-Oriented Methodology to Enhance Event Log Quality. International Conference Business-Process and Information Systems Modeling
Self-Adaptative Troubleshooting for to Guide Resolution of Malfunctions in Aircraft Manufacturing. IEEE Access
Q2
OVERVIEW ANALYSIS
EVENT LOG QUALITY
EVENT LOG PREPARATION
PROCESS OPTIMISTATION

FIGURE 1.5: Research methodology activities outputs related to the thesis scopes

1.4.2.1 Overview analysis

The activities related to this scope are represented in the first vertical lane of Figure 1.5. These activities are detailed, from top to bottom, as follows:

- **Activity 2: Objectives traced**. The objective related to the overview analysis (**OBJ 1**) is aligned with the problem **P-R**: the identification of research gaps.
- **Activity 3: Solutions developed**. The solution we have proposed to reach **OBJ 1** (**S-R**) takes the form of a systematic mapping study. This mapping study help as to scope the research in the area of research, as well as to detect open problems that led to the definition of the rest of the objectives and scopes.

- **Activities 4 and 5: Demonstrations and evaluations carried out**. The report obtained as a result of the mapping study has been used as input to detect research gaps in the context of a project in collaboration with the port of Seville.

- **Activity 6: Communications**. The results of **S-R** have been published in 2022 in the journal *Computers in Industry*, which is ranked in the first quartile of the JCR ranking.

The solution is detailed in Chapter 2 (Sections 2.2 and the publication is described in Chapter 3.

1.4.2.2 Event log quality

The activities related to this scope are represented in the second vertical lane of Figure 1.5. These activities are detailed, from top to bottom, as follows:

- **Activity 2: Objectives traced**. The objective related to measuring the quality of the event log (**OBJ 2**) aligns with the problem **P-DQ**. Therefore, the solution to these problems must include a methodology that provides mechanisms for measuring and evaluating the quality of the event logs.

- **Activity 3: Solutions developed**. The solutions we have developed to meet these objectives are: *(i)* a set of metrics to measure the quality of the process models (**S-DQ-1**) and *(ii)* a set of metrics to assess the quality of the processes discovered before and after the processing of the event log, to assess the impact of log quality on the quality and usability of the obtained process models (**S-DQ-2)**.

- **Activities 4 and 5: Demonstrations and evaluations carried out**. **S-DQ-1** and **S-DQ-2** solutions have been evaluated thanks to a real case study of an aircraft assembly line and by means of different datasets provided by the BPM community to be used with evaluation purposes.

- **Activity 6: Communications**. Since the results presented in this scope fall within a subsystem in the solutions presented in the event log improvement one, the communications presented match. For **S-DQ-1** a first solution was published at the international conference *Systems and Software Product Lines* in 2019 with a GGS SCIE Class 2 rank, which was subsequently extended in a publication in the journal *Empirical Software Engineering* that is ranked in the first quartile (Q1) of the JCR ranking. The solution **S-DQ-2** was submitted to *Business Process Modeling Development and Support*. In connection with this submission, a chapter was published in 2021 in the book *Enterprise, Business-Process and Information Systems Modeling*.

The two aforementioned solutions are detailed in Section 2.3 of Chapter 2. The publications obtained as a result of this scope are introduced in Chapter 3.

1.4.2.3 Preparation of event log

The activities related to this scope are represented in the third vertical lane of Figure 1.5. These activities are detailed, from top to bottom, as follows:

- **Activity 2: Objectives traced**. The objective (**OBJ 3**) is aligned with the problem group **P-AL**. Therefore, the solution to solve these problems should include a methodology that helps experts to introduce modifications in the event log to improve the level of abstraction.
- **Activity 3: Solutions developed**. We arrive at different solutions to solve the set of problems **P-AL**. The solutions are: (**S-AL-1**) the use of natural language processing techniques to enrich the logs and improve the event labels composed of textual information, and (**S-AL-2**) a framework capable of applying different clustering techniques to an event log, allowing one to simplify the traces and obtain more manageable process models when the originals turn out to be spaghetti.
- **Activities 4 and 5: Demonstrations and evaluations carried out**. Both **S-AL-1** and **S-AL-2** were assessed using two different case studies. **S-AL-1** with real data from an avionics company assembly line and **S-AL-2** with real data from different datasets available in the literature to test the proposed solutions.
- **Activity 6: Communications**. Finally, the results obtained were presented at two conferences, a book chapter, and a journal. **S-AL-1** was presented at *Business Process Modeling Development and Support*. In connection with this submission, a chapter was published in 2021 in the book *Enterprise, Business-Process and Information Systems Modeling*. In relation to **S-AL-2** a first solution was published at the international conference *Systems and Software Product Lines* in 2019 with a GGS SCIE Class 2 rank, which was subsequently extended in a publication that is ranked in the first quartile (Q1) of the JCR ranking in journal *Empirical Software Engineering*.

The two aforementioned solutions are presented in Chapter 2 (Section 2.4). The publications derived from this scope are discussed in Chapter 3.

1.4.2.4 Process optimisation

The activities related to this scope are represented in the fourth vertical lane of Figure 1.5. These activities are detailed, from top to bottom, as follows:

- **Activity 2: Objectives traced**. The objective related to the process optimisation activity (**OBJ 4**) is aligned with the problem **P-OP**. Therefore, solutions to this problem must consist of a methodology capable of guiding the expert in process improvement and better decision-making.
- **Activity 3: Solutions developed**. We have developed an **S-OP** solution based on flexible processes to guide the user through industrial diagnosis and troubleshooting processes to make better decisions and save costs based on previous events.

- **Activities 4 and 5: Demonstrations and evaluations carried out**. The solution (**S-OP**) has been tested on a real aircraft assembly line in conjunction with in-house domain experts. This solution has been implemented in production and is validated by users.
- **Activity 6: Communications**. **S-OP** was published in journal *IEEE Access*, that is ranked in the second quartile (Q2) of the JCR ranking.

The solution is presented in Chapter 2 (Section 2.5). The results that have been produced with respect to objective OBJ 4 are discussed in Chapter 3.

1.4.3 Resources

In terms of resources, the following were used in the development of this research project:

- **Scientific documentation.** The analysis of the literature has been used to conduct research. A variety of scientific databases are accessible through the University of Seville.
- **Cloud servers.** A private cloud is available thanks to the IDEA Research Group. Experiments have been conducted using this cloud.
- **Process mining tools** such as PROM, Disco, and PM4PY.
- **Database management systems** such as Oracle and Neo4j.
- **Additional applications to create documents** (Latex through Overleaf, Microsoft Word), presentations (such as draw.io, Microsoft Visio, Astah), diagrams (Google Presentations, Microsoft PowerPoint) and tools for literature reviews (Start).

Each software package that has been used is open-source or proprietary. Through licences offered by the University of Seville, proprietary software has been used.

1.5 Roadmap

The remainder of this thesis is organised as follows. The results presented in Chapter 2 are a summary of the results obtained for each objective defined in Section 1.3. The results are discussed and the publications that form and support this thesis are presented in Chapter 3. Finally, Chapter 4 discusses future work and draws conclusions from this thesis.

Chapter 2

Summary of the Results

This chapter summarises the results of this thesis, introducing the solutions that have been developed to meet the defined objectives. Section 2.1 provides an overview of the findings. Subsequently, Section 2.2 introduces the results of the state-of-the-art analysis we did to reveal research gaps. After that, the results related to the quality of the event log, the preparation of the event log, and the optimisation stages of the process are introduced in Sections 2.3, 2.4, and 2.5, respectively. Finally, Section 2.6 sets out brief conclusions of the results obtained.

2.1 Introduction to the results

As mentioned in Chapter 1, the objectives of this thesis can be grouped into four large blocks: Overview Analysis, Event Log Preparation, Event Log Quality, and Process Optimisation. The following sections introduce the results of each one. First, in Section 2.2 we show the results of the state-of-the-art analysis. Second, in Section 2.3 the results related to the Event Log Quality assessment activity are presented. There, we describe a set of metrics to assess the quality of the processes discovered before and after the log processing stage, and a set of metrics to assess the quality and fitness-per-purpose of the models obtained and the impact of the processing activities in the final models. Third, in Section 2.4, the results related to the Event Log Improvement stage are presented: an NLP-based relabelling framework and a clustering framework to enrich and simplify event logs, respectively. Finally, in Section 2.5, we present the results related to the Process Optimisation activity, a methodology capable of guiding the expert in process improvement and better decision-making.

2.2 OBJ 1: Overview Analysis

This section describes the solution related to the objective **OBJ 1**. To accomplish this objective, an exhaustive, systematic analysis of the state-of-the-art was carried out. The analysis explores the combination of business processes and complex event processing in complex systems, such as logistics or manufacturing. The objective is to have an overview of how they are integrated into these contexts.

The two disciplines that traditionally address issues related to business processes and complex events separately are business process management (BPM) and complex event processing (CEP). However, Event-Driven Business Process Management (EDBPM) [18] has recently emerged as a new discipline for integrating the two. Using both disciplines, EDBPM can analyse events produced by BPM systems in parallel.

The purpose of our study [100] is to study the current state of EDBPM to integrate business processes and complex events into the logistics context. To do this, we conducted a systematic mapping study with the purpose of classifying, counting, and presenting a broad overview of the research area. Our mapping study collects and synthesises the methods, frameworks, and tools that combine these two disciplines. As a result, 10,978 studies were collected, with 169 of them chosen for extraction. We classified the studies into different categories using a list of criteria, such as the stage of the business process life cycle in which they were used, the business process modelling language, and the event process modelling language, among others. Our analysis highlights the open problems and the most relevant frameworks and tools. The findings of this mapping study are useful for research because they identify the gaps that can drive future research lines.

We follow the guidelines proposed by Kitchenham et al. [66, 65] and Kuhrmann et al. [67] to conduct systematic literature reviews in software engineering. And Wohlin's guidelines for the use of snowballing [131].

Following this methodology, the first step was to define *research questions*, as they drive the whole review process. The research questions that were addressed in this study are summarised in Table 2.1. We classified the research questions into three groups: demographics, current trends, and research gaps. In doing so, our objective was to obtain a holistic view of the state of research in this area, with a special emphasis on identifying areas where its applicability is not mature enough to be used in real and complex scenarios.

Id	**Research question**
RQ1	What are the demographics of the published studies?
	RQ1.1 Which contributions were made over the years?
	RQ1.2 Which are the most influential researchers in the area and were are they from?
	RQ1.3 Which are the most influential studies?
	RQ1.4 Which are the top venues?
RQ2	What are the current trends in the area?
	RQ2.1 What is the type of contribution made by the study?
	RQ2.2 Which are the application areas of the studies?
	RQ2.3 What type and which business process modelling languages are utilised?
	RQ2.4 What type and which event processing languages are utilised?
	RQ2.5 How is logistics covered in the studies?
	RQ2.6 Which are the event producers?
RQ3	What are the potential gaps in the area?
	RQ3.1 Is event-driven business process treated in every phase of the process life cycle in a logistics context?
	RQ3.2 What are the challenges taken off in the field?

TABLE 2.1: Research questions.

This problem could have been approached from different points of view, but it was crucial for this thesis to analyse it from the point of view of each of the stages of the life cycle of the business processes. Similarly, given the interest in knowing to what extent these techniques, so popular in academia, have been applied in industry. Similarly, it was essential to know what kind of solution is proposed for the different problems that arise on a daily basis in these complex contexts. In other words, identify which of the challenges in the area have been addressed in a practical way.

To determine the potential gaps in the area, we took the categorisation of challenges provided as an outcome of the Dagstuhl Seminar 16341 on "Integrating Process-Oriented and Event-Based Systems" [51] (see Table 2.2).

Id	Challenge category description
CH01	**Event Models for BPM**: This group of challenges is related to how events can be used to process instance adaptation, how the change of event states can influence process instances and how processes can help to give context to events.
CH02	**Compliance, Audit, Privacy and Security**: The challenges in this group are related to the exploitation of CEP to processing audit logs, and BPM tools to express policies in event-based systems and take benefit of the richer access control of BPM.
CH03	**Automatic Event-Based Monitoring of Processes**: This group of challenges are related to the automatic discovery of event patterns to business process monitoring, to monitoring events to guide process adaptation and to use process information to guide the monitoring adaptation.
CH04	**Patterns and Models for Communication**: This group of challenges are related to how the the effects of the communication model impose by event-based middlewares are explicitly reflected in process models.
CH05	**Choreographies and Inter-Process Correlation**: This group of challenges is related to the extension of choreography languages to deal with advanced event-based concepts, and to enable the analysis of the information flow between processes.
CH06	**Abstraction Levels. Processes versus Events**: As process models usually follow a top-down approach, whereas event processing follow a bottom-up approach, the challenges in this group are related to find the adequate level of abstraction for a concrete modelling goal and to deal with conflicts in large-scale systems integration.
CH07	**Context in Events and Processes**: The challenges in this group are related to the representation of context, both in processes and in event patterns, to the scoping of context and to the relation of processes and context at runtime.
CH08	**Integrated Platforms for BPM & CEP**: The challenges in this group are related to the integration of BPM and CEP platforms, which also involve the development of a unified model for events and processes.
CH09	**Distributed Processes & The Role of Events**: The challenges in this group are related to event loss, to misdetection of complex events, to analysis of stream events in real time, to deal with privacy in the context of event and processes handled in a centralised or distributed sources.
CH10	**Optimisation opportunities**: The challenges in this group are related to the exploitation of BPM to improve event processing, and the other way around, the exploitation of CEP methods to improve processes.
CH11	**Event Data Quality**: The challenges in this group are related to making explicit uncertainties, to make business models aware of data quality and to how this quality-aware model influences decision making.
CH12	**From Event Streams to Process Models and Back**: The challenges in this group are related to automate the generation of CEP rules from business process monitoring, to use CEP constructs for process mining, to enrich the expressiveness of process models with CEP constructs and to execute business processes via CEP rules.

TABLE 2.2: Challenges categories.

Our analysis revealed that the challenges of the `CH08-Integrated Platforms for BMP & CEP` group are by far the most successfully completed, the `CH03-Automatic Event-based Monitoring of Processes` group is ranked second overall and the group `CH01-Event Models for BPM` is in third place. The challenges that are at the bottom of the ranking are those that belong to the groups `CH11-Event Data Quality`, `CH06-Abstraction Levels: Processes versus Event`, `CH10-Optimisation Opportunities`, and `CH02-Compliance, Audit, Privacy _- and Security .`

As a result of the analysis of the least beaten off challenges, we conclude that, in relation to the `CH11` challenge, the enrichment of process models with the specification of event quality has not been addressed in any proposal. Furthermore, very few proposals focus on defining sources of uncertainty, evaluating the quality of events, and translating them into a process-oriented specification. Only one proposal focusses on improving decision-making based on a quality-aware process. Regarding the challenges related to `CH06`, none of the analysis proposals addresses the issue of dealing with unexpected events, and only one of them refers to dealing with integration issues. Event logs, which are used as input for process discovery algorithms in almost all studies, are used to establish the relationship between events and process models, but very little attention is paid to comparing the abstraction levels of processes and events. Regarding `CH10`, it should be noted that most studies focused on the use of CEP to detect, predict, and improve processes at runtime in relation to the challenges related to `CH10`. The allocation of resources for event-driven architectures has been the subject of only two studies. However, three issues remain unresolved in the identified studies: how data from processes can be used to optimise complex event processing; distributed event stream query to improve BPM performance; and having a language to handle BPM and CEP that is expressive enough to deal with users (from the perspective of business processes), while also being effective for evaluation (from the point of view of CEP). Finally, only one of the five problems listed in `CH02` has been addressed: information access control and automation of compliance validation. As a result, issues such as audit log processing, application of Service Leve Agreement (SLA) to CEP engines, and use of process models to express policies in event-based systems remain unresolved.

The objectives of this thesis are closely related to the challenges that are less addressed. according to the analysis in OBJ 1.

- **OBJ 2**: `CH06-Abstraction Levels:Processes versus Event.`
- **OBJ 3**: `CH11-Event Data Quality`
- **OBJ 4**: `CH10-Optimisation Opportunities`

Furthermore, as a result of analysing how these challenges are addressed in the different phases of a process lifecycle. We discovered that `CH06` was never addressed during the monitoring phase. All challenges are addressed during the process analysis phase, but `CH10` and `CH11` are not addressed during the process implementation phase. The absence of `CH02`, `CH06`, `CH09`, and `CH11` from the redesign phase indicates that more research is required on topics such as compliance, levels of abstraction, distributed processes, and event quality. And finally, there are still many difficulties in the phases of identification and discovery (`CH04`, `CH06`, `CH07`, `CH09`, `CH10`, and `CH11`). As a result of these findings, the focus of this is on process identification and process discovery, two of the areas where we found gaps to cover.

2.3 OBJ2: Event Log Quality

This section addresses the proposed solutions to achieve the objective **OBJ2**. Here, solutions related to the assessment the quality of events of the logs that are used as input in process discovery tasks are introduced for a later modification of the event log quality and process mining techniques.

The viability of applying later process mining techniques is critically dependent on the quality of the event logs. In event logs, a wide range of information related to events can be stored, such as textual descriptions, timestamps, or used resources [5, 7]. Event logs are the footprints left by an organisation's information systems. Typically, event logs must be modified for later analysis (process mining), such as process discovery. Therefore, the first and most important step in any subsequent analysis is to evaluate the quality of an event log [15]. Any process mining technique, such as process discovery, applied to incomplete or inaccurate event logs will result in incomplete or inaccurate process models [132]. Preprocessing is required in this situation since process mining techniques are impractical in complex environments like logistics and manufacturing where information is unstructured and may contain errors. As a result, we propose a methodology that allows a customised description of measurement and evaluation of event log quality based on expert requirements.

Criteria for evaluating data quality have generally been established by several authors [91, 24], and also for event log quality, in particular [15, 132], including completeness, correctness, security, and trustworthiness. The Process Mining manifesto [15] defines event log quality as a level of quality maturity ranging from low (i.e., 1 star) to high (i.e., 5 stars). These maturity levels assign the quality *lowest* when the recorded events do not correspond to reality, for example, when they are recorded manually. The level of quality is *medium* when events are automatically extracted, but full coverage in their recovery cannot be guaranteed. While *high* quality refers to an automatic and complete recovery, the highest quality is achieved when each recorded event and each of its associated attributes have a known semantic meaning in relation to the process model. By examining activity labelling, timestamps, and case identification, among other things, imperfections that lead to poor event log quality can be reduced.

To address this challenge, we propose the solution **S-DQ-2**, which not only adapts a general methodology to measure and assess the quality of event log data [117], but also proposes a decision support system to help in the choice of the most appropriate techniques to be applied to event logs based on the current level of quality and expected assessment. To do this, it is necessary to define metrics for measuring event log quality as well as a method for determining how effective this level of quality is in various scenarios. This is called fitness-per-purpose [87] and requires that the level of quality be adjusted to meet needs, such as: *(i)* estimating the average length of the activity labels; *(ii)* the permissible noise level; and *(iii)* the common number of activities for each trace.

To accomplish this objective, we propose a set of metrics and dimensions that can be customised or extended. The most effective event log preparation methods to employ will depend on the definition of event log quality in each context, the quality of the event log prior to applying the methods, and the direction in which the experts want it to change following the method's application. The first step to determine when the event log is can be used, according to the measurement and assessment described by a set of decision rules about data quality (cf. Determine the usability of the event log quality). If the event log is of sufficient quality, a process mining analysis can be performed on it. However, the expert must identify the dimension or dimensions (i.e., accuracy, completeness, consistency, and uniqueness) that need to be adjusted, as well as the necessary assessment if the event log is deemed unsuitable (cf., Introduce dimensions and assessment to achieve). Using this knowledge, we suggest a decision-support system that offers various preprocessing techniques to enhance event log quality according to the decision rules discussed on data quality and the needs of experts. Until the resulting event log reaches the necessary quality level, the process assessment and improvement can be repeated.

The question of when an event log is of sufficient quality has no straightforward answer. The meaning of the term "event log quality" will vary depending on the situation. There are solutions that offer mechanisms for describing decision rules for measurements and assessments that are specific to each situation and need [117]. Using a set of metrics taken from the event log, we must first define decision-making guidelines based on the quality of the data. Consequently, it is necessary to define a set of metrics to assess the dimensions.

As stated in [20], certain metrics must be computed from the event log [2, 37] in order to measure the dimensions. In our case, the metrics used are as follows:

- **Number of traces**. The total number of traces included in the log.
- **Number of events**. Total number of events included in the log. Determines the size of the log.
- **Number of different labels**. Number of different labels that occur in every trace.
- **Number of unique labels**. Number of single (unique) labels that appear in the log.

Accuracy, completeness, consistency, and uniqueness are general terms used to describe the relevant characteristics of a data set in the context of the dimensions of data quality. However, we cannot guarantee that an event log with a high level of quality in those areas will lead to helpful business process models. This is why, as stated in [123], other dimensions are added to assess the quality of the event logs in the process mining. The use of NLP methods can have an impact on these dimensions. Using them as a foundation, we suggest the following dimensions, although our methodology can also be used with others:

$$m_{Uniqueness} = \left(\frac{Number\ of\ Unique\ Labels}{Number\ of\ Events}\right) \quad m_{Complexity} = \left(\frac{Avg.\ Number\ of\ Events}{Number\ of\ Traces}\right)$$

$$m_{Relevancy} = \left(\frac{Number\ of\ Different\ Labels}{Number\ of\ Events}\right) \quad m_{Consistency} = \sum_{i=1}^{\varepsilon} \left(\frac{|l(\varepsilon_i) - \overline{l(\varepsilon)}|}{Number\ of\ Events}\right)$$

As stated above, data quality is a factor closely related to how the data will be used in the future, so it must be tailored to the needs. The DMN (Decision Model and Notation) [90] can be used to simplify the description of data quality divided into measurement and assessment rules by following the procedure suggested in [117]. OMG proposed DMN as a declarative language for describing decision rules that are applied to a tuple of input data to produce a tuple of outputs based on the evaluation of a set of conditions described in FEEL (Friendly Enough Expression Language), which was also presented in the same OMG specification. A DMN table is composed of rows that describe a decision rule as an if-then condition, with the output returned if the condition is met. Moreover, DMN implements a hierarchical structure in which the output of one DMN table can be the input of another. We propose dividing measurement and assessment into two levels for each dimension involved using the DMN4DQ methodology. Furthermore, the final assessment is obtained by aggregating the evaluations for each dimension, as shown in Figure 2.1. We should keep in mind that the result of the measurement does not indicate whether the metric is good or bad; therefore, a subsequent evaluation is required. This assessment, as previously stated, has been defined based on the experts' knowledge of the event logs, but other assessments can be accommodated.

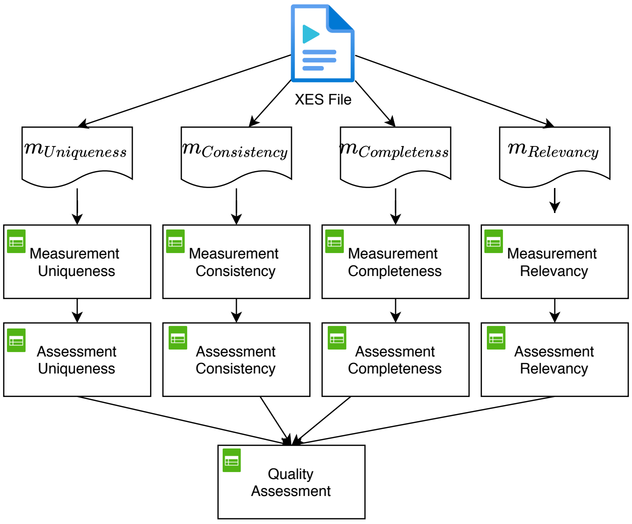


FIGURE 2.1: DMN for describing the Quality Assessment.

Once the quality of the logs has been assessed, and the processes discovered, it is necessary to analyse the impact these improvements have had on the resulting process model. Here is where our **S-DQ-1** solution is located. Once the process models had been obtained, we had to integrate an analysis of this chapter and we had to integrate an analysis of these models that would allow us to obtain metrics that would provide an objective result to compare results in terms of complexity and quality. These metrics will determine whether the models obtained can be further analysed or whether they need to be further improved. By having information on total model activities and transitions and outgoing transitions from the gateways, it is possible to obtain important values that allow us to estimate the quality and usability of the models. Our proposal focusses on those that provide a better perspective on the models' capacity to be understood, and therefore all those metrics whose results allow us to compare the different models generated when we aim to improve the information they contain. For this purpose, we have used the metrics defined in Section 1.2.2.5.

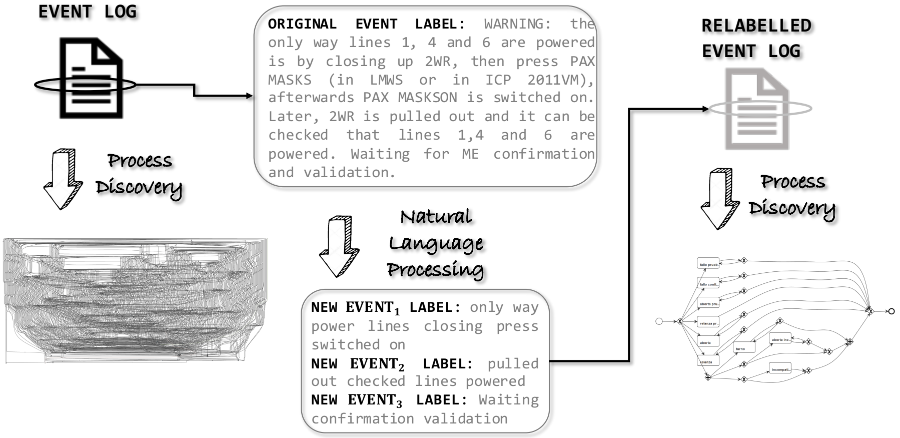
In conclusion, thanks to the application of the DMN4DQ methodology in event log assessment (**S-DQ-1** and **S-DQ-2**), we covered objective 2, where the aim was to automatically measure and evaluate the quality of the data.

2.4 OBJ 3: Event Log Preparation

This section focusses on the solutions related to **OBJ3**, that is, focused on the modification of the abstraction level of the event logs, according to the assessments obtained in OBJ2. Section 2.4.1 introduces the solution **S-AL-1**, which consists of LOADING-NLP, a methodology devised to improve certain types of event log, those that contain textual data written in natural language. LOADING-NLP applies different Natural Language Processing (NLP) techniques to raw event logs. Finally, Section 2.4.2 introduces the solution **S-AL-2**, the COLOSSI framework. The goal of COLOSSI is to abstract from logs with excessive detail level in the context of a configurator, a software tool that guides users through the configuration process of other software tools.

2.4.1 The LOADING-NLP methodology

The solution **S-AL-1** focusses on a specific kind of event logs, those that contain textual data written in natural language and detailed descriptions of the execution of activities. As textual data are unstructured and may contain errors, we need a pre-processing step. Our contribution in this matter is LOADING-NLP, a methodology that relies on the application of Natural Language Processing (NLP) techniques to raw event logs in order to relabel activities. The assessment of the event log quality, as well as the description of the measurement, depends on the context and, as a consequence, they could be customised by the expert. Our methodology also helps experts select the best NLP technique according to the characteristics of the event log. To evaluate the methodology, a real-world event log was used. The log holds event data that are stored to handle incidents that may arise during aircraft assembly in the aerospace manufacturing industry. Analysing activity labelling, timestamps, case identification, and other imperfections that can lead to low event log quality may help improve them. In the first step of LOADING-NLP, NLP techniques are used to extract the activities and relationships between them from textual descriptions. The activities are then given new labels in the event log. Figure 2.2 shows how the result of applying process mining techniques to the raw event log is a "spaghetti" process model, whereas if we relabel the event log and apply process mining techniques to relabel the event log, the result is a process model much more understandable by human users.



ORIGINAL EVENT LABEL: WARNING: the only way lines 1, 4 and 6 are powered is by closing up 2WR, then press PAX MASKS (in LMWS or in ICP 2011VM), afterwards PAX MASKSON is switched on. Later, 2WR is pulled out and it can be checked that lines 1,4 and 6 are powered. Waiting for ME confirmation and validation.

NEW EVENT$_1$ LABEL: only way power lines closing press switched on
NEW EVENT$_2$ LABEL: pulled out checked lines powered
NEW EVENT$_3$ LABEL: Waiting confirmation validation

FIGURE 2.2: Example of application of solution **S-AL-1**.

The best NLP techniques to obtain useful results depend on what is defined as quality in the application context, which is the quality of the event log before applying the NLP technique, and of course, on the expert's criteria. Figure 2.3 shows the process we propose in LOADING-NLP to support this solution. Specifically, **S-AL-1** is represented in activities *"Infer NLP technique to apply"* and *"Apply NLP techniques for relabelling"*

For selecting the NLP technique, LOADING-NLP defines a set of dimensions of event quality, selects a set of NLP techniques, and analyses the impact of the techniques on each of

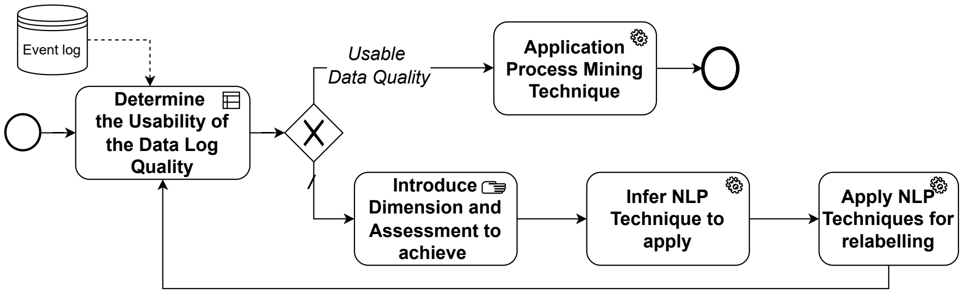

Event log
Determine the Usability of the Data Log Quality
Usable Data Quality
Application Process Mining Technique
Introduce Dimension and Assessment to achieve
Infer NLP Technique to apply
Apply NLP Techniques for relabelling

FIGURE 2.3: Methodology for the application of NLP techniques.

the dimensions. The results obtained can be seen in Table 2.3, where columns represent the different NLP techniques that can be applied (Sentence Detection, POS Tagging, Lemmatisation, Depedency Parsing, and Acronym Detection), and rows represent the different dimensions of the quality of logs. An arrow in a cell means that the dimension could be reduced (↓), increased (↑) or is unaffected (). As can be seen, all dimensions can be reduced, with the exception of $m_{Complexity}$, which could be increased. Furthermore, due to the often short length of acronyms, it can increase $m_{Uniqueness}$ and $m_{Relevancy}$ of acronym detection, while having little to no impact on $m_{Complexity}$ and $m_{Consistency}$.

Dimensions	Sentence Detection	POS Tagging	Lemmat.	Dependency Parsing	Acronym Detection
$m_{Uniqueness}$	↓	↓	↓	↓	↑
$m_{Consistency}$	↓	↓	↓	↓	-
$m_{Relevancy}$	↓	↓	↓	↓	↑
$m_{Complexity}$	↑	-	-	-	-

TABLE 2.3: Expected impact of NLP techniques on the log quality dimensions.

In conclusion, relabelling event logs using NLP techniques has proven to be a useful tool that guides the selection of the most appropriate set of techniques to be used to achieve a particular assessment of quality. Figure 2.4 shows an example of the application of our approach. The step-by-step proposed method to improve the evaluation of the dimension *Consistency* for an event log is explained. To do so, it is necessary to improve the measurement of the dimension and, consequently, to apply NLP techniques that cause this measurement to be minimised. The method is described as a business process model with four sequential activities: determine the usability of the data log quality, introduce dimensions and assessment to achieve, infer NLP techniques to apply and apply NLP technique for relabelling. Thanks to this proposal, it is possible to customise the granularity of the events through the application of NLP algorithms. More details can be found in [97].

2.4.2 Event log clustering: The COLOSSI framework

Another way to manage the abstract level of an event log for a later discovery process is related to the creation of clusters with more similar events. Grouping the events facilitates the application of elimination of noise and low frequency patterns of events, customising the abstract level of the event log according to the requirements. COLOSSI is the framework we propose for the **S-AL-2** solution. COLOSSI follows an approach based on trace clustering, whose

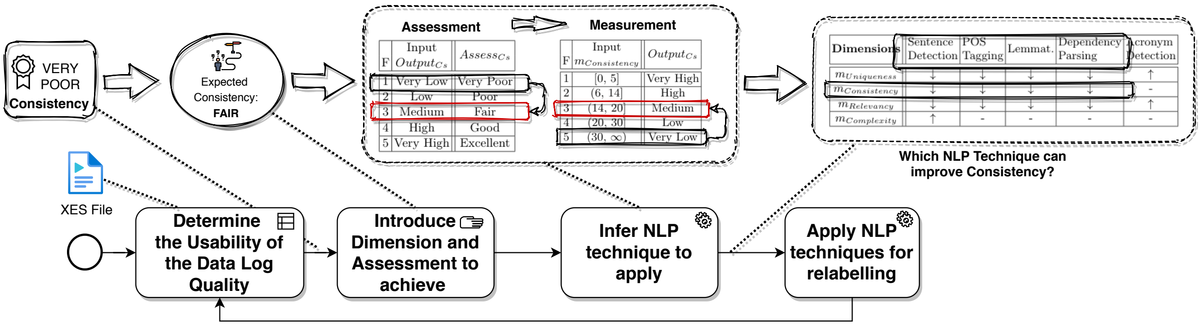


FIGURE 2.4: Inferring the NLP technique to apply.

goal is to simplify the complex process models obtained by mining observed behaviour coming from real cases. To address this problem, COLOSSI divides the traces (process instances) contained in the event log into subsets to obtain different views of the process. COLOSSI was presented in [121] and [98]. The rationale for dividing the traces into clusters is to group those traces that are most similar to each other, which means that the sequence of activities executed is also similar. Note that this similarity implies that less control structures are expected to be discovered, which complicates the comprehensibility of the model. COLOSSI constructs clusters using distance matrices. To construct these distance matrices, we have defined different entropy metrics [121], in such a way that COLOSSI constructs similarity matrices between each pair of traces (the previously called distance matrices). Thus, the best quality clusters (logs) will be those that minimise the sum of the entropies of the whole cluster (log). To segment the traces while minimising entropy in each cluster, different clustering techniques were used and evaluated. COLOSSI has been applied in an environment of variability management in configurators; Figure 2.5 shows the adopted solution, where activity *"Configuration log cluster generator"* constitutes the solution **S-AL-2**. The proposal for the simplification of the event logs can be seen in general terms. Starting from event logs that have generated complex process models, a set of clustering techniques is applied to group the traces according to their similarity and obtain a new subset of event logs that represent the same process from a different, more simplified, and manageable perspective.

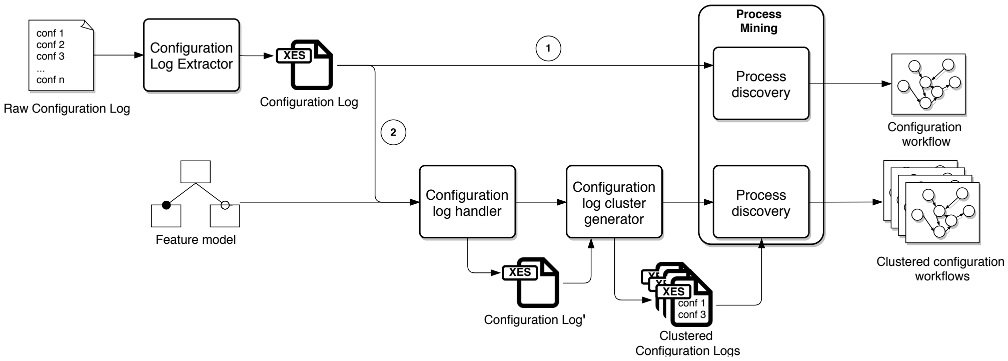


FIGURE 2.5: **S-AL-2** overview.

There are different techniques in the literature to find the best clustering configuration, such as those that use exhaustive techniques [64], the k-means algorithm [114], the hierarchical algorithm [52, 79, 78] or the greedy algorithm [57, 124]. COLOSSI offers the infrastructure for integrating different algorithms to cluster the traces involved in a configuration workflow because the best method to use will depend on the event log. Depending on the situation and the requirements, the expert can choose a specific technique. Due to the size and complexity of the generated logs, it is challenging to create clusters that aid in understanding the processes that are discovered. As a result, we suggest four approximations based on various techniques to assist in the development of clusters, as well as make an effort to enhance the distribution of logs later used as input for process discovery. The distribution of traces among clusters is determined by the practitioners' goals. In our case, the goal is to group the configuration traces that are most similar. The COLOSSI interpretation of "similar" is related to both events and transitions (any edge in the workflow that results from an activity) in the logs. As a consequence, we modified the classic information entropy metric [77] by introducing two distinct custom entropy metrics for the trace clustering context.

- *Entropy-activity ($S_{activities}$)* of a cluster: a metric which measures the similarity between traces according to the activities belonging to the same cluster. In other words, it is the ratio between the number of activities that do not appear in all traces ($activities_{nat}$) and the number of different activities in all traces ($activities_{diff}$):

$$S_{activities} = \frac{|activities_{nat}|}{|activities_{diff}|} \tag{2.1}$$

- *Entropy-transitions ($S_{transitions}$)* of a cluster: a metric that measures similarity between traces according to the transitions belonging to the same cluster. In other words, it is the ratio between the transitions that do not appear in all traces ($transitions_{nat}$) and the number of different transitions in all traces ($transitions_{diff}$):

$$S_{transitions} = \frac{|transitions_{nat}|}{|transitions_{diff}|} \tag{2.2}$$

It should be noted that the entropy has a range of [0..1]. Entropy values close to zero represent more similar traces in terms of activities executed, whereas values close to one indicate that different activities are involved in the cluster's traces. A cluster configuration that divides a set of traces into as many clusters as the optimal value of *k-cluster* indicates and, in turn, minimises the total cluster entropy is the best cluster configuration. The difficult part is determining the ideal cluster setup for preprocessing process discovery. All clustering algorithms should ideally aim to cluster data as uniformly as possible. This means that the distances between elements in the same cluster must be as small as possible, while the distances between elements in different clusters must be as large as possible. Clustering algorithms typically generate distance matrices during their execution to carry out this operation. These distance matrices are calculated on the basis of various criteria that determine which elements should and should not be grouped. But since our approach relies on minimising the sum of entropies, the clustering algorithm must take this into account when grouping data. For this reason, the entropy matrix is created beforehand and is just a square matrix that contains the entropy value between each pair of traces, as described in the aforementioned definition of entropy. This is equivalent to the distance between each pair of traces, based on our criteria. As a result, when necessary, our clustering procedure will use our entropy matrix as a distance matrix. For trace clustering, we propose using a greedy algorithm, a backtracking algorithm that is exhaustive, and two approximation algorithms (genetic and hierarchical agglomerative).

Our experimentation confirms that our clustering-based mechanism reduces the complexity of event logs and facilitates the understanding of subsequently discovered process models [121, 98]. This allowed us to identify some relevant behaviours in the proposed solutions that adapt to the level of granularity depending on the clustering. Hierarchical clustering typically yields positive outcomes in terms of effectiveness, speed, and efficiency. An important bottleneck is the creation of the entropy matrix. This will determine the amount of time and resources needed to obtain the results. In circumstances where constructing the entropy matrix becomes extremely challenging, less precise algorithms, such as greedy or genetic algorithms, may be preferable. When results that are extremely close to the optimum are required, the genetic algorithm is set at an intermediate level. The greedy algorithm produces acceptable results in less time. Backtracking also tracks distribution and ensures optimal entropy minimisation. Unfortunately, as the size of the case study's data increases, so will the amount of time and resources required. Unfortunately, if these algorithms are used without first computing the entropy matrix and the ideal *k*-cluster, the number of clusters chosen and the distribution of the traces will almost certainly result in an assignment that is not optimal. Finally, the results of the metrics in various examples, as well as the time and resources required to obtain the solutions for each one, suggest that there is a direct relationship among the following elements: *(i)* activities and how they are related, *(ii)* the time and resources needed to calculate the entropy matrix and the optimal $k - cluster$, *(iii)* the time required for each algorithm to achieve the clusters, and *(iv)* the range of values in which the metrics fall.

In conclusion, our solution **S-AL-2** has allowed us to demonstrate that the use of clustering in the traces of an event log helps to manage spaghetti-like models and produces simpler solutions with the appropriate level of optimality according to the resources available to experts and the abstraction lever required.

2.5 OBJ 4: Process Optimisation

Process optimisation is the practise of increasing organisational efficiency by improving processes. It is done to achieve business objectives. Although there are many ways to apply process optimisation, the bottom line is that it can reduce costs and maximise results. Therefore, the optimisation of business processes is a quantitative tool that helps to make better decisions.

This section is dedicated to the solution proposed in this thesis for the objective of process optimisation. We have applied it in the context of manufacturing. Specifically, on the Airbus Space and Defence aircraft assembly line. On this assembly line, we have managed to optimise its processes thanks to the inclusion of flexible processes that help to make better decisions. In this case, we improved costs, which is really important if we scale the level of expenses that can accumulate in such a large industry.

Our proposal was to optimise the business processes defined for diagnosing the assembly processes. The kinds of decision that are made in this context cannot be solved with a decision tree or a DMN table, as they are based on past decisions and occurrences. The complexity of these systems makes it necessary to execute troubleshooting processes when a malfunction is found. Since hundreds of components can be responsible for the problem, to reduce the cost of analysing all of them, the definition of a troubleshooting process that infers the most probable component to evaluate according to the former diagnosis and the cost associated with each action is required. In addition, we deal with complex events, with different variables that help us isolate the failure.

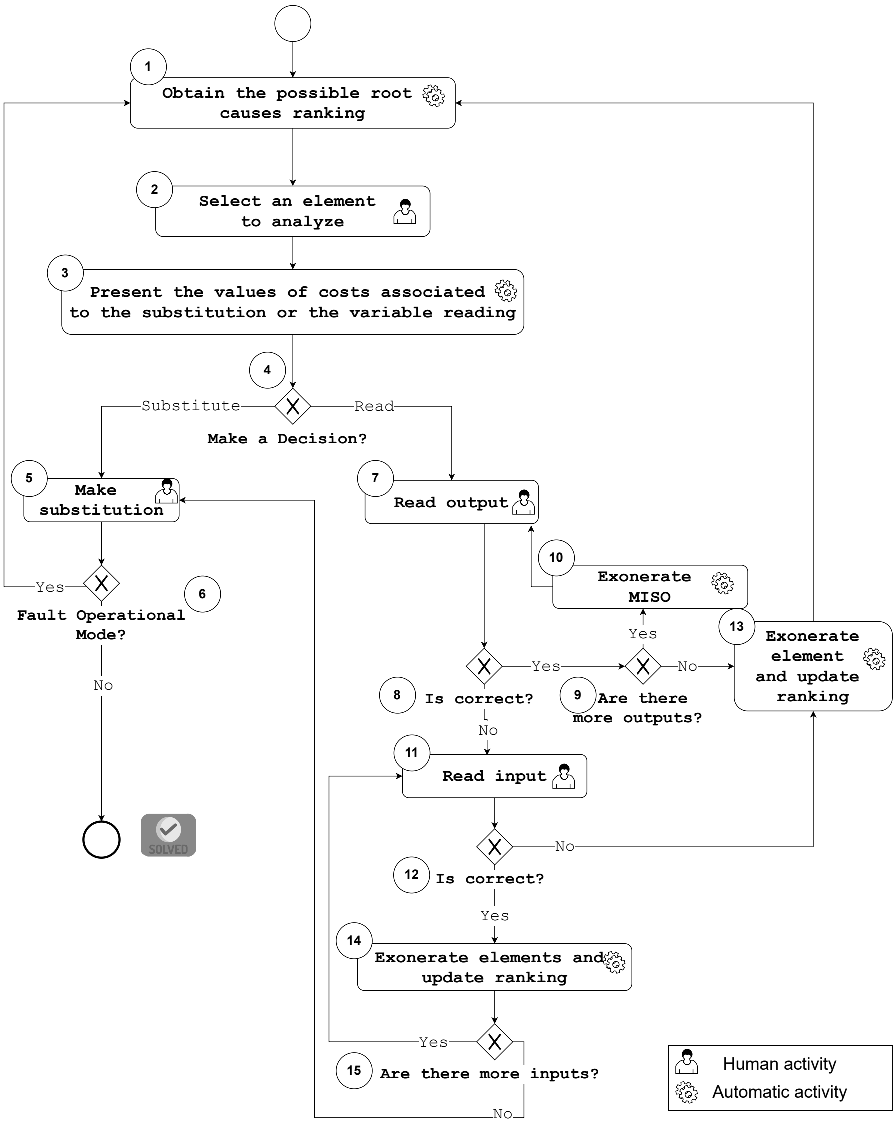

1 Obtain the possible root causes ranking
2 Select an element to analyze
3 Present the values of costs associated to the substitution or the variable reading
4 Make a Decision?
Substitute
Read
5 Make substitution
6 Fault Operational Mode?
Yes
No
SOLVED
7 Read output
8 Is correct?
Yes
No
9 Are there more outputs?
Yes
No
10 Exonerate MISO
11 Read input
12 Is correct?
No
Yes
13 Exonerate element and update ranking
14 Exonerate elements and update ranking
15 Are there more inputs?
Yes
No
Human activity
Automatic activity

FIGURE 2.6: Stages of the flexible guided process

Due to the increasing complexity of systems and the diverse origins of potential malfunctions, troubleshooting processes must be redefined. After detecting a malfunction, troubleshooting is a set of steps to perform a systematic analysis of symptoms. In certain complex systems, such as aircrafts, the cause of a malfunction can be any of several factors, and diagnostic techniques help engineers determine the cause of the unexpected behaviour. However, because an aircraft contains a large number of components, the list of the origin of possible faults can be extremely long, and analysing each element on the list until the responsible element is found can be time-consuming and error-prone. Alternatively, by indirectly validating a component's behaviour, certain input/output signals can be read to prevent the replacement of a properly functioning component. To determine the signals to read and the components to replace, we identified the relevant component of the model. We then proposed a troubleshooting process in an effort to minimise the cost of the action by combining structural analysis, the likelihood that a component will malfunction, and the cost of each additional signal read and replaced component. The proposal was validated in a system using real-world data that was acquired in partnership with Airbus Defence and Space.

Our approach is a framework to aid the troubleshooting process after the discovery of a malfunction in order to isolate the component at fault from a sorted list by automating as many steps as possible and guiding the operator to reduce the expense of detection and component replacements. The guide is based on the creation of a flexible business process that incorporates both human experience and the former traces in the decisions. The method is based on the incorporation of fault probability, substitution cost, reading cost, and structural analysis, resulting in a customised solution for each malfunction and signal sequence. Because both the probability and the substitution cost must be combined for each element of the model, a function must be included to optimise multiple objectives.

The process proposed for the detection of the component responsible for the failure can be seen in Figure 2.6 that describes the steps of the troubleshooting process. Once a malfunction is detected, Activity 1 obtains the possible root cost through a structural analysis of the aircraft system, including the probability of fault of each of them according to the information extracted from previous traces of troubleshooting. An expert selects one of the possible components in Activity 2, obtaining the possible cost associated with substituting it (Activity 5) or making readings to ensure that this is the responsible one (following Activity 7). If other parts of the system are read, that information is stored to be taken into account for future isolation processes and be included in the decision-making to ascertain the possible responsible. The knowledge and analysis of the possible responsible taking into account former instances of the processes have been included in Activity 1, as detailed in [99].

2.6 Conclusions

In this thesis, different approaches have been presented to cover different stages of the process management life cycle. Firstly, to achieve the first objective **(OBJ 1)**, we conducted a systematic mapping study to analyse, within these life cycles, the phases and activities that have been less studied and have fewer proposed solutions. As a result, we extracted three challenges corresponding to different stages, which have been addressed in this thesis. The first challenge is related to the measurement and assessment of the quality of event logs **(OBJ 2)**, for which we have developed a set of metrics and a methodology capable of estimating the quality of a given event log based on different dimensions of data quality. Then, with the results of this estimation and following the methodology, which is adaptable to any domain and size of the problem, it must be decided whether a further step is needed to improve this log of events and consequently improve its quality assessment **(OBJ3)**. To carry out this improvement, we propose two frameworks. The first is LOADING-NLP, a framework that aids in activity relabelling by applying simple natural language processing techniques to activity labels that have been extracted from textual descriptions that specify the action at different levels of granularity. The second framework is COLOSSI, which addresses the problem of spaghetti process as a result of a process discovery step. Thanks to COLOSSI, we have been able to prove that by performing trace clustering tasks on the event log, it is possible to obtain simpler sub-logs and sub-views of the discovered process model. Finally, we have made a proposal related to the challenge of process optimisation **(OBJ4)**, using a framework capable of automating diagnostic tasks, which incorporates tasks extracted from human experience and past decisions into the verification process, to minimise costs.

Chapter 3

Publications

This chapter discusses the results and presents the contributions that comprise and support this thesis. As a result, Section 3.1 presents the results of this thesis, by means of an enumeration of the contributions (Section 3.1.1). Finally, Section 3.2 contains the main publications.

3.1 Summary of the Publications

This section introduces the thesis’s contributions. Table 3.1 summarises the results achieved for each objective traced for this thesis, classified by scope, as well as the publications produced for each objective and the tools that support each proposal.

Scope	**OB1 - Overview analysis**	**OBJ2 - Event log quality**		**OBJ3 - Event log preparation**		**OBJ4 - Process optimisation**
Problem	P-R	P-DQ		P-AL		P-OP
Solution	S-R	S-DQ-1	S-DQ-2	S-AL-1	S-AL-2	S-OP
Publications	COMIND [100]	SPLC’19 [121] EMSE [98]	BPMDS’21 [97]	BPMDS’21 [97]	SPLC’19 [121] EMSE [98]	IEEE Access [99]
Tools	Systematic Mapping Study	LOADING-NLP	COLOSSI	COLOSSI	LOADING-NLP	Guided Troubleshooting

TABLE 3.1: Summary of the objectives and results of this thesis.

3.1.1 Contributions

The contributions of this thesis are divided into two categories. The first is the compendium of articles that comprises the core of this thesis. It is made up of indexed papers in which the PhD candidate is the first author. The second category includes articles that support the objectives of this thesis but are not included in the compendium. The contributions that make up the **compendium of articles** are the following:

- Belén Ramos-Gutiérrez, Antonia M. Reina Quintero, Luisa Parody, María Teresa Gómez López. **When business processes meet complex events in logistics: A systematic mapping study**. *In Computers in Industry (Vol. 144, p. 103788). Elsevier BV.* [100]. **Rank**: Q1 (JIF’21 11.245)

- Belén Ramos-Gutiérrez, Ángel Jesús Varela-Vaca, José A. Galindo, María Teresa Gómez López, David Benavides. **Discovering configuration workflows from existing logs using process mining.** *In Empirical Software Engineering (Vol.26(1)). Springer* [98]. **Rank**: Q1 (JIF'21 3.762).

- Belén Ramos-Gutiérrez, María Teresa Gómez López, Diana Borrego, Rafael Ceballos, Rafael M. Gasca, Antonio Barea. **Self-Adaptative Troubleshooting for to Guide Resolution of Malfunctions in Aircraft Manufacturing.** *In IEEE Access (Vol. 9, p. 9380188). IEEE.* [99]. **Rank**: Q2 (JIF'21 3.76).

The additional contributions that support this thesis are listed below:

- Belén Ramos-Gutiérrez, Ángel Jesús Varela-Vaca, F. Javier Ortega, María Teresa Gómez López, Moe Thandar Wynn. **A NLP-Oriented Methodology to Enhance Event Log Quality.** *In International Conference Business-Process and Information Systems Modeling (BP-MDS 2021).* [97].

- Ángel Jesús Varela-Vaca, José Angel Galindo, Belén Ramos-Gutiérrez, María Teresa Gómez López, David Benavides. **Process mining to unleash variability management: discovering configuration workflows using logs.** *In International Conference on Software Product Lines (SPLC 2019).* [121] **Rank**: SCIE RANK GGS Class 2.

3.2 Publications

This section includes the details of the publications that make up the compedium of articles and one additional publication in which the Ph.D. candidate is the first author. Sections3.2.1, 3.2.2 and 3.2.3 include the contributions of the compendium, while Sections 3.2.4 and 3.2.5 present additional contributions.

3.2.1 When business processes meet complex events in logistics: A systematic mapping study

Published in Computers in industry, vol. 144, pag. 103788, 2023.

- ***Authors**: Belén Ramos Gutiérrez, Antonia M. Reina Quintero, Luisa Parody, and María Teresa Gómez López.*
- ***DOI**: `https://doi.org/10.1016/j.compind.2022.103788`.*
- ***Rating**: Q1 (JIF'21: 11.245) .*

Computers in Industry 144 (2023) 103788

Contents lists available at ScienceDirect

Computers in Industry

journal homepage: www.sciencedirect.com/journal/computers-in-industry

When business processes meet complex events in logistics: A systematic mapping study

Belén Ramos Gutiérrez [a], Antonia M. Reina Quintero [a,*], Luisa Parody [b], María Teresa Gómez López [a]

[a] Dpto. Lenguajes y Sistemas Informáticos, Universidad de Sevilla, Spain
[b] Dpto. Métodos Cuantitativos, Universidad Loyola Andalucía, Spain

ARTICLE INFO

Keywords:
Logistics
Event-driven business process
Complex event processing
Business process
Systematic mapping study
Systematic literature review

ABSTRACT

Logistics processes are attracting growing attention because of the globalisation of the market. Its growing complexity and the need for reducing costs have provoked the seek of new solutions based on the processing of the complex events that the business processes produce. Event-Driven Business Process Management (EDBPM) is a discipline that studies the integration of business processes and complex events. The analysis of the maturity level of the approaches and gaps to point out future lines of research could help not only logistics organisations, but also academia. Logistics organisation could benefit from producing more environmentally friendly and optimal solutions in transport, and academia could benefit from revealing open problems. Thus, this study aims to identify current approaches, frameworks, and tools that integrate business processes and complex events in the logistics domain. To do so, we follow a systematic approach to do a mapping study that captures and synthesises the approaches, frameworks, and tools that integrate these two fields. As a result, 10,978 articles were gathered and 169 of them were selected for extraction. We have classified the selected studies according to several criteria, including the business process life cycle in which they are being applied, the business process modelling language, and the event process modelling language, among others. Our synthesis reveals the open challenges and the most relevant frameworks and tools. However, there is no mature enough framework or tool ready to be used in companies, and a promising research must provide solutions that cover all phases in the process life cycle.

1. Introduction

Product exchange and globalisation of the market have caused an increase in logistics processes as a key part of supply chain management. However, the rise in shipping rates and the cost of cargo containers, the increase in fuel cost, the lack of wooden pallets, the clogging of containers in port yards, or the low availability of storage make the logistics problem a global concern that Forbes has classified as much bigger than the pandemic (Broadman, 2021). To solve these problems, Information and Technology (I&T) is playing an important role in supply chain organisations, as stated in the Gartner report (Klappich et al., 2020). This report claims that “by 2023, 50% of global product-centric companies will invest in real-time transportation visibility platforms”.

In this globalisation context, logistics processes might be very complex since they can choreograph various entities, including resources and time restrictions. Furthermore, the collaboration of various entities, which may be geographically distributed, generates an enormous number of events that are complexly related. As a consequence, the extraction of knowledge from logistics processes implies the analysis of the processes that choreograph the organisations that collaborate, and the analysis of single events to derive other events defined at a higher level of abstraction, known as complex events.

Traditionally, Business Process Management (BPM) and Complex Event Processing (CEP) are the two disciplines that face the problems related to business processes and complex events separately. However, recently, Event-Driven Business Process Management (EDBPM) (Ammon, 2009a) has emerged as a new discipline to integrate both, in such a way that the events generated through the BPM systems can be paralelly analysed by a CEP taking the advantages of both disciplines. On the one hand, Business Process Management (BPM) (Weske, 2012) is the most valuable corporate asset, and according to the Business Process Management Market Research Report (Future, 2021), the “BPM market is expected to grow to approximately USD $16 billion by

[*] Corresponding author.
E-mail addresses: brgutierrez@us.es (B. Ramos-Gutiérrez), reinaqu@us.es (A.M. Reina Quintero), mlparody@uloyola.es (L. Parody), maytegomez@us.es (M.T. Gómez-López).

https://doi.org/10.1016/j.compind.2022.103788
Received 21 January 2022; Received in revised form 22 September 2022; Accepted 24 September 2022
Available online 14 October 2022

2023". BPM represents an integration of technologies and methodologies that facilitates the modelling and deployment of process models by coordinating a set of activities choreographed to achieve the objectives of an organisation (Pérez-Álvarez et al., 2018). Complex Event Processing (CEP) (Luckham, 2012) refers to a set of concepts and principles for processing events and the methods for implementing those concepts. The advantages of integrating both BPM and CEP are widely known (Eyers et al., 2016), and have been considered in logistics (Ammon et al., 2008; Schiefer et al., 2007; Emmersberger et al., 2009), due to the complexity and heterogeneity of the systems and the types of events that are handled in this context. However, there is no comprehensive overview of the state of research on Event-Driven Business Process Management in the logistics context.

The aim of this paper is to analyse the state-of-the-art of EDBPM in the logistics context to understand how business processes and complex events are being integrated into logistics processes. To do so, we conducted a systematic mapping study that aims at giving an overview of the research area through classification and counting contributions in relation to the categories of that classification (Khan et al., 2019). The results of this mapping study are not only valuable for organisations because they can be aware of the most mature approaches, frameworks, and tools, but also for researchers because they reveal the gaps that can drive future research lines. Therefore, if an organisation with logistics management wants to leverage the level of digitalisation of its processes, our study can provide information on how many approaches, techniques, and tools exist to support the deployment of business processes through complex event processing in the logistics context.

The rest of the paper is structured as follows: Section 2 introduces the background needed to understand the analysis; Section 3 presents the related work; Section 4 details the method we used in our systematic mapping study; Section 5 presents the data extracted following the method previously mentioned; in Section 6, the research questions are discussed; in Section 7 the threads to validity are analysed; and finally, Section 8 draws conclusions.

2. Background

This section defines the concepts needed to fully understand this study according to the main areas of research that frame this work: logistics, business processes and complex events.

2.1. Logistics

The importance of supply chain management has increased drastically during the last decades, especially with the necessity of coordinating various entities. While supply chain is related to the more general actions related to coordinate and manage items and persons in an organisation, the term logistics is used to refer to the set of activities that describe the flow of items in a company or its interchanging between various of them. In this context, a **logistics system** (Ghiani et al., 2013) describes the activities that determine the flow of the materials and information among the facilities of the organisations that can be in different places. It includes the infrastructure, equipment, means and resources (including humans) necessary for performing the activities. The capacity of modelling (Bassil et al., 2004) and reasoning over these systems (Chow et al., 2005) can improve performance in real scenarios. For this reason, the advantages that business process management can offer make them prone to logistics systems.

2.2. Business processes

Business process management is a mechanism for modelling, observing and improving the activities developed in an organisation to achieve their goals according to a set of constraints that govern their behaviour (Gómez-López et al., 2015). Process modelling is the first step to understand how the activities performed by organisations, both manually and supported by information systems, are related. According to Weske (2012), a **business process**[1] is composed of a set of activities belonged from one or over one organisation. These activities are performed in a coordinated way to achieve a business goal. Thus, **business process management** (Dumas et al., 2013) gathers the techniques and methods to support the life cycle of business processes. Furthermore, according to Augusto et al. (2019), the business process life cycle is composed of different phases: identification, discovery, analysis, redesign, implementation, and monitoring and controlling.

For modelling business processes in the phases of the life cycle different languages can be used, both imperative and declarative. Thus, a **business process modelling language** is a language oriented towards the description of a set of rules to govern how the elements involved in a process can be combined (Dumas et al., 2013). One of the most widely used business process modelling language is BPMN (OMG, 2013), although others can be used, such as Petri-nets (van Hee et al., 2013), YAWL (Ter Hofstede et al., 2009), Declare (Pesic and Van der Aalst, 2006), BPEL (OASIS, 2007), GSM (Guard Stage Milestone) (Hull et al., 2010) and UML Activity Diagram (Force, 2001).

2.3. Complex events

The complexity of logistics processes is derived from the multiple entities that can be involved, the distribution of the processes, and the different levels of abstraction of the generated events. This complexity makes necessary the analysis of the single events produced to infer more complex actions. These complex actions are represented by more than one event, known as **complex events**.

In computer science, an event can be defined as "anything that happens, or is contemplated as happening" (Chandy et al., 2011). However, the definition of event varies depending on the context (BPM or CEP). From a business point of view, an **event** is an action or occurrence that affect the business, and that happens in a timestamp that may be handled and stored by a software system. Events can include attributes, such as time at which they happened, resources or location (Luckham, 2012). Furthermore, in the context of business processes, event logs used to be represented by using the XES standard (eXtensible Event Stream).[2] Event logs are formed of a set of traces that contains a sequence of events.

The term **complex event processing (CEP)** (Luckham, 2005) gathers a set of technologies to discover relationships between single events to infer the existence of relevant information, such as timing, causality, membership, or existence of patterns of behaviour or correlation (Schiefer and McGregor, 2004). To describe the type of events and the attributes they hold, we need an event model, and for processing them an event processing language is necessary. Thus, an **event model** is the mechanism for describing the events produced by the involved systems (Cugola et al., 2015), that includes a set of attributes At and a set of domains D for each attribute, $\{At_1; D_1, \ldots, At_n; D_n\}$. Whereas, an **event processing language (EPL)** is a high-level declarative language that permits defining functions to manage events within a data stream.

[1] There is a wide range of definitions of business process in literature (Davenport, 1993; Hammer and Champy, 1993; Johansson et al., 1993). We have chosen the definition proposed by Weske because it is the most recent and one of the most relevant taking into account the number of citations.

[2] https://www.xes-standard.org/.

The declarative capacity of the EPL facilitates the description of complex event conditions, event correlations, and the inclusion of possibly spanning time windows (Luckham, 2008).

The discipline that integrates business processes and complex events is named event-driven business process management. **Event-Driven Business Process Management (EDBPM)** (Ammon, 2009b; Goetz, 2010) combines both Business Process Management (BPM) and Complex Event Processing (CEP) techniques to support the actions performed in a business process that originates complex events. In this context, the BPM engine generates events during the execution of the daily processes, that are analysed by the CEP in a parallel way, with the objective to detect the relevant information. These events can also influence the execution or the process and give as result other events.

3. Related work

We consider as related work the secondary studies focused on the combination of complex events and business processes in the logistics context. Secondary studies are those studies that summarise and classify the published literature, in contrast to primary studies that present a primary work and are the focus of the analysis performed in secondary studies. Secondary studies in software engineering can be classified into surveys, systematic literature reviews, and systematic mapping studies.

A survey is a kind of secondary study that is not conducted systematically nor with a systematic protocol (Khan et al., 2019). On the contrary, systematic literature reviews and systematic mapping studies follow a systematic approach by following a systematic protocol. One of the fundamental differences between systematic literature reviews and systematic mapping studies is their goals (Kitchenham et al., 2010, 2015). While the goal of systematic literature reviews is to address very specific questions, systematic mapping studies aim at giving an overview of a research area through classification and counting contributions in relation to the categories of that classification (Khan et al., 2019).

None of the analysed secondary studies has been conducted as a mapping study and only a few analyse the integration of business processes and complex events in the area of logistics. To analyse these secondary studies, we have classified them into systematic literature reviews, surveys, and other secondary studies. The following subsections analyse in detail these groups of studies.

3.1. Systematic literature reviews

The research questions of systematic literature reviews are more focused compared to the questions proposed in mapping studies. We have found various systematic literature reviews focused on some aspects of the relationship between complex events and business processes (Krumeich et al., 2014; Amjad et al., 2018; Augusto, 2020). These studies explore how events drive the management of business processes (Krumeich et al., 2014), how business requirements can be modelled and validated with event-driven processes (Amjad et al., 2018), and the accuracy and efficiency of process mining in the context of event-driven processes (Augusto, 2020). However, these studies do not give an overview of how, within the life cycle of processes, events can be applied or affected, and of course, how their study can be applied to the field of logistics.

Falco (Jaekel, 2019) conducts a systematic review of the existing literature in the area of cloud logistics. He selects 83 studies and classifies them according to the meaning of the term “cloud logistics”. However, the logistics encompasses much more than what is covered by the term cloud logistics, so this study is insufficient for answering the research questions we proposed.

3.2. Surveys

Surveys are another type of secondary study. In general, surveys are less rigorous than mappings and systematic literature reviews. We have found a set of surveys related to CEP, BPM, and logistics (Davidsson et al., 2005; Li, 2005; Fulop et al., 2010; Akila et al., 2016; Li et al., 2017; Dayarathna and Perera, 2018). As in some of the aforementioned studies, the surveys have not been conducted systematically, and as a result, neither the research questions nor the identified studies are explicit, so they are difficult to replicate.

Davidsson et al. (2005) presents a consistent view of the research efforts made in freight transportation and some work of traffic and transport of people, but the more generic aspects of logistics, such as the organisational part, external factors (traffic, weather, …), or even the part of supply chain management are not included. Furthermore, they do not consider the role of CEP and BPs in logistics. In Li (2005), Li surveys the recent progress of event-driven applications and investigates their potential implications at the system and middleware levels. On the one hand, this study focuses on the military and intelligence community and the logistics is not included. On the other hand, activities are not contextualised within a business process. Fulop et al. (2010) examine the complex event processing and the related field of predictive analytic. The survey includes the terminology, research achievements, existing solutions, and open issues related to both areas. In Akila et al. (2016), Akila et al. analyse the techniques, challenges, and future directions of complex event processing over uncertain events. Lastly, Dayarathna and Perera (2018) summarise the latest cutting-edge work in 2018 done on event processing system architectures, event processing use cases, and event processing open research topics. In these three surveys, the focus is on events without considering business processes or logistics. Finally, Li et al. (2017) presents a variety of data-driven techniques and applications with a focus on computing system management.

3.3. Other secondaries studies

This group of studies includes secondary studies that cannot be formally classified as mapping studies, systematic literature reviews, or surveys. There are several points that differentiate the studies in this group from ours. Firstly, neither of the studies deals with the combination of CEP and BPM in the logistic context. Alias et al. (2016) focus on the combination of events with logistics, while Soffer et al. (2019) focus on the combination of events with processes. Secondly, the number of studies they analyse is considerably smaller than the number of studies we have considered in our mapping study. In addition, the rigour of such mappings precludes the exact replication of the studies.

The mapping study by Alias et al. (2016) aims at determining the updated status in 2016 of complex event processing and predictive analytics in the transportation and logistics sector. One of the major problems with this study is that it does not follow a formal method for conducting the mapping. Among other things, they do not specify exactly what terms have been used to perform the searches, making the replication of the study difficult. They identified 58 studies that are categorised into the areas of transport, logistics, and supply chain management. However, they do not mention any relation with the business process technology.

Soffer et al. (2019) focus on the challenges and opportunities for the combination of CEP and process mining. Although they illustrate and motivate the study through a logistics example, the focus of the study is not the logistics sector. Furthermore, in relation to replicability, they do not present the number of articles analysed, nor the sources used, nor what criteria they use to select the articles. Finally, as their research is focused on process mining, they do not cover the studies that are framed on the rest of phases of the business process life cycle.

Table 1
Research questions.

Id		Research question
RQ1	What are the demographics of the published studies?	
	RQ1.1	Which contributions were made over the years?
	RQ1.2	Which are the most influential researchers in the area and were are they from?
	RQ1.3	Which are the most influential studies?
	RQ1.4	Which are the top venues?
RQ2	What are the current trends in the area?	
	RQ2.1	What is the type of contribution made by the study?
	RQ2.2	Which are the application areas of the studies?
	RQ2.3	What type and which business process modelling languages are utilised?
	RQ2.4	What type and which event processing languages are utilised?
	RQ2.5	How is logistics covered in the studies?
	RQ2.6	Which are the event producers?
RQ3	What are the potential gaps in the area?	
	RQ3.1	Is event-driven business process treated in every phase of the process life cycle in a logistics context?
	RQ3.2	What are the challenges taken off in the field?

4. Method

The main purpose of our study is to analyse the state-of-the-art of integrating business processes and complex events in the logistics context. Our goal is to identify the current approaches, frameworks, and tools, their maturity level, and the areas not explored in depth to point out future lines of research. Thus, the principal objectives of this mapping study are as follows:

OBJ1: To synthesise the studies in the logistics domain with approaches related to event-driven business processes to provide insight into this area.

OBJ2: To investigate the trends in event-driven business processes in the logistics domain.

OBJ3: To reveal the gaps and foresee future directions of research.

To conduct the mapping study, we have followed the guidelines proposed by Kitchenham and Charters (2007), Kitchenham et al. (2015) and Kuhrmann et al. (2017) for performing systematic literature reviews in software engineering. And the guidelines for the use of snowballing by Wohlin (2014).

The review process consists of three main stages: planning, conducting, and documenting the review. Fig. 1 depicts these stages and the principal activities involved in each phase. The protocol is defined during the planning stage (phase 1), which includes assigning responsibilities and activities to the reviewers. The result of this stage is the protocol. The activities defined in the protocol are executed in the conducting stage (phase 2). These activities include identification, selection, assessment of the quality of primary studies, data extraction, and synthesis of the extracted data. Since phase 2 is the stage that comprises more activities, we have detailed it in Fig. 2 by means of a process specified with the BPMN notation (OMG, 2013). Fig. 2 describes the activities performed in Phase 2, in which the different search engines are searched. Duplicated studies are removed from the set of studies to be analysed. The rest of the studies are submitted for first screening and, if they are related to the objectives of the study, for a later careful read. After this analysis, the researchers discuss the selection to arrive at a consensus on possible doubts. The selected studies are used to perform a backward and a forward snowballing, that is, the studies that are included in the references and the studies that cite a selected study are included in the set of studies to be analysed. After that, each selected study is used to extract the required data, and its quality is assessed. Eventually, the extracted data is synthesised to facilitate later knowledge extraction. Finally, in phase 3, a report is produced to distribute the results.

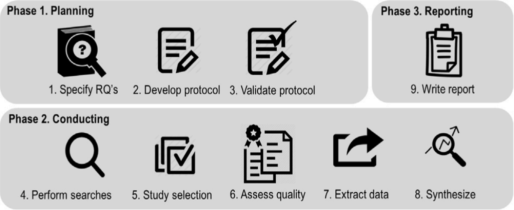


Fig. 1. Main stages for conducting the SMS according to Kitchenham and Charters (2007).

4.1. Research questions

The specification of the research questions is an essential part of the conduction of a systematic mapping study because they drive the entire review process (Kitchenham et al., 2015). As we have developed a mapping study, some research questions are broad and are concerned with classifying the literature. The questions are aligned with the three objectives mentioned previously. We have defined three broad fundamental questions (R1-R3) that are based on the categories of research questions identified by Khan et al. in Khan et al. (2019): demographics, current trends, and research gaps. Additionally, each question has been broken down into some secondary questions.

Table 1 lists the research questions addressed in this study. Question RQ1 belongs to the demographics category and aims to characterise the studies published in the area. This question has been broken down into sub-questions RQ1.1-RQ1.4. These subquestions will help us identify the number and frequency of publications, the most influential researchers and their countries, the most influential studies, and the top venues. Question RQ2 belongs to the category of current trends and aims to identify the nature of the existing work in the area. We have broken this question down into sub-questions RQ2.1-RQ2.6 that help us to identify the type of contributions, the application areas, the business process modelling and event processing languages, the logistics coverage, and the event producers. Finally, RQ3 belongs to the research gaps category and aims at the identification of gaps in the area. This question has been broken down into sub-questions RQ3.1-RQ3.2. The rationale behind these sub-questions is the identification of the process life cycle phases in which the approaches are framed, and the identification of the challenges in the area.

4.2. Search process

To collect the studies, we used a broad automated search followed by an automated backward snowballing and a later forward snowballing. Table 2 lists the digital libraries and search engines used as data

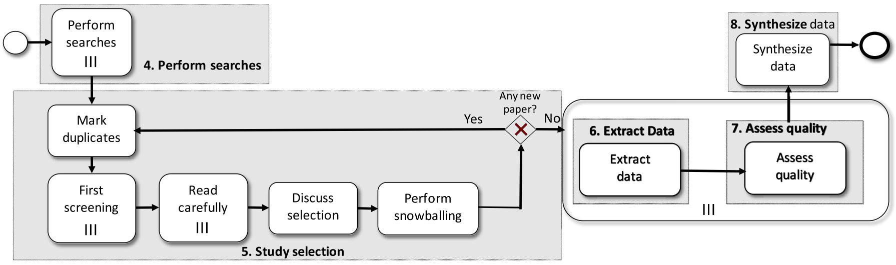


Fig. 2. Phase 2: Process definition for conducting the review.

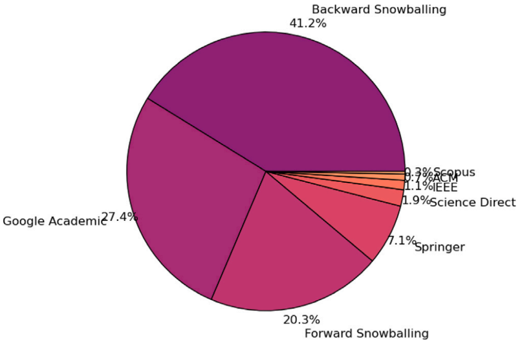


Fig. 3. Proportion of studies recovered in relation to the data source.

Table 2
Digital libraries and search engines employed as data sources.

Search engine	URL
ACM DL	http://dl.acm.org
Springer	http://www.springer.com
IEEE Xplore	http://ieeexplore.ieee.org
ScienceDirect	http://www.sciencedirect.com
Google Academic	http://scholar.google.es
Scopus	http://www.scopus.com

sources during the automated search process. Furthermore, to arrive at proper search strings, we have followed a "Trial-and-error Search", as recommended in Kuhrmann et al. (2017). The terms resulting from this approach are the following:

- "complex event" AND "business process" AND "logistics"
- "event-driven business process" AND "logistics"
- "complex event" AND "workflow" AND "logistics"

As our goal is to have an overview of how event-driven business processes are used in the logistics domain, we have selected the following keywords: complex event and business process. Furthermore, since the term workflow is used as a synonym of business process in this context (von Ammon, 2018), we have also included workflow as a search string.

The queries to collect the studies from the digital databases and search engines were executed in December 2020, and later, in March 2022, a second round has been carried out to update the searches. It deserves to be mentioned that the searches in ACM DL, Springer, IEEE Xplore, Science Direct and Scopus have been performed through the Web interface, while searches in Google Academic have been performed using *Publish or Perish*.[3] This tool returns a maximum of 1000 results for this search engine.

As search engines have different query syntaxes, the previous search strings must be rewritten for every search engine. Table 3 lists the query strings used in each search engine. Note that a Search ID identifies every search. This ID is automatically generated by the Start tool,[4] which is the tool that we have used as a support and helps us trace the origin of the recovered study.

Regarding snowballing, we have performed an automatic extraction of the references of the selected studies (the so-called backward snowballing) following a two-step approach: first, we have queried Crossref[5] with a Python-based script to extract references[6]; second, if the study is not indexed in Crossref, we have used GROBID (2008–2021) to extract the references directly from pdf files. The forward snowballing has been performed by querying Opencitations[7] with a Python-based script,[8] and has been complemented by manually querying Scopus.

As a result of the search process 10,978 studies have been found. Fig. 3 summarises the number of studies obtained from the data sources, as well as the number of studies obtained with the backward and forward snowballing process. Note that more than the 61% of the incorporated studies have been obtained thanks to the snowballing process.

4.3. *Study selection*

Once the studies have been recovered from the data sources, they are loaded into the Start tool, which helps us detect some duplicates (see the first activity in Fig. 2). As the tool only detects duplicates by checking the exact match of titles, some duplicates are marked manually. After that, an initial screening is performed based on title, abstract, and keywords.

The initial screening of the 3450 studies obtained as a result of the automatic searches in digital databases is undertaken by all researchers. The screening of the 7528 obtained with backward and forward snowballing was also undertaken by all researchers.[9] We kept a full record of the researchers who conducted the initial screening of the study. Table 4 lists the inclusion and exclusion criteria used in the review. Note that some of these criteria are de facto standards, as noted in Kuhrmann et al. (2017).

As a result of the screening, 621 studies have been selected. These studies have been thoroughly reviewed by all researchers. Note that the researcher who performed the screening of one study is not in charge

[3] https://harzing.com/resources/publish-or-perish.
[4] http://lapes.dc.ufscar.br/tools/start_tool.
[5] https://www.crossref.org/.
[6] https://github.com/reinaqu/bibreferences.
[7] https://opencitations.net/.
[8] https://github.com/reinaqu/snowballing-opencitations.
[9] See Section 7 to see who did what.

Table 3
Search queries employed in the different search engines.

Source	SearchIDs	Search string
ACM DL	SEARCH0	[Full Text: "complex event"] AND [Full Text: "business process"] AND [Full Text: logistics]
	SEARCH6	[Full Text: "event-driven business process"] AND [Full Text: logistics]
	SEARCH7	[Full Text: "complex event"] AND [Full Text: workflow] AND [Full Text: logistics]
Google Scholar	SEARCH5	"complex event"+"business process"+ logistics
	SEARCH16	"event-driven business process"+ logistics
	SEARCH17	"complex event"+"workflow"+ logistics
IEEE Xplore	SEARCH3	((("Full Text & Metadata":"complex event") AND "Full Text & Metadata":"business process") AND "Full Text & Metadata":logistics)
	SEARCH10	(("Full Text & Metadata":"event-driven business process") AND "Full Text & Metadata":logistics))
	SEARCH11	((("Full Text & Metadata":"complex event") AND "Full Text & Metadata":workflow) AND "Full Text & Metadata":logistics)
Science Direct	SEARCH2	"complex event"+"business process"+ logistics
	SEARCH12	"event-driven business process"+ logistics
	SEARCH13	"complex event"+"workflow"+ logistics
Springer	SEARCH1	"complex event"+"business process"+ logistics
	SEARCH8	"event-driven business process"+ logistics
	SEARCH9	"complex event"+"workflow"+ logistics
Scopus	SEARCH4	TITLE-ABS-KEY ("complex event" AND "business process" AND logistics)
	SEARCH14	TITLE-ABS-KEY ("event-driven business process" AND logistics)
	SEARCH15	TITLE-ABS-KEY ("complex event" + "workflow" + AND logistics)

Table 4
Exclusion and inclusion criteria.

Criteria	Description
Exclusion	The study is not written in English.
	It is a proceeding book.[a]
	It is an index or a table of contents, not a book chapter.
	It is an editorial.
	It is not a paper, but the slides of a presentation.
	It is a call for paper, not a paper.
	It is an encyclopaedia entry, not a paper.
	It is an extended abstract.
	The study occurs multiple times in the result set.
	The study is not accessible electronically.
	It is the result of an importation error.
	Title, keyword list, and abstract make explicit that the paper is NOT related to logistics, business processes, or complex events.
Inclusion	Title, keyword list, and abstract make explicit that the paper is related to logistics, business processes and complex events.

[a] This criteria helps to avoid duplicates due to the behaviour of some search engines, which returns two results when searching for a study: one that refers to the single study, and another one that refers to the whole book of the proceedings in which the paper was published.

Table 5
Assessment of Intrinsic IQ.

Category	Intrinsic IQ output
J1 ∨ C1	HIGH
J2 ∨ C2	MEDIUM
J3 ∨ C3 ∨ O1 ∨ O2 ∨ O3	LOW

of reading it carefully and extracting the information of that same study. Furthermore, if one reviewer has doubts about the acceptance or rejection of a study after a thorough review, another researcher reviews that study, and the final decision on the acceptance of that study is taken in a voting workshop in which all researchers are involved. That is, we have followed an approach similar to the alternative approach proposed in Kuhrmann et al. (2017).

Once a study is selected for extraction, a backward snowballing of the references included in the study and a forward snowballing of the papers that cite the study is made. As Start does not work properly with the references obtained through snowballing, we have set up a procedure to add the references to the tool and to maintain the traceability of the snowballing process. On the one hand, the references/citations obtained automatically from a study are included in a BibTex/RIS file that is loaded in a new search session in Start. The session has as a keyword the *id* of the study that includes the references. On the other hand, we have used a spreadsheet to annotate the *id* of the search session, the *id* of the study, and the number of references recovered.

As a result of the selection, 169 studies are collected. Subsequently, their quality is assessed as explained in Section 4.4, and the relevant information is extracted for our mapping study (see Section 4.5). Fig. 4 shows a diagram and a Start screenshot that summarises the results obtained during the three principal activities carried out in the review stage: identification, selection, and extraction.

4.4. Quality assessment

The quality assessment of the studies has been carried out in parallel with the data extraction activity, because, as in Varela-Vaca and Reina Quintero (2021), we have used an Information Quality framework (Wang and Strong, 1996) to assess the studies. The quality assessment depends on the type of publication and the quality of the venues and the amount and completeness of the extracted data. The framework (Wang and Strong, 1996) defines four dimensions of Information Quality, namely: Intrinsic, Contextual, Representational, and Accessibility. Information Quality (IQ) can be measured regarding one or several dimensions depending on the extracted attributes. As the format of the data and the access to information are irrelevant in our context, we measure only Intrinsic and Contextual IQ.

Intrinsic IQ (Wang and Strong, 1996) measures the accuracy, objectivity, believability, and reputation of the data for the task at hand. Thus, to measure this dimension, we take into consideration the type of publication, the venue, and the ranking of the venue, and as a result, a value is obtained based on a three-point Likert scale with HIGH, MEDIUM, and LOW values. On this scale, HIGH is the best result, and LOW is the worst result. To calculate the value, first, the study is classified into one of the categories shown in Fig. 5, and then the value is obtained according to the category (Journal, Conference/Workshop or Others), as shown in Table 5. Afterwards, to obtain the value of the Intrinsic IQ, we have to look after the category in Table 5, and return the associated value. For example, if the category of the paper is J1 or C1, then the value is HIGH.

Contextual IQ (Wang and Strong, 1996) measures the amount of data and the completeness of the extracted data. To obtain this measure, we have defined a questionnaire composed of Yes/No questions (see Table 6). In relation to the number of Yes and Noes, we have defined the metric depicted in Eq. (1) that returns a value on a scale of [0-1]. The contextual IQ is defined based on a three-point Likert

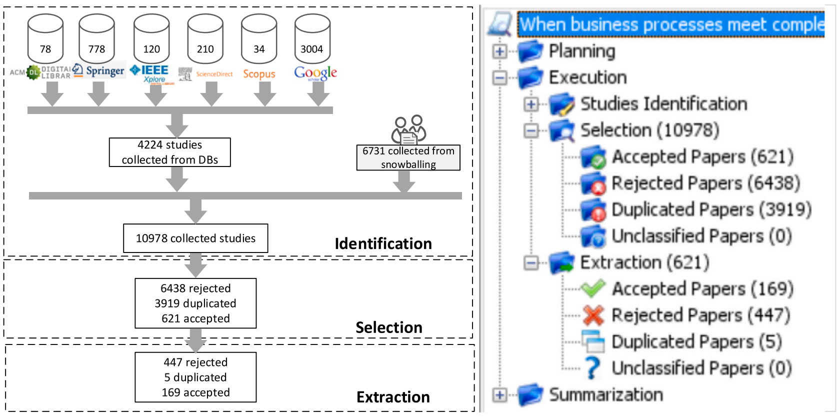


Fig. 4. Summary of the results.

$$
study = \begin{cases} Journal & \begin{cases} J1: & Journal\ indexed\ as\ Q1\ or\ Q2\ in\ the\ ISI\ WoK\ and/or\ Scimago\ Journal\ rankings. \\ J2: & Journal\ indexed\ as\ Q3\ or\ Q4\ in\ the\ ISI\ WoK\ and/or\ Scimago\ Journal\ rankings. \\ J3: & Journal\ not\ indexed\ in\ the\ ISI\ WoK\ nor\ Scimago\ Journal\ rankings. \end{cases} \\ Conference/Workshop & \begin{cases} C1: & Conference\ indexed\ as\ Class\ 1\ or\ Class\ 2\ in\ the\ SCIE\ Conference\ ranking. \\ C2: & Conference\ indexed\ as\ Class\ 3\ in\ the\ SCIE\ Conference\ ranking. \\ C3: & Conference\ not\ indexed\ in\ the\ SCIE\ Conference\ ranking. \end{cases} \\ Others & \begin{cases} O1: & Book\ chapters. \\ O2: & arXiv\ papers. \\ O3: & Technical\ report,\ Phd\ thesis,\ Msc\ thesis\ and\ others. \end{cases} \end{cases}
$$

Fig. 5. Classification of a study based on the type and quality of the venue.

scale with HIGH, MEDIUM and LOW values, in which HIGH is the best result and LOW is the worst result. We calculate the Contextual IQ value according to the criterion defined in Table 7. As stated in Even et al. (2010), data quality assessment cannot be defined in a general way because it is context dependent and, as a consequence, must be defined by domain experts. As our context differs from Varela-Vaca and Reina Quintero (2021), we maintain the same level of granularity as them, but we use different thresholds. Note that they present a multivocal mapping study that includes information from the grey literature, which requires a higher level of extracted data to classify a study with a high level of contextual IQ. In our case, a study with fewer than four extracted data has a low Contextual IQ, a study with seven or more extracted data has a high Contextual IQ; otherwise, the study has a medium Contextual IQ.

$$
m_{Completeness} = \frac{|Number\ of\ answered\ questions|}{|Total\ number\ of\ questions|} \tag{1}
$$

Finally, the quality of a study is determined based on the level of quality obtained in the two dimensions, that is, intrinsic and contextual. We consider the study acceptable when the HIGH or MEDIUM quality level is obtained in both dimensions. The study is rejected when both quality dimensions have a LOW value.

4.5. Data extraction and synthesis

The data extraction process has two stages: first, the data are extracted; and second, the extracted data are reviewed. All researchers are involved in the data extraction process. The researcher who selects a study for extraction is responsible for extracting the data from that study. If the researcher in charge of the extraction has doubts about the extraction of certain labels, we discussed them at a later meeting in which all researchers were involved. Data revision is made in parallel with data synthesis, because of inconsistencies, and errors are better detected at this point.

Table 6
Yes/No questions included in the quality questionnaire.

ID	Yes/No question
QA01	Could the type of contribution be extracted?
QA02	Could the application area be extracted?
QA03	Could the logistic coverage be extracted?
QA04a	Could the type of business process modelling language be extracted?
QA04b	Could the business process modelling language be extracted?
QA05a	Could the type of event processing language be extracted?
QA05b	Could the event processing language be extracted?
QA06	Could the event producer be extracted?
QA06	Could the process life cycle phase be extracted?
QA07	Could the challenge be extracted?

Table 7
Assessment of Contextual IQ.

Metric	Values	Contextual IQ
$m_{Completeness}$	>0.66	HIGH
$m_{Completeness}$	(0.33,0.66]	MEDIUM
$m_{Completeness}$	0.33≤	LOW

Data extraction is performed using three different tools: a Publication Form, a form that provides the Start tool with common information about publications, such as title, authors, and so on; an Extraction Form, a customised form created in the Start tool with data about the specific domain of the literature review; and an Excel Datasheet, to store information that has been obtained automatically, such as the

Table 8
Type of data extraction form and alignment of extraction fields with research questions.

Tool	Extracted data	RQ
Publication form	Publication title	RQ1.1, RQ1.2, RQ1.3
	Publication authors	RQ1.1, RQ1.2, RQ1.3
	Publication venue	RQ1.4
	Publication year	RQ1.1
Excel datasheet	Citations of the publication	RQ1.3
	Author's countries	RQ1.2
Extraction form	Type of contribution	RQ2.1
	Application area	RQ2.2
	Type of business process modelling language	RQ2.3
	Business process modelling language	RQ2.3
	Type of event processing language	RQ2.4
	Event processing language	RQ2.4
	Logistic coverage	RQ2.5
	Event producer	RQ2.6
	Process lyfecycle phase	RQ3.1
	Challenge	RQ3.2

citations of a publication or the author's country. Table 8 lists the extracted data, the tool in which the data is stored, and the research questions that can be answered using the extracted data.

Regarding the synthesis, we employ an approach based on both qualitative and quantitative methods to analyse the data. We use a qualitative approach when we are interested in questions about "what" and "how". To complement this qualitative analysis, we used descriptive statistics to discuss frequency and distribution.

4.6. Replication package

To strengthen the replicability of our review, we have published a bundle with all the artefacts and the final results of our study in url.[10] This bundle includes a Jupyter notebook with all generated graphics and some additional information, such as the raw numbers behind the graphics or the concrete studies under the different classifications made in the paper; three GitHub repositories, one with the Python code behind the graphics generation, and two with the Python-based scripts we have used to query Crossref[11] and Opencitations[12]; and the datasets used in the study (Ramos-Gutiérrez et al., 2021). Furthermore, the selected studies can be accessed online by following the information reported in a separate Selected Studies section in the Appendix included as supplementary material.

5. Data extraction results

The review process has been conducted from December 2020 to March 2022. During this process, we developed the protocol, identified and selected primary studies, performed data extraction and synthesis, and reported the results. All researchers participated in the entire process, as explained in Section 4.

5.1. Primary studies

Our set of primary studies is composed of 169 studies. For space reasons, the selected studies are listed in the Appendix as a table which is ordered by year and type of venue. The table shows the reference, the title, and the venue. Fig. 6 shows the percentage of studies by type of publication. As can be observed, almost half of the studies (46.2%) have been published in conferences; more than a quarter of the studies (26%) have been published in journals, and only 18.3% of the studies have been published in workshops. This demonstrates that the field is quite mature.

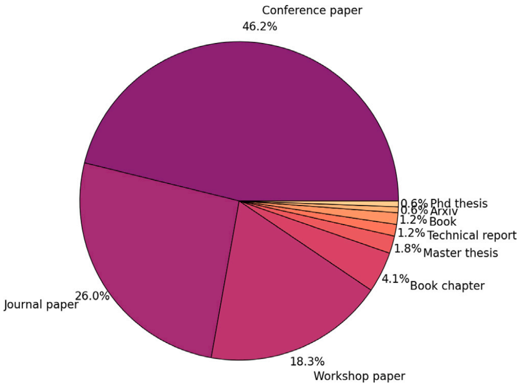


Fig. 6. Percentage of publications per type.

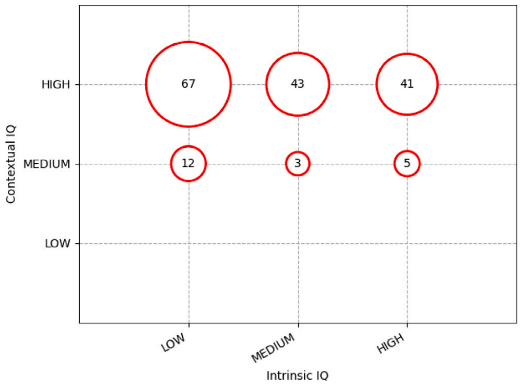


Fig. 7. Studies quality assessment.

5.2. Study quality assessment

The quality of each study has been determined by calculating the Intrinsic and Contextual IQ measurements (see Section 4.4). Fig. 7 shows a summary of the quality measurements. The figure depicts two dimensions, Contextual IQ and Intrinsic IQ, in a bubble chart. Each dimension can have the values LOW, MEDIUM, and HIGH. A bubble represents the number of studies that have the values depicted on the X and Y axes, respectively. Therefore, there are 41 studies that have HIGH Intrinsic and Contextual IQ; 43 studies that have a Medium Intrinsic IQ and a High Contextual IQ; and 67 studies that have a Low Intrinsic IQ and a High Contextual IQ. In summary, 151 out of 169 studies have a HIGH Contextual IQ, which means that in more than 89% of the studies the data can be extracted. Finally, it also deserves to be noticed that there is no study with Low Intrinsic and Contextual IQ, so no study has been rejected for quality reasons.

6. Discussion of research questions

In this section, we discuss the research questions by synthesising the results obtained from the collected studies.

RQ1. What are the demographics of the published studies?

This research question helps identify the number and frequency of publications, the top researchers in the area, the top countries, the

[10] http://www.idea.us.es/when-bp-meet-ce-in-logistics-a-sms/.
[11] https://www.crossref.org/.
[12] https://opencitations.net/.

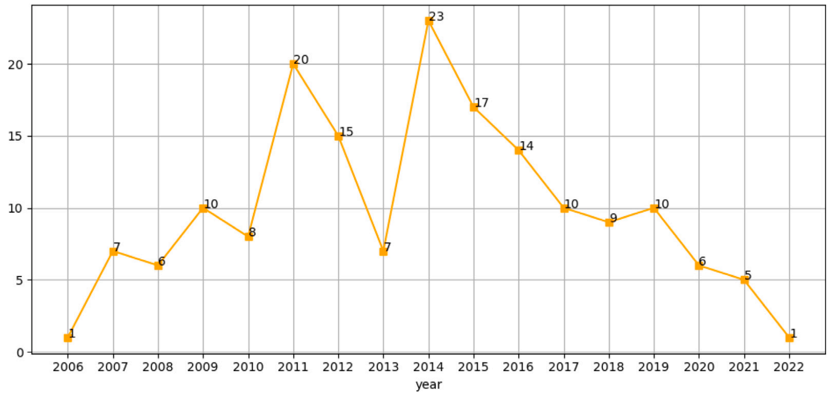


Fig. 8. Number of publications per year.

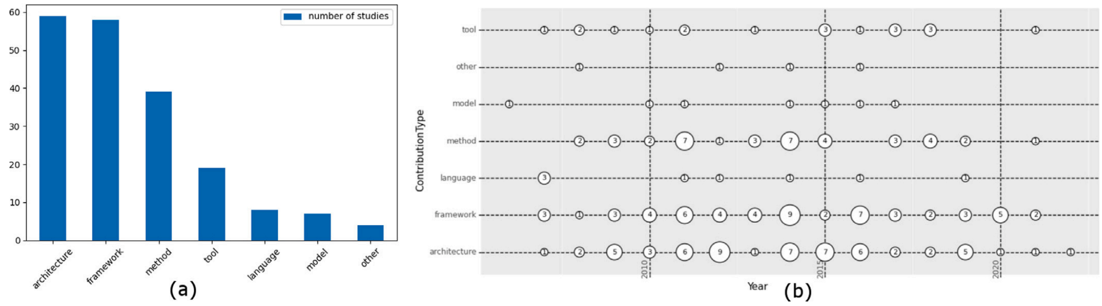


Fig. 9. (a) Proportion of type of contribution, and (b) Number of studies per contribution type and year.

most influential studies, and the top venues. For space reasons, this section is mainly focused on the discussion of RQ1.1. However, details on questions RQ1.2-RQ1.4 (authors, leading countries, most influential studies, and leading venues) are provided in the Appendix.

The goal of RQ1.1 is to provide information on the number and frequency of publications over time. The rationale behind this question is to analyse trends, such as the maturity of the field. Fig. 8 shows the number of relevant studies extracted in our review per year. As can be observed, the year with the highest number of publications is 2014, with a total of 23. Since 2014, the number of publications has been decreasing until 2018. This implies that event-driven business processes in logistics were an emerging research topic in 2007–2014. Afterwards, the number of publications has decreased until 2018, when it began to increase. Since then, it has started to decrease again in the years of pandemics (2020 and 2021).

RQ2. What are the current trends in the area?

This research question helps identify the nature of existing work in the area. The following subsections discuss the questions RQ2.1-RQ2.4, because they are the ones that allow us to obtain more interesting conclusions. The reader interested in RQ2.5 and RQ2.6 can consult the Appendix.

RQ2.1. What is the type of contribution made by the study?

This question aims to identify the nature of the work in the area. As in Petersen et al. (2008), the categories were determined by carefully reading the studies. It should be noted that there are studies that produce more than one type of contribution. For example, the study (Linden et al., 2013) proposes an architecture and a framework. Fig. 9(a) shows the proportion of types of contributions, and Fig. 9(b) shows the distribution of contributions per type and year. Architecture is the most frequent type of contribution (59 out of 169 studies contribute with an architecture), although the number of studies that contribute with a framework (58 out of 169) is also important. Unfortunately, most of the proposals that contribute with frameworks, tools and architectures do not offer a definitive version of their solution. In fact, many of them propose solutions that are still under development or are purely conceptual (24.5%), proofs of concept (10.7%) and prototypes (19.8%) (see the Appendix). Furthermore, most of the approaches that provide more details on the implementation of their solutions do not make their software publicly available (19.1%) and only 9.9% of the studies propose the integration of well-known commercial tools (see Fig. 10). Finally, all these proposals have one thing in common: they focus on the design or development of prototypes and software tools for academia; none of them constitutes a solution close to being a commercial solution applicable in industry.

RQ2.2. Which are the application areas of the approaches?

The aim of this research question is to identify the areas within the logistics sector in which the approaches are applied. The application areas have been obtained from Alias et al. (2016). They include maritime transport, air transport, multimodal transport, general transport, manufacturing, foodstuffs, transshipment, chemical and pharmaceuticals, retail, and others. After carefully reading the studies, we have classified the use cases, running examples, or descriptions they present according to the areas of application previously mentioned. Note that a study can be associated with more than one area of application. As

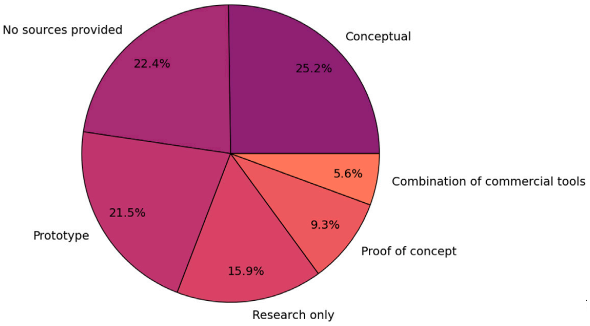


Fig. 10. Proportion of studies that contribute with frameworks, tools and architectures by feature.

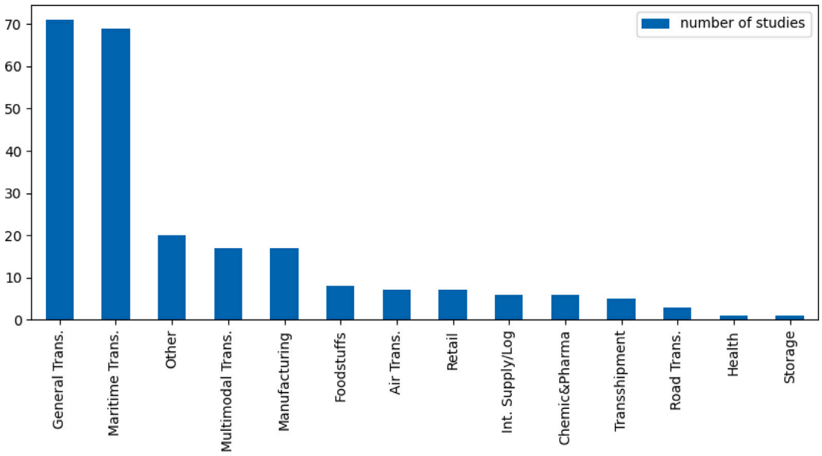


Fig. 11. Number of studies that cover the different domains.

shown in Fig. 11, the top areas of application are *general transport* and *maritime transport* with 71 and 69 studies, respectively. The third area is *other*, a hodgepodge of minority domains, such as military, art, and industry 4.0.

RQ2.3. What type and which business process modelling languages are utilised?

The rationale behind this question is to classify and identify the business process modelling languages utilised in the different studies. Each language has been classified according to its imperative or declarative nature. According to Fahland et al. (2009), imperative process modelling languages focus on the continuous changes of the process' objects, whereas declarative process modelling languages focus on the logic of the actions and objects of a process. Note that sometimes we have extracted the modelling language nature, even though the paper does not mention the concrete modelling language. For example, the study (Mousheimish et al., 2016) does not mention any concrete modelling language; however, we are able to infer the declarative nature of the language. Fig. 12 shows the proportion of business process modelling languages according to their nature. The 89.2% of the business process modelling languages have an imperative nature, and only the 10.2% of them have a declarative nature.

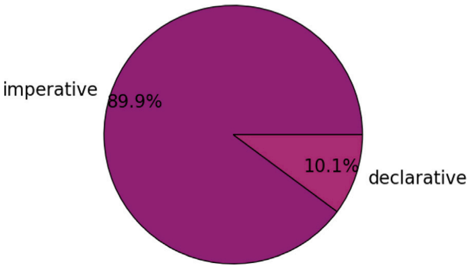


Fig. 12. Studies per business process modelling language type.

Regarding modelling languages, we have found 31 different ones. Fig. 13 shows a bar graph with the number of studies per modelling language. For clarity purposes, the graph only includes the business process modelling languages that have been mentioned in more than one study.[13] BPMN (Business Process Modelling Notation) (OMG, 2013) is by far the winner of the ranking (with 78 studies). BPEL (Business Process Execution Language) (OASIS, 2007) occupies the second place in the ranking (with 14 studies), while Petri Nets (van Hee et al., 2013) occupies the third place (with 11 studies). Finally, note that a study could mention more than one modelling language.

[13] For the reader interested in the ranking of all the languages, see the Appendix.

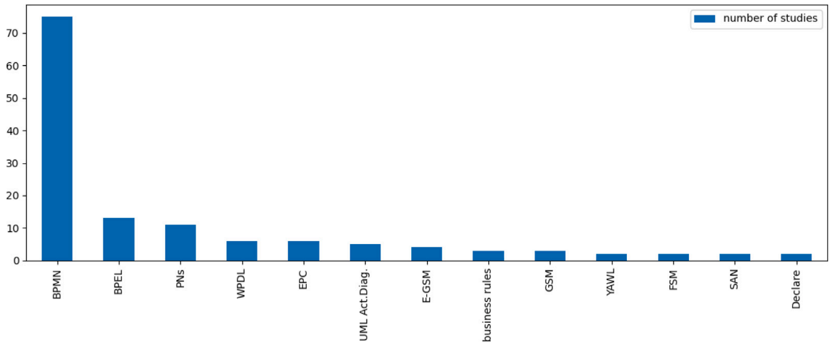


Fig. 13. Modelling languages: PNs = Petri Nets; SAN = Situation-Action-Network; FSM = Finite State Machine.

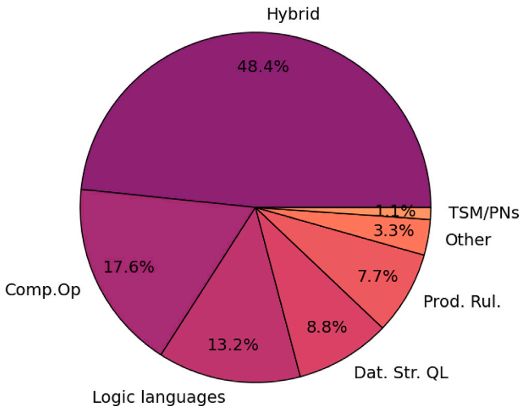


Fig. 14. Studies by event processing language style. Comp. Op. = Based on composition operators; Prod. Rul. = Production rules; Dat. Str. QL = Data stream query language; TSM/PNs = Timed stated machines/Petri nets.

RQ2.4. What type and which event processing languages are utilised?

The rationale behind this question is to classify and identify the event processing languages utilised in the different studies. Each language has been classified according to the five language styles for CEP enumerated in Eckert et al. (2011): languages based on composition operators, data stream query languages, production rules, timed state machines, logic languages, and hybrid approaches.

Languages based on composition operators define complex events using composition operators, such as conjunction of events or a sequence of events. Data stream query languages are usually based on SQL, where events are represented as tuples that flow in data streams, and queries are evaluated on these data streams. Production rules are not really query languages, but they provide a flexible way of implementing event queries. They relate event occurrences to facts, and event queries are expressed as conditions on these facts. Timed-state machines are a formalism in which a system that reacts to events is represented as a graph. Nodes are the states of the system, and edges represent events with associated temporal conditions that change the system state. The state machine implicitly defines the complex events. Logic languages define event queries through logic formulas. Finally, hybrid approaches introduce pattern matching into data stream query languages. Fig. 14 shows a pie chart of the distribution of studies per language style. More than 45% of the studies employ hybrid languages. This could be related to the fact that, by far, the most widely used event processing language is Esper (EsperTech Inc., 2006), which is a hybrid language.

Regarding event processing languages, we have found 33 different languages. Fig. 15 shows a bar graph with the number of studies per language. The graph only includes the event processing languages that have been mentioned in more than one study.[14] EsperTech Inc. (2006) is by far the most widely used EPL, with 31 studies. ECA Rules (Berndtsson and Mellin, 2009) are used in 10 studies. Finally, the third place in the ranking is occupied by SPARQL (W3C, 2013) and event calculus (Shanahan, 1999), with 7 studies each.

RQ3. What are the potential gaps in the area?

This research question helps identify potential gaps in the area. The following subsections discuss the secondary questions in which we have broken it down.

RQ3.1. Is event-driven business process treated in every phase of the process life cycle in a logistics context?

The rationale behind this question is to classify approaches along the life cycle of the process, which, as stated in Leitner and Rinderle-Ma (2014), "has proven to be a viable method to gain a holistic view". As introduced in Section 2, there are different phases in a process life cycle (Dumas et al., 2013): process identification, process discovery, process analysis, process redesign, process implementation, and process monitoring. Fig. 16(a) shows the distribution of studies per phase in the life cycle and Fig. 16(b) shows the distribution of studies per phase and year. One study can cover more than one phase. For example, the study (Mousheimish et al., 2016) covers the phases of analysis and monitoring.

Monitoring and analysis are the phases covered more by the studies. In fact, 43.1% and 33.1% of the studies cover these phases. The redesign phase is treated in 17.7% (30 out of 169) studies. The implementation and discovery phases are covered by 20 and 16 studies, respectively, and, finally, the least covered phase is the identification, with only 8,2% of the studies. Finally, regarding the holistic view, it deserves to be mentioned that only one study (Conforti et al., 2013) covers all the phases.

Regarding the distribution per year of the studies, it can be seen that there are some phases such as monitoring and design that maintain a constant flow of publications over the years. However, there are others, such as identification and implementation, with important publication gaps. It is also interesting to note that the number of publications has decreased drastically in the last two years, in which only the

[14] For the reader interested in the ranking of all the languages, see the Appendix.

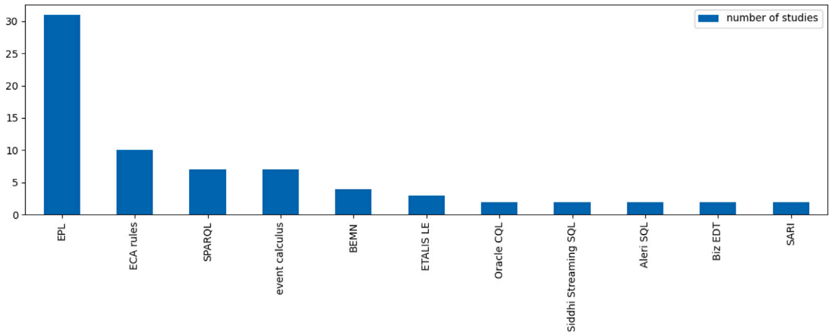


Fig. 15. Event Processing Languages: EPL = Esper Event Processing Language; BEMN = Business Event Modelling Notation; EPN = Event Processing Network; ETALIS LE = ETALIS Language for Events; Biz AL = Business Aware Language.

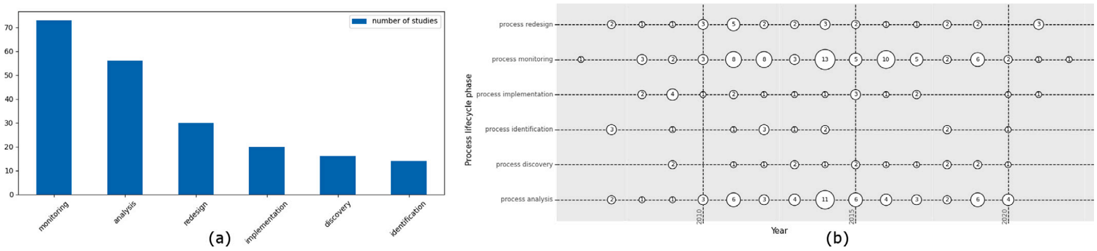


Fig. 16. (a) Number of studies per phase along the process life cycle, and (b) Number of studies per phase in the life cycle and year.

implementation, monitoring, and redesign phases have been covered by the studies. This situation could be due to the COVID pandemic or to a lack of interest in the field.

RQ3.2. What are the challenges taken off in the area?

The rationale behind this question is to identify which of the challenges in the area have been faced. To classify the challenges, we have taken as basis the challenges identified in the Dagstuhl Seminar 16341 on "Integrating Process-Oriented and Event-Based Systems" (Eyers et al., 2016), whose goal was to outline the research challenges in this field. The group of challenges reported in the seminar is listed in Table 9. The table contains an id and a brief description of the topic of the challenges in the group. Fig. 17(a) shows the percentage of studies that address the group of challenges, while Fig. 17(b) shows the distribution over the years of the number of studies per challenge. Note that a study can face more than one challenge.

The most beaten-off challenges are by far those of the CH08-Integrated Platforms for BMP & CEP group with 33.7% (57 out of 169) of the studies. The second place in the ranking (with 39 studies) is the CH03-Automatic Event-based Monitoring of Processes group. The third place is occupied by the CH01-Event Models for BPM group, with 27 studies. In the lowest part of the ranking are the challenges of the groups CH11-Event Data Quality, CH06-Abstraction Levels: placed. Processes versus Event, CH10-Optimisation Opportunities, CH02-Compliance, Audit, Privacy and Security, with 3, 5, 7 and 7 studies, respectively.

The lowest part of the ranking could point out areas that need further research. Thus, if we check if the three main challenges identified in Eyers et al. (2016) for CH11 have been addressed, we can conclude that: no proposal has yet addressed enrichment of process models with the specification of the quality of events yet; only one proposal focusses on making better decisions based on the quality-aware process, and very few approaches focus on defining sources of uncertainty, assessing the quality of events, and translating them into a process-oriented specification.

Regarding the four challenges identified in CH06, none of the analysed proposals addresses the problem of handling unexpected events, and only one of them mentions how to manage integration problems. Furthermore, the analysis of the abstraction levels of processes and events is very limited, and in almost all studies, the relationship of events with process models is established through event logs that represent the input of process discovery algorithms.

In relation to the eight challenges identified in CH10, it should be noted that most of the studies focused on the use of CEP to detect/predict/improve processes at run-time. Only two studies focus on resource allocation for event-driven architectures. But there are three challenges that are not addressed in the identified studies: how information extracted from processes can help optimise complex event processing; distributed query of event streams to improve the performance of BPM; and having a language to handle BPM and CEP that is expressive enough to deal with users (from the point of view of business processes) and, at the same time, efficient for evaluation (from the point of view of CEP).

Finally, of the five challenges identified in CH02, only the one related to information access control and the automation of compliance validation has been addressed. Therefore, challenges such as the processing of audit logs, the use of process models to express policies in event-based systems, and the application of SLAs to CEP engines remain open.

In addition to this, Fig. 18 represents the proportion of proposals that address a given challenge within each phase of the process lifecycle. This helps us to know what type of challenges have been addressed in each of the phases. Thus, it can be seen that CH06 is not addressed in the monitoring phase, leaving open the area of process model and event abstraction analysis within monitoring. In the analysis phase, all

Table 9
Challenges categories.

Id	Challenge category description
CH01	**Event Models for BPM**: This group of challenges is related to how events can be used to process instance adaptation, how the change of event states can influence process instances and how processes can help to give context to events.
CH02	**Compliance, Audit, Privacy and Security**: The challenges in this group are related to the exploitation of CEP to processing audit logs, and BPM tools to express policies in event-based systems and take benefit of the richer access control of BPM.
CH03	**Automatic Event-Based Monitoring of Processes**: This group of challenges are related to the automatic discovery of event patterns to business process monitoring, to monitoring events to guide process adaptation and to use process information to guide the monitoring adaptation.
CH04	**Patterns and Models for Communication**: This group of challenges are related to how the effects of the communication model impose by event-based middlewares are explicitly reflected in process models.
CH05	**Choreographies and Inter-Process Correlation**: This group of challenges is related to the extension of choreography languages to deal with advanced event-based concepts, and to enable the analysis of the information flow between processes.
CH06	**Abstraction Levels. Processes versus Events**: As process models usually follow a top-down approach, whereas event processing follow a bottom-up approach, the challenges in this group are related to find the adequate level of abstraction for a concrete modelling goal and to deal with conflicts in large-scale systems integration.
CH07	**Context in Events and Processes**: The challenges in this group are related to the representation of context, both in processes and in event patterns, to the scoping of context and to the relation of processes and context at runtime.
CH08	**Integrated Platforms for BPM & CEP**: The challenges in this group are related to the integration of BPM and CEP platforms, which also involve the development of a unified model for events and processes.
CH09	**Distributed Processes & The Role of Events**: The challenges in this group are related to event loss, to misdetection of complex events, to analysis of stream events in real time, to deal with privacy in the context of event and processes handled in a centralised or distributed sources.
CH10	**Optimisation opportunities**: The challenges in this group are related to the exploitation of BPM to improve event processing, and the other way around, the exploitation of CEP methods to improve processes.
CH11	**Event Data Quality**: The challenges in this group are related to making explicit uncertainties, to make business models aware of data quality and to how this quality-aware model influences decision making.
CH12	**From Event Streams to Process Models and Back**: The challenges in this group are related to automate the generation of CEP rules from business process monitoring, to use CEP constructs for process mining, to enrich the expressiveness of process models with CEP constructs and to execute business processes via CEP rules.

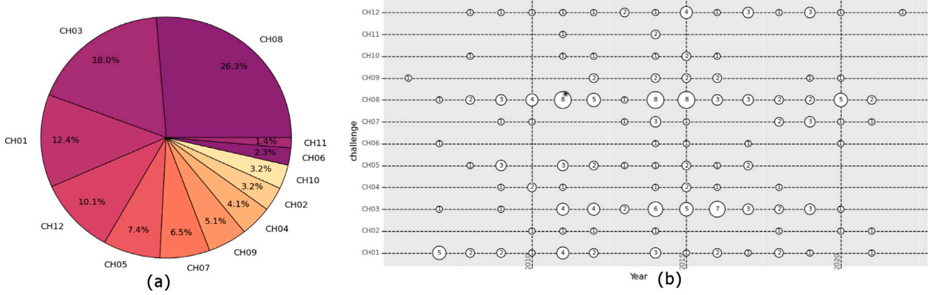


Fig. 17. (a) Percentage of challenges by number of studies, and (b) Number of studies per challenge and year.

the challenges are covered, while in the implementation phase, CH10 and CH11 are not addressed.

At the redesign phase, CH02, CH06, CH09 and CH11 are not addressed, which indicates that issues such as compliance, levels of abstraction, distributed processes and event quality needs further research. Finally, there are still many challenges open in the identification and discovery phases (CH04, CH06, CH07, CH09, CH10 and CH11).

7. Threats to validity

To ascertain the validity of the results obtained in our mapping study, we have used as a checklist the list of threats to validity proposed by Wohlin et al. (2012). As standardised in software engineering, we will introduce threats to validity by grouping them into the four categories proposed by Cook and Campbell (1979): construct, internal, external, and conclusion validity.

The threats in **construct validity** are concerned with generalising the results of the experiment to the concept or theory behind it (Wohlin et al., 2012). There are two kinds of threat in this category: design threats that are related to the design of the experiment and social threats that are concerned with the behaviour of the subjects and the experimenters.

Some design threats that could affect the validity of this study are related to the suitability of the research questions, the inclusion and exclusion criteria defined to select the primary studies, and the classification scheme used for data extraction. To mitigate the threat related to the research questions, we follow the guidelines proposed in Kitchenham et al. (2015), Kuhrmann et al. (2017) and Petersen et al. (2015) to design our research questions, and we use some of the research questions that are most frequently addressed by systematic mapping studies (Khan et al., 2019). The inclusion and exclusion criteria threat is also mitigated by using the standard inclusion and exclusion criteria proposed in Kuhrmann et al. (2017). In relation to the categorisation scheme, we faced this threat by using taxonomies and categorisations published in relevant references to classify the selected primary studies. Finally, it also deserves to be noticed that we have carefully documented the whole process using the Start tool.

Regarding social threats, this mapping study could suffer from the experimenter expectancy; the experimenters can bias the results of the study based on their expectancies from the experiment. To mitigate this threat, each study is analysed by at least two researchers. For example, if a researcher is responsible for the selection of a study in the selection stage, a different researcher is responsible for the data extraction of that study. If unclear questions arise about a concrete study, a third researcher breaks the deadlock. In addition, some results of selection

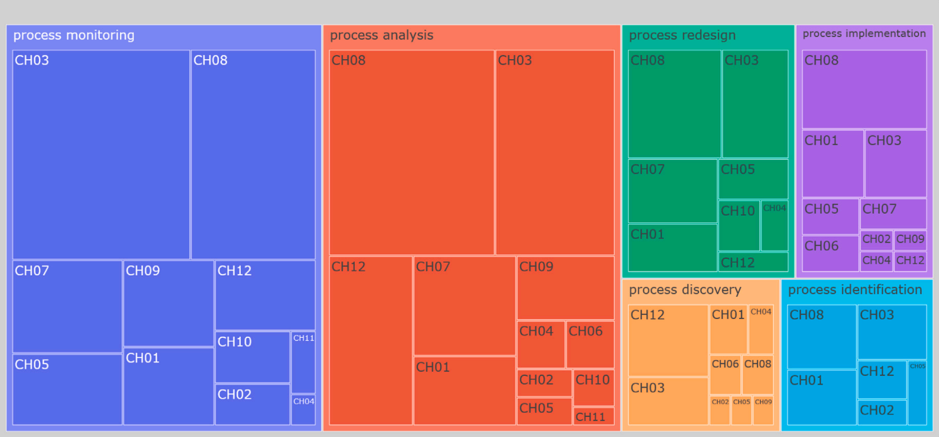


Fig. 18. Relationship between challenges and process lifecycle phases.

and extraction were discussed by the four researchers in weekly meetings. Furthermore, the different tasks have been assigned as follows: The four authors have classified the studies. The first screening and careful reading have not been performed for the same researcher in the same paper. Regarding the rest of the activities involved in the study selection, Author 2 was in charge of collecting studies from data sources (search engines and snowballing).

Internal validity checks the reliability of the results of a study in terms of how well the study is conducted. To face this kind of threat, we have followed a systematic approach (Kitchenham and Charters, 2007; Petersen et al., 2008). On the one hand, we performed a formal automatic search in six different online digital databases and search engines that are the most commonly used in other systematic mapping studies (Khan et al., 2019). On the other hand, an automatic backward and forward snowballing process has been included to incorporate the references included in selected studies. Finally, in relation to the bias that could be introduced by applying the inclusion/exclusion criteria, it has been partially faced due to the fact that different researchers test the decision of the researcher who has been responsible for the study at a previous stage.

The threats in **external validity** are concerned with the condition that limits the possible generalisation of the results and the interest of other people outside the review. To mitigate the threats in this category, we have prepared a bundle with all datasets produced during our analysis and we have publicly available the scripts that we used to synthesise the extracted data (see Section 4.6). Hence, researchers who wish to replicate or extend our material have all the material at their disposal. However, a threat that remains in this study is related to access to the whole set of selected studies. Although our institution maintains subscriptions to every digital library we have queried in this study, there may be researchers who have no access to those libraries.

Finally, **conclusion validity** is related to drawing the correct conclusions and to being reproducible for other researchers. In this mapping study, all graphs, charts, and tables are generated from datasets. On the one hand, datasets are publicly available, and, on the other hand, the Python scripts used to generate them are also available in a GitHub repository[15] and as a Jupyter notebook. Hence, our results are completely traceable. Data from primary studies have not only been extracted carefully (as explained in Section 4.5), but have also been reviewed by at least two different researchers. Another point that could threaten validity is the scheme used to score the quality of the studies. To mitigate this threat, a framework commonly used in the field of data quality has been used.

8. Conclusions

Logistics processes are receiving substantial mainstream attention because of the recent logistics crisis. These processes involve the choreography of various entities and produce an enormous number of complex events. The analysis of the state-of-the-art of event-driven business management approaches in logistics can help not only organisations with logistics management that want to leverage the level of digitalisation of their processes, but also researchers to foresee new opportunities of research by revealing open problems. However, there are no secondary studies focused on the analysis of approaches that integrate business processes and complex events in the logistics domain. To bridge this gap, this paper reports the results of a systematic mapping study that analyses and classifies the selected studies according to different criteria, such as the type of contribution they provide, their area of application, the logistics coverage, the business process modelling language they used to model processes, the event processing language they used to process complex events, the process life cycle they cover, and the open challenges. The main conclusions of this study can be drawn from the perspectives of companies and academia.

From the company's point of view, it is important to count on a variety of mature frameworks and tools that support the deployment of a solution in a real scenario. However, after analysing all selected articles, we can state that there is no framework or tool that companies can apply directly to manage the integration of business processes and complex event processing in logistics environments. More than 51% of the proposed solutions are proofs of concept, prototypes, or solutions entirely oriented toward academic research, which makes their application to a real scenario difficult. For this reason, companies must develop made-by-measure solutions for adapting the technologies they use to manage processes for taking the advantages that complex event processing could provide. This means that most of the proposals are still open for further analysis, improvement, and extension. In particular, it could be interesting to continue improving those tools that cover various phases of the life cycle to cover all phases.

From the academic point of view, the most interesting findings are those related to the less mature areas, because they could provide opportunities for new research lines. Regarding the phases of the life cycle, only 5% of the studies cover the discovery phase and approximately

[15] https://github.com/reinaqu/reportingslroo.

the 7% cover the identification phase. There is also a lack of approaches that cover the entire life cycle from the first phase to have a holistic view. In relation to business process modelling languages, only slightly more than 11% of the studies use a declarative language style. This is important in those cases in which the application of the government regulations describe the order of the actions allowed or prohibited, but not always specify the exact sequence. An example of this kind of regulation in the logistics context is "before the entrance of a boat into a port, security documentation should have been verified". Finally, there still are some challenges that remains open, above all those ones related to event data quality (CH11), abstraction levels (CH06), optimisation opportunities (CH10), and compliance, audit, privacy and security (CH02). For example, we have found that no proposal has yet addressed the enrichment of business process models with the specification of the quality of events or that there is a lack of approaches that deal with the processing of audit logs or translation of laws into security policies in this context.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

We have share data in mendeley and github through respective links.

Acknowledgements

This work has been funded by the projects AETHER-US (PID2020-112540RB-C44/AEI/10.13039/501100011033), COPERNICA (P20_01 224), METAMORFOSIS (US-1381375) and FEDER funds through programme Interreg V-A España-Portugal (POCTEP) 2014–2020.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.compind.2022.103788.

References

Akila, V., Govindasamy, V., Sandosh, S., 2016. Complex event processing over uncertain events: Techniques, challenges, and future directions. In: ICCPEIC. pp. 204–221. http://dx.doi.org/10.1109/ICCPEIC.2016.7557198.

Alias, C., Rawet, V.L., Neto, H.X.R., do Egypto Neirão Reymão, J., 2016. Investigating into the prevalence of complex event processing and predictive analytics in the transportation and logistics sector: Initial findings from scientific literature. In: MCIS. University of Nicosia / AISeL, p. 2.

Amjad, A., Azam, F., Anwar, M.W., Butt, W.H., Rashid, M., 2018. Event-driven process chain for modeling and verification of business requirements-A systematic literature review. IEEE Access 6, 9027–9048. http://dx.doi.org/10.1109/ACCESS.2018.2791666.

Ammon, R.v., 2009a. Event-driven business process management. In: Liu, L., Özsu, M.T. (Eds.), Encyclopedia of Database Systems. Springer US, Boston, MA, pp. 1068–1071. http://dx.doi.org/10.1007/978-0-387-39940-9_577.

Ammon, R.V., 2009b. Event-driven business process management. In: Encyclopedia of Database Systems.

von Ammon, R., 2018. Event-driven business process management. In: Liu, L., Özsu, M.T. (Eds.), Encyclopedia of Database Systems. Springer New York, New York, NY, pp. 1399–1403. http://dx.doi.org/10.1007/978-1-4614-8265-9_577.

Ammon, R.v., Emmersberger, C., Springer, F., Wolff, C., 2008. Event-driven business process management and its practical application taking the example of DHL.

Augusto, A., 2020. Accurate and Efficient Discovery of Process Models from Event Logs (Ph.D. thesis). University of Melbourne and University of Tartu, Melbourne.

Augusto, A., Conforti, R., Dumas, M., Rosa, M.L., Maggi, F.M., Marrella, A., Mecella, M., Soo, A., 2019. Automated discovery of process models from event logs: Review and benchmark. IEEE Trans. Knowl. Data Eng. 31 (4), 686–705. http://dx.doi.org/10.1109/TKDE.2018.2841877.

Bassil, S., Keller, R.K., Kropf, P.G., 2004. A workflow-oriented system architecture for the management of container transportation. In: Business Process Management. In: Lecture Notes in Computer Science, vol. 3080, Springer, pp. 116–131. http://dx.doi.org/10.1007/978-3-540-25970-1_8.

Berndtsson, M., Mellin, J., 2009. ECA rules. In: Liu, L., Özsu, M.T. (Eds.), Encyclopedia of Database Systems. Springer US, Boston, MA, pp. 959–960. http://dx.doi.org/10.1007/978-0-387-39940-9_504.

Broadman, H.G., 2021. Global supply chains' crisis is much bigger than the pandemic; the transformation they're undergoing is the cure.

Chandy, M.K., Etzion, O., von Ammon, R., 2011. 10201 Executive summary and manifesto – event processing. In: Chandy, K.M., Etzion, O., von Ammon, R. (Eds.), Event Processing. In: Dagstuhl Seminar Proceedings (DagSemProc), vol. 10201, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, pp. 1–60. http://dx.doi.org/10.4230/DagSemProc.10201.1.

Chow, H.K.H., Choy, K.L., Lee, W.B., Chan, F.T.S., 2005. Design of a knowledge-based logistics strategy system. Expert Syst. Appl. 29 (2), 272–290. http://dx.doi.org/10.1016/j.eswa.2005.04.001.

Conforti, R., Rosa, M.L., Fortino, G., ter Hofstede, A.H.M., Recker, J., Adams, M., 2013. Real-time risk monitoring in business processes: A sensor-based approach. J. Syst. Softw. 86 (11), 2939–2965. http://dx.doi.org/10.1016/j.jss.2013.07.024.

Cook, T.D., Campbell, D.T., 1979. Quasi-Experimentation: Design & Analysis Issues for Field Settings. Houghton Mifflin Company.

Cugola, G., Margara, A., Matteucci, M., Tamburrelli, G., 2015. Introducing uncertainty in complex event processing: model, implementation, and validation. Computing 97 (2), 103–144. http://dx.doi.org/10.1007/s00607-014-0404-y.

Davenport, T., 1993. Process Innovation: Reengineering Work through Information Technology. Harvard Business School Press, Boston.

Davidsson, P., Henesey, L., Ramstedt, L., Törnquist, J., Wernstedt, F., 2005. Agent-based approaches to transport logistics. In: Klügl, F., Bazzan, A.L.C., Ossowski, S. (Eds.), Applications of Agent Technology in Traffic and Transportation. In: Whitestein series in software agent technologies and autonomic computing, Springer, pp. 1–15. http://dx.doi.org/10.1007/3-7643-7363-6_1.

Dayarathna, M., Perera, S., 2018. Recent advancements in event processing. ACM Comput. Surv. 51 (2), 33:1–33:36. http://dx.doi.org/10.1145/3170432.

Dumas, M., Rosa, M.L., Mendling, J., Reijers, H.A., 2013. Fundamentals of Business Process Management. Springer Publishing Company, Incorporated.

Eckert, M., Bry, F., Brodt, S., Poppe, O., Hausmann, S., 2011. A CEP babelfish: Languages for complex event processing and querying surveyed. In: Helmer, S., Poulovassilis, A., Xhafa, F. (Eds.), Reasoning in Event-Based Distributed Systems. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 47–70. http://dx.doi.org/10.1007/978-3-642-19724-6_3.

Emmersberger, C., Springer, F., Wolff, C., 2009. Location based logistics services and event driven business process management. In: IMC. In: Communications in Computer and Information Science, vol. 53, pp. 167–177. http://dx.doi.org/10.1007/978-3-642-10263-9_15.

EsperTech Inc., 2006. Pluto: The 'other' red planet. https://www.espertech.com/esper/, Accessed: 2021-11-24.

Even, A., Shankaranarayanan, G., Berger, P.D., 2010. Evaluating a model for cost-effective data quality management in a real-world CRM setting. Decis. Support Syst. 50 (1), 152–163. http://dx.doi.org/10.1016/j.dss.2010.07.011.

Eyers, D.M., Gal, A., Jacobsen, H., Weidlich, M., 2016. Integrating process-oriented and event-based systems (Dagstuhl seminar 16341). Dagstuhl Rep. 6 (8), 21–64. http://dx.doi.org/10.4230/DagRep.6.8.21.

Fahland, D., Mendling, J., Reijers, H.A., Weber, B., Weidlich, M., Zugal, S., 2009. Declarative versus imperative process modeling languages: The issue of maintainability. In: Business Process Management Workshops. In: Lecture Notes in Business Information Processing, vol. 43, Springer, pp. 477–488. http://dx.doi.org/10.1007/978-3-642-01862-6_29.

Force, U.R.T., 2001. Omg unified modeling language specification, version 1.4 (final draft).

Fulop, L.J., Tóth, G., Rácz, R., Pánczél, J., Gergely, T., Beszédes, Á., 2010. Survey on Complex Event Processing and Predictive Analytics. Tech. Rep., University of Szeged, Department of Software Engineering.

Future, M.R., 2021. Business Process Management Market. Market Research Report.

Ghiani, G., Laporte, G., Musmanno, R., 2013. Introduction to Logistics Systems Management. In: Wiley Series in Operations Research and Management Science, Wiley.

Goetz, D.M., 2010. Integration of business process management and complex event processing.

Gómez-López, M.T., Gasca, R.M., Pérez-Álvarez, J.M., 2015. Compliance validation and diagnosis of business data constraints in business processes at runtime. Inf. Syst. 48, 26–43. http://dx.doi.org/10.1016/j.is.2014.07.007.

GROBID, 2008–2021. GROBID. arXiv:1:dir:dab86b296e3c3216e2241968f0d63b68e8209d3c https://github.com/kermitt2/grobid.

Hammer, M., Champy, J., 1993. Reengineering the Corporation: A Manifesto for Business Revolution. Harper Business.

van Hee, K.M., Sidorova, N., van der Werf, J.M.E.M., 2013. Business process modeling using Petri nets. Trans. Petri Nets Other Model. Concurr. 7, 116–161. http://dx.doi.org/10.1007/978-3-642-38143-0_4.

Hull, R., Damaggio, E., Fournier, F., Gupta, M., Heath, F.T., Hobson, S., Linehan, M., Maradugu, S., Nigam, A., Sukaviriya, P., et al., 2010. Introducing the guard-stage-milestone approach for specifying business entity lifecycles. In: International Workshop on Web Services and Formal Methods. Springer, pp. 1–24.

Jaekel, F., 2019. Systematic review of cloud logistics knowledge. In: Cloud Logistics: Reference Architecture Design. Springer Fachmedien Wiesbaden, Wiesbaden, pp. 191–256. http://dx.doi.org/10.1007/978-3-658-22837-8_5.

Johansson, H.J., McHugh, P., Pendlebury, A.J., Wheeler, W.A., 1993. Business Process Reengineering: Breakpoint Strategies for Market Dominance. John Wiley & Sons.

Khan, M.U., Sherin, S., Iqbal, M.Z., Zahid, R., 2019. Landscaping systematic mapping studies in software engineering: A tertiary study. J. Syst. Softw. 149, 396–436. http://dx.doi.org/10.1016/j.jss.2018.12.018.

Kitchenham, B.A., Budgen, D., Brereton, O.P., 2010. The value of mapping studies - A participant-observer case study. In: EASE. In: Workshops in Computing, BCS.

Kitchenham, B.A., Budgen, D., Brereton, P., 2015. Evidence-Based Software Engineering and Systematic Reviews, first ed. CRC Press, An optional note.

Kitchenham, B., Charters, S., 2007. Guidelines for performing Systematic Literature Reviews in Software Engineering. Tech. Rep. EBSE-2007-01, School of Computer Science and Mathematics, Keele University.

Klappich, D., Muynck, B.D., Aimi, G., Titze, C., Stevens, A., 2020. Cpredicts 2021: Supply chain technology.

Krumeich, J., Weis, B., Werth, D., Loos, P., 2014. Event-driven business process management: where are we now?: A comprehensive synthesis and analysis of literature. Bus. Process. Manage. J. 20 (4), 615–633. http://dx.doi.org/10.1108/BPMJ-07-2013-0092.

Kuhrmann, M., Fernández, D.M., Daneva, M., 2017. On the pragmatic design of literature studies in software engineering: an experience-based guideline. Empir. Softw. Eng. 22 (6), 2852–2891. http://dx.doi.org/10.1007/s10664-016-9492-y.

Leitner, M., Rinderle-Ma, S., 2014. A systematic review on security in Process-Aware Information Systems - Constitution, challenges, and future directions. Inf. Softw. Technol. 56 (3), 273–293. http://dx.doi.org/10.1016/j.infsof.2013.12.004.

Li, C.-S., 2005. Real-time event driven architecture for activity monitoring and early warning. In: Conference, Emerging Information Technology 2005. p. 4 pp.. http://dx.doi.org/10.1109/EITC.2005.1544382.

Li, T., Zeng, C., Jiang, Y., Zhou, W., Tang, L., Liu, Z., Huang, Y., 2017. Data-driven techniques in computing system management. ACM Comput. Surv. 50 (3), 45:1–45:43. http://dx.doi.org/10.1145/3092697.

Linden, I., Derbali, M., Schwanen, G., Jacquet, J., Ramdoyal, R., Ponsard, C., 2013. Supporting business process exception management by dynamically building processes using the BEM framework. In: EWG-DSS. In: Lecture Notes in Business Information Processing, vol. 184, Springer, pp. 67–78. http://dx.doi.org/10.1007/978-3-319-11364-7_7.

Luckham, D.C., 2005. The Power of Events - An Introduction to Complex Event Processing in Distributed Enterprise Systems. ACM.

Luckham, D., 2008. The power of events: An introduction to complex event processing in distributed enterprise systems. In: RuleML. In: Lecture Notes in Computer Science, vol. 5321, Springer, p. 3. http://dx.doi.org/10.1007/978-3-540-88808-6_2.

Luckham, D., 2012. Event Processing for Business. In: Organizing the Real-Time Enterprise, Wiley.

Mousheimish, R., Taher, Y., Zeitouni, K., 2016. The butterfly: An intelligent framework for violation prediction within business processes. In: IDEAS. ACM, pp. 302–307. http://dx.doi.org/10.1145/2938503.2938541.

OASIS, 2007. Web Services Business Process Execution Language Version 2.0. Tech. Rep., OASIS.

OMG, 2013. Business Process Model and Notation (BPMN), Version 2.0.2. Tech. Rep., Object Management Group.

Pérez-Álvarez, J.M., Maté, A., Gómez-López, M.T., Trujillo, J., 2018. Tactical business-process-decision support based on KPIs monitoring and validation. Comput. Ind. 102, 23–39. http://dx.doi.org/10.1016/j.compind.2018.08.001.

Pesic, M., Van der Aalst, W.M., 2006. A declarative approach for flexible business processes management. In: International Conference on Business Process Management. Springer, pp. 169–180.

Petersen, K., Feldt, R., Mujtaba, S., Mattsson, M., 2008. Systematic mapping studies in software engineering. In: EASE. In: Workshops in Computing, BCS.

Petersen, K., Vakkalanka, S., Kuzniarz, L., 2015. Guidelines for conducting systematic mapping studies in software engineering: An update. Inf. Softw. Technol. 64, 1–18. http://dx.doi.org/10.1016/j.infsof.2015.03.007.

Ramos-Gutiérrez, B., Reina Quintero, A., Parody Núñez, M., Gómez López, M., 2021. When business processes meet complex events in logistics: A systematic mapping study'', mendeley data. http://dx.doi.org/10.17632/xwy9djwnh4.1.

Schiefer, J., McGregor, C., 2004. Correlating events for monitoring business processes. In: ICEIS (1). pp. 320–327.

Schiefer, J., Roth, H., Suntinger, M., Schatten, A., 2007. Simulating business process scenarios for event-based systems. In: ECIS. University of St. Gallen, pp. 1729–1740.

Shanahan, M., 1999. The event calculus explained. In: Artificial Intelligence Today. Springer, pp. 409–430.

Soffer, P., Hinze, A., Koschmider, A., Ziekow, H., Ciccio, C.D., Koldehofe, B., Kopp, O., Jacobsen, H., Sürmeli, J., Song, W., 2019. From event streams to process models and back: Challenges and opportunities. Inf. Syst. 81, 181–200. http://dx.doi.org/10.1016/j.is.2017.11.002.

Ter Hofstede, A.H., Van der Aalst, W.M., Adams, M., Russell, N., 2009. Modern Business Process Automation: YAWL and its Support Environment. Springer Science & Business Media.

Varela-Vaca, Á.J., Reina Quintero, A.M., 2021. Smart contract languages: A multivocal mapping study. ACM Comput. Surv. 54 (1), 3:1–3:38. http://dx.doi.org/10.1145/3423166.

W3C, 2013. SPARQL 1.1 Query Language. Tech. Rep., W3C.

Wang, R.Y., Strong, D.M., 1996. Beyond accuracy: What data quality means to data consumers. J. Manage. Inf. Syst. 12 (4), 5–33. http://dx.doi.org/10.1080/07421222.1996.11518099.

Weske, M., 2012. Business Process Management - Concepts, Languages, Architectures, second ed. Springer.

Wohlin, C., 2014. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In: EASE. ACM, pp. 38:1–38:10. http://dx.doi.org/10.1145/2601248.2601268.

Wohlin, C., Runeson, P., Höst, M., Ohlsson, M.C., Regnell, B., 2012. Experimentation in Software Engineering. Springer, http://dx.doi.org/10.1007/978-3-642-29044-2.

Belén Ramos Gutiérrez, (Phd. Student) Universidad de Sevilla, Dpto. Lenguajes y sistemas informáticos – Spain.

Belén Ramos-Gutiérrez is currently a Software Engineering and Technology Ph.D. Student at the University of Seville. Her research interests include process mining, data extraction, and optimisation for process mining techniques; and troubleshooting and decision support systems in industrial environments. She is also working as a predoctoral researcher with the University of Seville and collaborates on projects involving industrial aspects related to logistics-port and aeronautical environments. Her research goal is to improve and automate the extraction and processing of heterogeneous data from industrial environments for exploitation with process mining techniques.

Antonia M. Reina Quintero, (Assistant Professor) Universidad de Sevilla, Dpto. Lenguajes y sistemas informáticos – Spain.

Antonia M. Reina Quintero obtained her Ph.D. with honours in Computer Engineering from the University of Seville (2012). She has worked as a full-time lecturer at the Computer Languages and Systems Department from the University of Seville since 2000, although she also has worked as a computer engineer for a leading company in traffic control systems. Her current research is focused on advanced separation of concerns, Model-Driven Engineering applied to business processes, and systematic literature reviews in software engineering. She has participated in public research projects and prestigious conferences. She has published several high-impact papers.

Luisa Parody, (Associate Professor), Universidad Loyola Andalucía, Sevilla, Spain.

Luisa Parody studied Computer Engineering (including a minor in Systems Engineering) at the Universidad de Sevilla (Spain) and graduated with honours in July 2009. She then earned an M.Sc. Degree in Software Engineering and Technology(2010) and obtained her international Ph.D. with honours at the Universidad Sevilla (2014). Since 2018, she has been working as an associate professor at Dpto. Métodos Cuantitativos at the Universidad Loyola Andalucía. She belongs to the IDEA Research Group and has participated in several private and public research projects. She was nominated as a member of Program Committees, such as ISD and SIMPDA. She has participated in prestigious conferences and has published several high-impact papers.

María Teresa Gómez-López is a Full Professor at the University of Seville and the head of the IDEA Research Group (www.idea.us.es). Her research areas include Business Processes and Data management, and how to improve the business process models including better decisions and enriching the model with Data Perspectives. She has led several private and public competitive research projects and has published more than 30 articles in high-quality conferences and journals. She was nominated as a member of Program Committees, such as ER, BPM, EDOC, ISD or CAISE Doctoral Consortium, and she has given various keynotes or was invited speaker in international forums.

3.2.2 Discovering configuration workflows from existing logs using process mining

Published in Empirical Software Engineering Vol. 26 (1), 2021, Springer

- ***Authors**: Belén Ramos Gutiérrez, Ángel Jesús Varela-Vaca, José A. Galindo, María Teresa Gómez-López, David Benavides.*
- ***DOI**:* `https://doi.org/10.1007/s10664-020-09911-x`.
- ***Rating**: Q1 (JIF'21 3.762).*

Empirical Software Engineering (2021) 26: 11
https://doi.org/10.1007/s10664-020-09911-x

Discovering configuration workflows from existing logs using process mining

Belén Ramos-Gutiérrez[1] · Ángel Jesús Varela-Vaca[1] · José A. Galindo[1] · María Teresa Gómez-López[1] · David Benavides[1]

Accepted: 30 October 2020 / Published online: 26 January 2021

Abstract

Variability models are used to build configurators, for guiding users through the configuration process to reach the desired setting that fulfils user requirements. The same variability model can be used to design different configurators employing different techniques. One of the design options that can change in a configurator is the configuration workflow, i.e., the order and sequence in which the different configuration elements are presented to the configuration stakeholders. When developing a configurator, a challenge is to decide the configuration workflow that better suits stakeholders according to previous configurations. For example, when configuring a Linux distribution the configuration process starts by choosing the network or the graphic card and then, other packages concerning a given sequence. In this paper, we present COnfiguration workfLOw proceSS mIning (COLOSSI), a framework that can automatically assist determining the configuration workflow that better fits the configuration logs generated by user activities given a set of logs of previous configurations and a variability model. COLOSSI is based on process discovery, commonly used in the process mining area, with an adaptation to configuration contexts. Derived from the possible complexity of both logs and the discovered processes, often, it is necessary to divide the traces into small ones. This provides an easier configuration workflow to be understood and followed by the user during the configuration process. In this paper, we apply and compare four different techniques for the traces clustering: greedy, backtracking, genetic and hierarchical algorithms. Our proposal is validated in three different scenarios, to show its feasibility, an ERP configuration, a Smart Farming, and a Computer Configuration. Furthermore, we open the door to new applications of process mining techniques in different areas of software product line engineering along with the necessity to apply clustering techniques for the trace preparation in the context of configuration workflows.

Keywords Variability · Configuration workflow · Process mining · Process discovery · Clustering

Communicated by: Laurence Duchien, Thomas Thüm and Paul Grünbacher

This article belongs to the Topical Collection: *Configurable Systems*

✉ Belén Ramos-Gutiérrez
brgutierrez@us.es

Extended author information available on the last page of the article.

1 Introduction

Variability models, such as Feature Models (FMs) (Galindo et al. 2018), describe commonalities and variabilities in Software Product Lines (SPLs) and are used along all the SPL development process. After an FM is defined, products can be configured and derived. We can find FM depicting a diverse set of domains such as Wordpress (Rodas-Silva et al. 2019), surveillance videos (Galindo et al. 2014b; Alférez et al. 2019) or Android systems (Galindo et al. 2014a) among others. In the configuration and derivation process, users select and deselect features using a *configurator*. A configurator (Galindo et al. 2015) is a software tool that presents configuration options to the users in different stages. An example of a configurator tool is KConfig (She et al. 2010) where developers can configure the Linux kernel with more than 12.000 configuration options.

An important aspect of a configurator is to determine the *configuration workflow* (Hubaux et al. 2013), i.e., the order in which features and options are presented to configuration stakeholders. For instance, when configuring the Linux kernel using KConfig (She et al. 2010), there can be different user configuration profiles depending on interests or skills. The configuration workflow used by a configurator can impact the user experience in the configuration process. Therefore, selecting a well-suited configuration workflow is a challenge. Up to now –to the best of our knowledge– the selection of a configuration workflow is made either intuitively or following the structure and properties of a variability model (Galindo et al. 2015; Varela-Vaca and Gasca 2013; Varela-Vaca et al. 2019b).

In this paper, we present COLOSSI, a framework that takes a feature model and a set of existing configuration logs and automatically retrieves configuration workflows. A configuration log is a set of configurations performed in the past in a given domain taking into account a configuration order. Our solution relies on *process mining* (Augusto et al. 2019) techniques. Process mining is a well-established area of business process management that uses different techniques to extract business processes from traces of execution. In our approach, we conceptually map a business process model to a configuration workflow and traces to configuration logs making it possible to reuse process mining techniques to infer configuration workflows.

Although using process mining can automatically retrieve configuration workflows, the results can be difficult to interpret to domain engineers to build a configurator. This is mainly because, very often, mined processes are “spaghetti-like” models in which the same activity needs to be duplicated (van der Aalst 2011). To illustrate the difficulty, Fig. 1 shows the result of directly applying process mining techniques to the ERP system presented in Pereira et al. (2018b) and detailed in Section 3.

The simplification of spaghetti processes is an open challenge and active research area in process mining (Augusto et al. 2019). The application of clustering techniques is used to retrieve a set of simple workflows instead of a single–complicated one. These techniques are used to divide the configuration log according to different aspects. There are different techniques to create clusters that facilitate the understanding of the discovered processes. The clustering can be expressed as an optimisation problem. Thus, exhaustive techniques have been proposed for finding the best possible clustering (Hompes et al. 2015), although with potential high time- and resource-consuming. This is the reason why other algorithms have been applied, such as hierarchical algorithm (Ferreira and Alves 2011; Makanju et al. 2008, 2009), k-means algorithm (Song et al. 2008) or greedy algorithm (Greco et al. 2006;

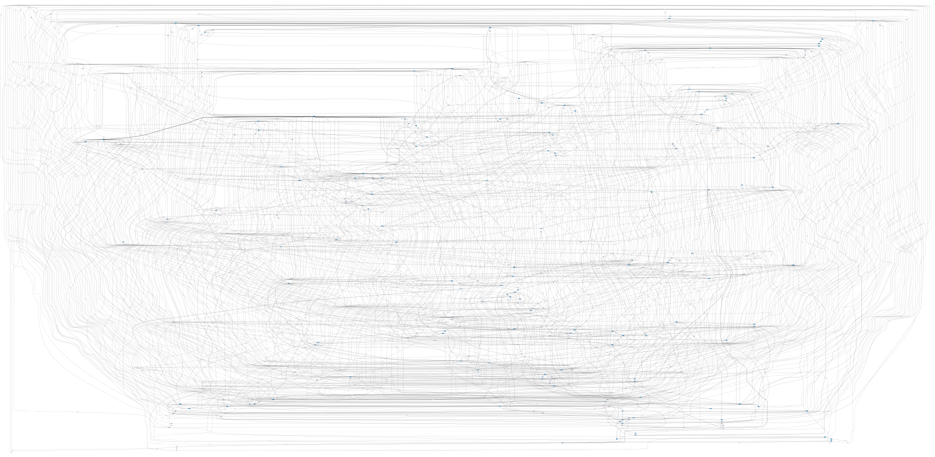


Fig. 1 Spaghetti process of the ERP presented in Pereira et al. (2018b)

De Weerdt et al. 2013). The wide types of possible algorithms to apply are caused by the most proper in each case will depend on the data log. To overcome this difficulty, COLOSSI provides the infrastructure to integrate various algorithms for clustering the traces involved in the configuration workflow, being possible the selection of the used algorithm according to the case and the necessities. Our solution takes information from the variability model as input and retrieves less complex configuration workflows that can assist the development of better configurators. Due to the size and complexity of the logs, the creation of clusters that facilitate the understanding of the discovered processes is not an easy task. Thereby, we propose four approximations based on different techniques to facilitate the creation of clusters trying to improve the distribution of the logs later used as input of the process discovery.

COLOSSI is validated using three different case studies: an ERP system, a smart farm and computer configurations taken from Pereira et al. (2016a, 2018a, b). Moreover, different clustering algorithms are applied to determine when each is more appropriate (greedy, backtracking, genetic and hierarchical algorithms). Results show that the metrics of the retrieved configuration workflows are improved depending on the clustering and the distribution algorithm of traces applied. Besides, the metrics help to support the hypotheses of reducing the complexity and enhancing the understandability of the retrieved configuration workflows. Moreover, an statistical analysis of metrics related to the distribution of traces is done to attempt to prove that the hypotheses that the selected techniques do not influence the obtained results (i.e., the algorithm has no impact on the distribution of traces).

This paper is an evolution of Varela-Vaca et al. (2019a) where the main contribution was the definition of COLOSSI as a framework to support the clustering of configuration logs to discover configuration workflows. In this paper, the previous proposal is enhanced by:

- The extension of COLOSSI to support different sort of clustering algorithms to apply.
- We extended the validation of COLOSSI with three additional case studies.

- The measurement of a set of metrics to compare the suitability of each algorithm of clustering to discover the configuration workflows that better fits the configuration traces generated by the users.

The remainder of this paper is organised as follows: Section 2 details the solution and concepts that grounds our proposal; Section 3 presents empirical results from analysing COLOSSI in different scenarios; Section 4 presents the related work and Section 5 presents concluding remarks and lessons learned.

2 COLOSSI: configration workflow process mining solution

COLOSSI combines a feature model and a configuration log to create a configuration workflow. Figure 2 shows an overview of the framework, in which, using a configuration log, it is possible to apply process mining techniques to derive a valid configuration workflow representing all the possible paths defined in the configuration logs. Very often, the resulting workflow follows the so-called spaghetti-style (van der Aalst 2011), being difficult to understand and manipulate. Nevertheless, it is important to remark that by applying some simplification techniques and extracting some metrics, these workflows could be exploited using automated process mining tools, to carry out many more additional analyses. Also, any generated configuration workflow can be already used to automatically build a configurator.

In an ideal process mining approach, as shown in path ① of Fig. 2, an event log is used to directly execute a process discovery task. However, in general, if no data preparation task is carried out in between, the results are usually unmanageable, and even more, when we are dealing with a context of high variability. That is why the data preparation phase in our proposal must necessarily include simplification mechanisms. To this end, also, to the usage of process mining techniques, we propose diverse handling and clustering methods to reduce and group similar configuration traces according to some properties. Those clusters can then be used again as an input of process mining techniques to obtaining a set of configuration workflows depending on the observed behaviour of the configuration logs. Those workflows could obtain better metrics concerning the original complex workflows of step

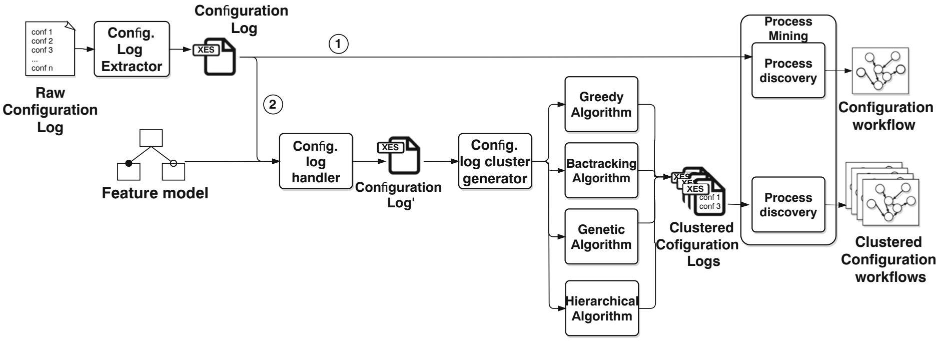


Fig. 2 COLOSSI solution overview

①. We conjecture that the resulting configuration workflows of step ② will better guide the domain engineers in the construction of a configurator as well as the analysis mentioned previously.

In the following, we describe the overall process of COLOSSI, detail its different phases and explain its implementation in general.

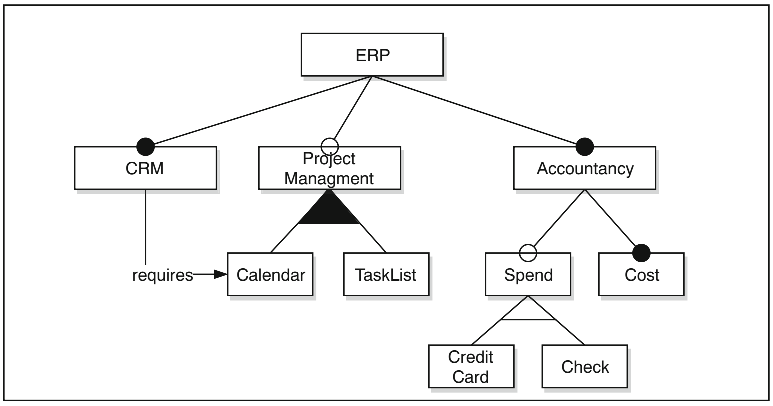
2.1 COLOSSI process

As previously mentioned COLOSSI combines a feature model and a configuration log to create configuration workflows. Consequently, it needs both as input.

A feature model is an arranged set of features that describes variability and commonality using features and relationships among them (Durán et al. 2017; Schobbens et al. 2007). FMs describe all the potential combinations of features. Figure 3 shows an excerpt of a feature model from the ERP domain where features are arranged in a tree–like structure and different relationships are established among them. FMs can be used to build configurators that are pieces of software that guide the configuration process while selecting and deselecting features. An example of a configurator is KConfig, a tool that helps to configure the Linux kernel. As an FM can define a configuration space defined by all the possible feature combinations, it can also define different possible configuration workflows that can be derived using the same FM.

To define a configuration log, we use some concepts that are used in the process mining area to describe events and traces, and we map those concepts to define a configuration log.

An event log is a multiset of traces:



Feature model

1st configuration: {*crm, tasklist, cost, accountancy*}
2nd configuration: {*project management, accountancy, credit card, calendar*}
3rd configuration: {*project management, check, crm, task list*}
4th configuration: {*crm, task list, calendar, credit card*}

Configuration log

Fig. 3 ERP domain based example

Definition 1 (Event Log). Let L be an event log $L = \{\tau_1, \cdots, \tau_m\}$ as a multiset of traces τ_i.

A trace is a tuple with an identifier and a sequence of events that occurred at some point in time:

Definition 2 (Trace). Let τ be a trace $\tau = \langle case\_id, \mathcal{E}\rangle$ which consists of a $case\_id$ which identifies the case, and a sequence of events $\mathcal{E} = \{\varepsilon_1, \cdots, \varepsilon_n\}$, ε_i occurring at a time index i relative to the other events in $\mathcal{E}$.

An event occurrence is a 3–tuple with an identifier of an activity that occurred at some timestamp and that can have additional information:

Definition 3 (Event occurrence). Let ε be an event occurrence $\varepsilon = \langle activity\_id, timestamps, others\rangle$ which is specified by the identity of an activity which produces it and the timestamps. It can store more information (i.e., states, labels, resources, etc.) which fall into the category of $others$ and which are not used in this approach.

In COLOSSI, we conceptually map elements from the feature modelling domain to the process mining domain as shown in Table 1. Concretely, an event log is conceptually a configuration log. A trace is an ordered configuration, i.e., a *configuration trace*, thus, it is a set of selected features that follow a given order. Finally, an event occurrence is a feature. Additionally, a feature can have more information like attributes associated with this feature, such as preferences, metrics or the like, which are not used in our proposal.

Thanks to this mapping, COLOSSI can be used in different scenarios to leverage process mining in variability management, like the one presented in this paper, which is the building of configurators, if the order of previous configurations are known and can be extracted. However, we envision other areas where process mining can be used to automate different tasks. Next, we describe those scenarios, also related to software product lines, from our experience and perspective:

- **Configurator building**. Up to now, configurators building is performed using manual mechanisms or, at most, using the information present in the variability model (e.g., tree traversal in feature models) (Lettner et al. 2019). With COLOSSI, we open the door to use existing configuration logs to build configurators. This novel approach can open the door to new ways of assisting configurators builders by using the generated configuration workflow to optimise configurators.
- **Data analysis**. From the generated configuration workflow it is possible to perform such analysis in terms of graph metrics. Deadlocks identifications, misalignment analysis, metrics extraction –to just mention a few– are areas where process mining techniques can be useful.

Table 1 Mapping concepts

Process mining	Product Line
Event log	Configuration log
Trace	Configuration trace
Event occurrence	Feature

- **Testing**. From the data extracted in the former item, it could be possible to define new sampling techniques (Thüm et al. 2014) that can improve the identification of bugs or feature interactions in existing product lines.
- **Variability reduction**. One of the challenges for companies that develop software product lines is variability reduction (Bosch 2018). While variability is a must in a software product line approach, it is always difficult to find a trade-off between a high degree of variability and systematic management of such variability. In this context, experts claim for techniques and tools to reduce variability while preserving configurability. Process mining techniques presented in this paper can be a first step towards defining tools to assist in the decision of variability reduction.
- **Reverse engineering**. One of the inputs used when reverse engineering feature models are configurations (a.k.a. product matrix). We envision that the techniques described in this paper can be used in reverse engineering of variability models. For instance, the generated configuration workflow can be analysed to better guide reverse-engineering algorithms.

2.2 Detailing COLOSSI steps

Once the overall process of COLOSSI has been described. We proceed to deepen the operation of each of its parts, following the structure described in Fig. 2.

2.2.1 Configuration logs extractor

A configuration log is composed of a set of configuration traces where each configuration trace encodes not only the features of a configuration but the timestamps indicating when each feature was selected. In a raw configuration log, we can find a diversity of meta-information among the selected or deselected features. Moreover, this meta-information can be presented in an unstructured or structured fashion that must be properly extracted and transformed (Valencia-Parra et al. 2019a, b) to obtain the traces in the correct format.

In this first step, we take as input a raw configuration log and output a set of configuration traces. Therefore, we need to i) search for the meta-information encoding the timestamps for each feature. Note that this might not be explicit and can be provided using other mechanisms (e.g., line numbers in a plain text format); ii) use this meta-information to represent the feature selection order, and; iii) store the set of configuration traces in a format that can be used throughout the configuration workflow retrieval process (e.g., XES serialization 2016). After this, we end up with a set of configuration traces that represent the selection order used by the users to configure the systems. However, there might be non-valid configurations and other erroneous configurations with respect to domain information. Practitioners have to make decisions to this regard depending on the kind of input logs that want to be considered in the next steps of the discovering process.

2.2.2 Configuration logs handler

At this step, the configuration log might contain non-valid configurations, erroneous partial selection of features among other domain-related errors such as those depicted in Felfernig et al. (2018). To remove clutter and noise out of the workflows, users might prefer to remove such information from the configuration log. This cleaning step consists of removing the wrong selection of features (a.k.a non-valid partial configurations) as well as generate metrics that can be later exploited to optimise the workflow retrieval process. For example, the use of atomic-sets to complete partial configurations.

Depending on the expected workflow usage, domain engineers have to define the meaning of a valid configuration and the metrics to rely on. For example, an SPL engineer might consider only configurations with complete assignments of features to develop a configuration while other might find interesting to consider the partial assignments (i.e., to configure only the variability part of the product line, keeping aside the common parts).

COLOSSI (Fig. 2), *Process mining - process discovery* module enables to read an event log and generates a process model that fits these traces as shown in Fig. 4. In the case of the variability context, a *configuration log* is read and a *configuration workflow* is obtained using the same techniques used for classical process mining. As depicted in Fig. 2, the process discovery can be applied to a single configuration log or a set of them.

2.2.3 Configuration logs cluster generator

Configuration processes can have a high degree of variability, especially when the configuration order is defined by human decisions. The application of process discovery in this type of scenarios tends to produce spaghetti-like processes, making it necessary to apply pre-processing techniques. Configurability contexts are especially variable in relation to the executed activities derived from high human intervention. Thereby, we propose to divide the traces into subsets, to model different profiles of users, thus avoiding the discovery of non-user understandable processes. In these contexts where process discovery is used to infer spaghetti-like processes, clustering techniques such as a pre-processing step are frequently applied (Hompes et al. 2017). We propose the definition of the suitable number of clusters and the division of the configuration traces into multiple clusters before the application of a process discovery method to adapt the solution to configuration tasks. This division leads to

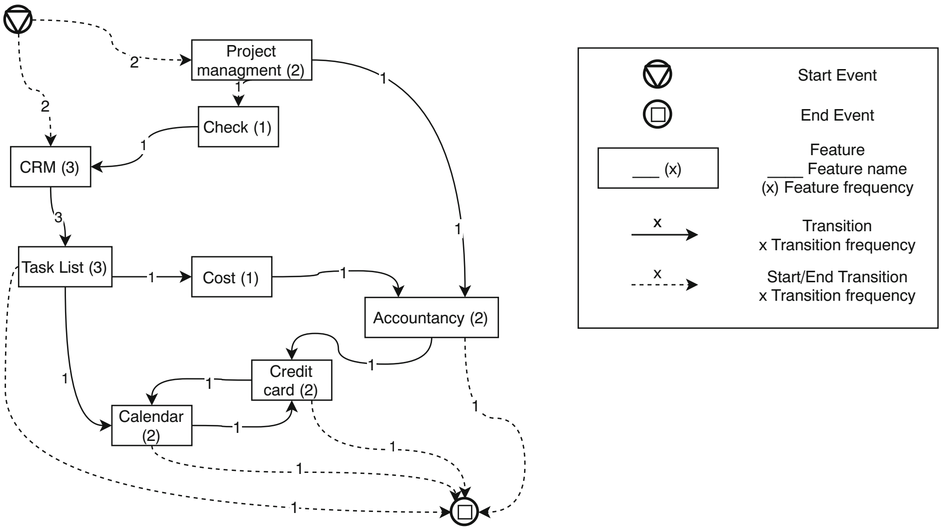


Fig. 4 Process discovered for configuration log of ERP domain based example

discover configuration workflows with more quality. This section describes what a cluster is and the metrics (e.g., entropy) used to divide the traces among a number of clusters.

Definition 4 (Number of Clusters). Let $k\text{-}cluster$ be the optimal number of clusters in which the traces must be grouped.

Like in any other clustering problem, the selection of the most optimal value of $k\text{-}cluster$ is one of the first issues to deal with, since, before being able to execute any grouping algorithm, it is necessary to previously know in how many groups the data must be distributed, in such a way that the distribution is optimal based on an established criterion.

This is a widely researched topic in which many solutions have been proposed and applied to other scenarios. In our proposal, 17 different indicators are used as reference to choose the best number of clusters: kl (Krzanowski and Lai 1988), ch (Caliński and Harabasz 1974), hartigan (Hartigan 1975), cindex (Hubert and Levin 1976), db (Davies and Bouldin 1979), duda and pseudot2 (Duda et al. 1973), ratkowsky (Ratkowsky and Lance 1978), ball (Ball and Hall 1965), ptbiserial (Milligan 1980, 1981), frey (Frey and Van Groenewoud 1972), mcclain (McClain and Rao 1975), gamma (Baker and Hubert 1975), tau (Rohlf 1974), dunn (Dunn 1974), sdindex (Halkidi et al. 2000), sdbw (Lebart et al. 2000).

Being L a configuration log composed of a set of configuration traces (i.e., $[\tau_1, \cdots, \tau_m]$), a cluster is a subset of configuration traces from L that complies certain properties.

Definition 5 (Cluster of Configuration Traces). A partition of a set of configuration traces is a set of non empty and disjoint subsets C=$\{c_1, \ldots, c_n\}$ of configuration traces, where $\bigcup_{c \in C} c = L$ and $\forall c_i, c_j \rightarrow c_i \cap c_j = \emptyset$.

The distribution of configuration traces between various clusters depends on the purpose of the practitioners. In our case, the goal is to group the more similar configuration traces. COLOSSI understanding of 'similar' is related to both features and transitions involved in the logs. Understanding as transition any edge present in the workflow that comes out of an activity.

For this reason, we adapted the classical information entropy metric (MacKay 2002) by introducing two different custom entropy metrics for clustering in the configuration context:

- *Entropy-features ($S_{features}$)* of a cluster: a metric which measures the similarity between traces according to the features that belong to the same cluster. Thus, it is the ratio between the number of features that do not appear in all configuration traces ($features_{nat}$) and the number of different features in all the configuration traces ($features_{diff}$):
$$S_{features} = \frac{|features_{nat}|}{|features_{diff}|} \tag{1}$$
- *Entropy-transitions ($S_{transitions}$)* of a cluster: a metric which measures the similarity between traces according to the transitions that belong to the same cluster. Thus, it is the ratio between the transitions that do not appear in all configuration traces ($transitions_{nat}$) and the number of different transitions in all the configuration traces ($transitions_{diff}$):
$$S_{transitions} = \frac{|transitions_{nat}|}{|transitions_{diff}|} \tag{2}$$

In order to illustrate the calculation of entropies per cluster, the $S_{features}$ and $S_{transitions}$ for the *Cluster 1* and *Cluster 2* of Fig. 5 are determined in Table 2. Reminding that we

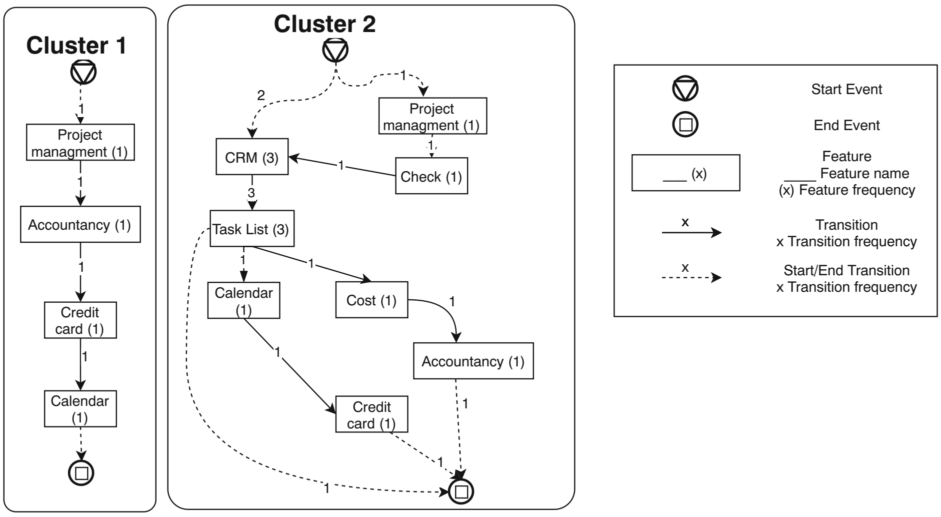


Fig. 5 Clustering for the ERP excerpt using the *Entropy-features*

consider as transition any edge present in the workflow out of an activity. In the case of Entropy-features, in the first cluster, the result is 0, since there is only one trace and there is no possibility of comparing the features. In the second cluster, we found six features that only occur in one of the two traces (*Project Management, Check, Calendar, Cost, Accountancy and Credit Card*), out of the eight that are totalled. Similarly, the presence of a single trace makes Entropy-transitions 0 in the first cluster, while in the second cluster we count nine transitions that only occur in one of the two traces (*Project Management - Check, Check - CRM, Task List - End, Task List - Calendar, Task List - Cost, Calendar - Credit Card, Cost - Accountancy, Credit Card - End, Accountancy - End*), among the ten possible ones.

Note that the range of the entropy is [0..1]. The values of entropy that are close to zero represent more similar traces, whilst when they are close to one represent that there are different features involved in the traces of the cluster. The best configuration of clusters obtained from a set of configurations traces is the one that has been partitioned into as many clusters as indicated by the optimal value of $k\text{-}cluster$, and which, in turn, minimises the summation of the entropy of all clusters. The challenge is how to obtain the best configuration of clusters as a pre-processing of process discovery.

To find out the best configuration traces divided into clusters, minimising the entropy of the resulting clusters, different algorithms can be used. In accordance with Jain et al. (1999) *clustering* provides an unsupervised classification of patterns (observations, data items, or feature vectors) into groups (clusters). *Clustering* (Kobren et al. 2017) brings together a large set of algorithms that can be classified in different ways according to the point of view necessary in the case of study.

Table 2 Entropies for the clusters of the Fig. 5

	Entropy-features	Entropy- transitions
Cluster 1	$\frac{0}{4} = 0$	$\frac{0}{4} = 0$
Cluster 2	$\frac{6}{8} = 0,75$	$\frac{9}{10} = 0,9$

Ideally, all clustering algorithms seek to group the information into clusters as homogeneous as possible. This implies that the distances between elements in the same cluster must be minimal and that, in turn, the distances between elements in a different cluster must be maximum. In order to carry out this operation, clustering algorithms usually generate, during their execution, what is known as distance matrices. These distance matrices are calculated according to different criteria which elements should be grouped and which should not. In this area, we find widely used methodologies to create a distance matrix, such as euclidean, manhattan, etc. (Grabusts et al. 2011). But, because our approach is based on minimising the sum of entropies, we need the clustering algorithm to be aware of this when grouping information. For this purpose, the entropy matrix is constructed previously and not during the execution of the algorithm, being just a square matrix containing the entropy value between each trace pair following the definition of entropy above. It is equivalent to the distance between each pair of traces according to our criteria. For this reason, our entropy matrix will be used as a distance matrix in our clustering process when necessary.

However, as previously mentioned, before the application of the clustering algorithms, it is necessary to determine the optimal number of clusters. Classically, the algorithms to determine the number of clusters (optimal k-$cluster$) also build distance matrices. For this reason, our approach will also use the entropy matrix as the distance matrix to calculate the optimal k-$cluster$. To find it, we obtain the optimal value of k-$cluster$ for each of the 17 indicators. Nevertheless, the values of k-$cluster$ derived from the different indicators obtained could be very dissimilar in some cases, therefore, frequently the optimal k-$cluster$ is selected from the most frequent value, in other words, the most voted k-$cluster$ is the number of optimal clusters by the indicators. This is the reason why we use a dendrogram to help approximate the optimal k-$cluster$, since if after the votes made by the indicators, a clear consensus is not reached, the user can use the dendrogram to be able to visually interpret how the groups would turn out. So, if for example, the indicators propose to divide the information into 2 or 3 clusters, without consensus, the user can decide, based on what is observed in the dendrogram, to work with a k-$cluster$ value of 2. Figure 9 shows an example of dendrogram.

In our proposal, we compute the different values of k-$cluster$ in a range between [0-10] for each indicator using the hierarchical algorithm explained in Section 2.2.3 and when the most voted, for a specific case study and entropy, is selected, that k-$cluster$ value is used by all the clustering algorithms in our proposal. For the example presented in Fig. 3, we have obtained an optimal k-$cluster$ value of 2.

Obviously, the entropy matrix and the k-$cluster$ determinations represent two of the weak points of our proposal, since, without an entropy matrix, it is not possible to determine the k-$cluster$ and without it, it is not possible to execute any clustering algorithm. In fact, as it will be seen in Section 3, some results have not been possible to obtain due to the impossibility of computing the entropy matrix, as is the case of Smart Farm with entropy by transitions.

The selection of the most suitable distribution of traces among clusters to discover the later process, and therefore, the value of k-$cluster$, is crucial since an incorrect distribution of the traces will produce non-understandable processes. In addition, there exists a high computational complexity to find out the optimal distribution. As a first approach, we propose to use a trivial algorithm (greedy), a complete algorithm (backtracking), and two approximation algorithms (genetic and hierarchical). For this reason, the comparison of the application of four algorithms greedy, backtracking, hierarchical, and genetic algorithm is proposed to ascertain the most suitable distribution. In general, to find a solution to a described problem, each algorithm must define how solutions are created. This implies the definition of the following five components:

- **A candidate set** used to create a solution, this is the list of disordered traces that confirm the problem that will be assigned to a cluster.
- **A selection function**, which chooses the best candidate among the candidate set to be added to the solution. It chooses the cluster to be assigned for a trace.
- **An objective function**, which assigns a value to a solution, or a partial solution used to compare the adjustment of the solutions. The objective function aims to minimise the entropy of the traces assignment among the cluster.
- **A solution function**, which indicates when a complete and most appropriate solution is discovered. It implies the assignment of every trace to a cluster.

Greedy algorithm Greedy algorithms are frequently applied in the case which the computational complexity of the problem is very high, as is in the distribution of thousands of traces in various clusters. A greedy algorithm is a strategy that evaluates each decision to reach the optimal solution only taking into account the current state, with the goal of this eventually leading to a globally optimum solution. This implies that the greedy algorithm selects the best solution in each moment without regard for consequences in a far future, only regarding the next candidate. The use of a greedy algorithm cannot assure to achieve the best solution, however, the solution can be found in a short time.

We propose the ordered assignation of each trace to a cluster that minimises the global defined entropy.

Let be $\{t_1, \ldots, t_i, \ldots, t_n\}$ the set of traces and $\{c_1, \ldots, c_e, \ldots, c_k\}$ the set of clusters that represent the candidate set for each trace. The selection function choose the best cluster c_e, that minimises the entropy after the assignment (objective function). At the start, the entropy is zero, when a new trace, t_i, is assigned to a cluster, for instance c_e, a new entropy e_i is obtained. In each step of the algorithm a trace is assigned to a cluster, if the set of traces $\{t_1, \ldots, t_{i-1}\}$ have been already assigned to any clusters and being the current entropy e_{i-1}, when a new trace (t_i) is assigned to any cluster (c_e) a new entropy $e_i(t_i, c_e)$ will be obtained.

The trace t_i is assigned to c_e iff $\forall c_d \mid c_d \in \{c_1, \ldots, c_e, \ldots, c_k\}\ e_i(t_i, c_d) \geq e_i(t_i, c_e)$

Firstly, applying the proposed greedy algorithm to the ERP domain example with two clusters, the traces distribution obtained the processes as shown in Fig. 6. Two of the four traces have been grouped in each cluster according to $S_{features}$. Secondly, three traces have been grouped in one cluster and one trace in another concerning the $S_{transitions}$. The second distribution is similar to the one shown in Fig. 5. The process models depicted in the figure are in BPMN[1] format.

Table 3 shows the resulting entropies for each algorithm, entropy type and distribution. The entropy value present in the table for each pair of entropy type and algorithm is calculated as the mean of the partial entropies obtained by each cluster. As previously mentioned, the entropies help to understand how similar are the configuration traces grouped into the clusters concerning features and transitions. The values close to zero are the most desirable. In this case, the backtracking and hierarchical with the entropy of features returned the best results of distributions. Note that, the drawback of backtracking is affordable due to the small size of the problem.

[1] BPMN: Business Process Model and Notation

Table 3 Entropy per algorithm and entropy for the ERP domain example of Fig. 5

Greedy Algorithm		Backtracking Algorithm		Genetic Algorithm		Hierarchical Algorithm	
Entropy features	Entropy trans.	Entropy features	Entropy trans.	Entropy features	Entropy trans.	Entropy features	Entropy trans.
0.625	0.9166	0.4375	0.9166	0.5	1	0.15	0.83

Backtracking algorithm A technique to analyse every possible solution is backtracking algorithms, they solve problems recursively to building a solution incrementally, one decision at a time, such as the greedy algorithm but selection every assignment and removing those solutions that fail to optimise the function.

The complete analysis of every solution implies an exponential computation time, making backtracking algorithms a high time-consuming technique. For this reason, we applied several improvements to reduce the possibilities:

- Apply a branch and bound mechanism to avoid the exploration of some branches, where several solutions from the greedy algorithm are used to bound.
- Avoid symmetric solutions, to analyse equivalent solutions (assignments) with the same number of traces together but in different clusters.

Let be $\{t_1, \ldots, t_i, \ldots, t_n\}$ the set of traces and $\{c_1, \ldots, c_e, \ldots, c_k\}$ the set of clusters that represent the candidate set for each trace. The selection function chooses every cluster c_e (if more than one has no traces assigned yet, only one of them will be analysed to avoid the creation of equivalent solutions), avoiding those that generate a higher entropy that the best solution found until the moment. At the beginning, the entropy is the one obtained from the greedy algorithm, when a new trace t_i is assigned to a cluster c_e, a new entropy e_i is

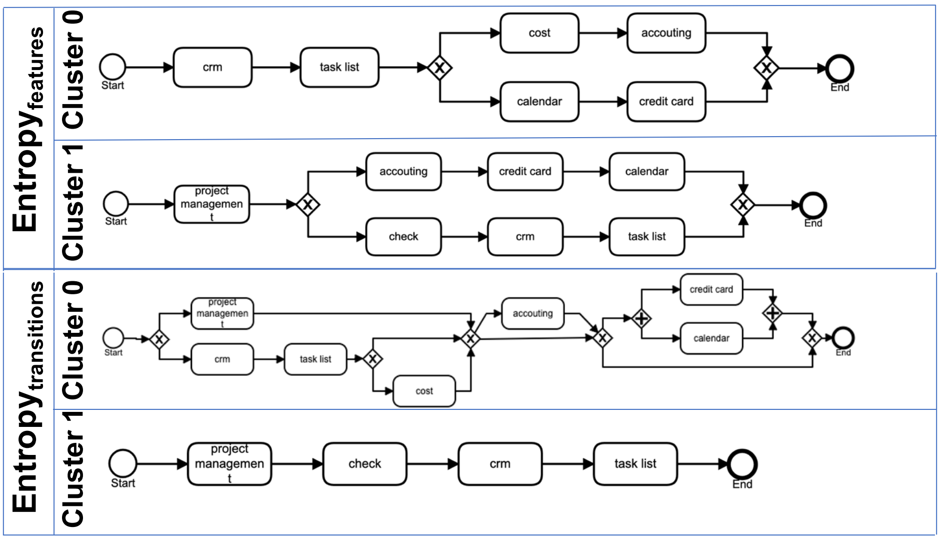


Fig. 6 Processes obtained per clusters and entropies using a greedy algorithm

obtained. In each step of the backtracking algorithm, a trace is assigned to a cluster, then if $\{t_1, \ldots, t_{i-1}\}$ have been already assigned to the clusters and being the current entropy, e_{i-1}, when a new trace (t_i) is assigned to a cluster (c_e), a new entropy $e_i(t_i, c_e)$ will be obtained. If the new entropy $(e_i(t_i, c_e))$ is a worse entropy than the best found until the moment $(e_{bestSol})$, the candidate will be skipped, a new trace will be assigned otherwise.

For all cluster c_d a solution is created where a trace t_i is assigned to c_d iff $e_{bestSol} > e_i(t_i, c_e)$. Each new created solution will be used recursively in the backtracking algorithm.

Applying the proposed backtracking algorithm to the ERP domain example, with two clusters, the traces distribution obtained is shown in Fig. 7. In this case, the models obtained for both entropy are the same, for this reason, only the models for $S_{features}$ are shown. The *Cluster 0* grouped one trace and the *Cluster 1* grouped the rest. Regarding the entropy of features, this is the one of best distribution (i.e., assignment of traces to clusters). This distribution is not obtained by the greedy solution, but the backtracking ensures that it is the optimum solution in terms of the objective function.

However, even applying the aforementioned improvements, it has not been possible to obtain results in our study cases with this algorithm, since, due to their size, the search could not finish in an acceptable time as we will show in Section 3.

Genetic algorithm Genetic algorithms provide a trade-off between high computational requirements and the necessity to find the best solution. Genetic algorithms are commonly used to generate high-quality solutions that optimise the problems by relying on bio-inspired operators such as mutation, crossover and selection by using a population of candidate solutions, where it is necessary to define:

- A genetic representation of the solution domain is a list of assignments, l_i. Starting from a set of traces $\{t_1, \ldots, t_i, \ldots, t_n\}$ and from a set of clusters $\{c_1, \ldots, c_i, \ldots, c_n\}$, it is possible to build another list whose size is equal to the number of traces, in which each l_i value will imply that the t_i trace has been assigned to the c_i cluster. For instance, let be $\{t_1, t_2, t_3, t_4, t_5\}$ a set of traces and 2 the number of clusters to distribute them. A possible assignment would be [0, 1, 0, 1, 1], which means that the traces t_1 and t_3 belong to cluster 0 and traces $\{t_2, t_4, t_5\}$ to cluster 1.
- A fitness function to evaluate the solution domain seeks to group the maximum number of traces according to their entropy and is a maximisation function based on the entropy value of the assignment. It is calculated as the inverse of entropy (*Entropy-features or Entropy-transitions*) plus one to which we add three weights according to the advantages of the assignment. Each of the weights refers to one goodness of allocation: (g_1)

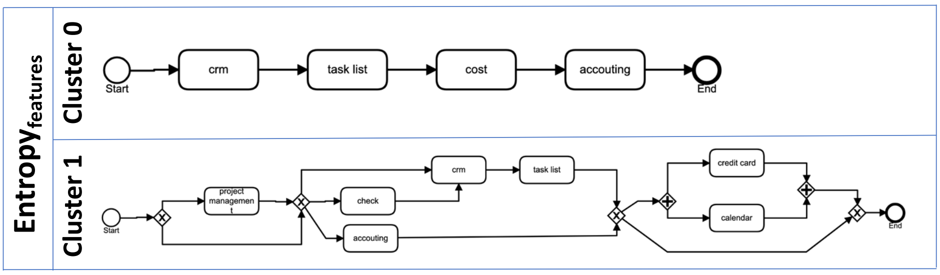


Fig. 7 Processes obtained per clusters and entropies using a backtracking algorithm

there are not empty clusters; (g_2) the number of traces assigned to each cluster is balanced, and; (g_3) the similarity between the traces within each group is also balanced. With this fitness function the individual will be better, the higher fitness value it gets. We preserve the same objective function for all algorithms, which is the entropy, but in the case of the genetic, just the entropy as the objective function is insufficient, because there are situations in which clusters can be empty or unbalanced. To adjust better the solution, we included some weights. These weights are discrete values and allow us to penalise or favour a certain individual regarding its characteristics. These weights are not necessary for the rest of the algorithms applied, because, by their construction, they take care that the events considered in functions g_1, g_2, g_3 do not occur. For instance, the backtracking algorithm includes mechanisms to avoid those solutions that include empty clusters or in which the clusters are unbalanced. However, the genetic algorithm we apply does not possess these abilities and requires the objective function to explicitly penalise candidates that would not be acceptable solutions. As these weights are discrete values, they are customisable and adaptable to the context of each problem and user, so that the user can decide which properties want to enhance or soften in the allocations.

$$fitness = \frac{1}{1 + entropy} + g_1 + g_2 + g_3 \tag{3}$$

- Different rates will determine the evolution of successive populations during the execution of the algorithm, and therefore, of the final solution. the maximum population growth has been limited to 600,000 generations, each one with a size of 80 individuals. We have used an elitism rate of 30%, a crossover rate of 80% and a mutation rate of 70%. This selection of the parameters is due to different trial and error tests, in which this parameterization has yielded the best results. The trial and error tests were performed with all the cases to find the best general parameterization. Once the best combination of parameters was determined, the same one was used for all the experimentation. The designed test consists of the execution of each case several times with different parameters configuration and the calculation of the average execution time (in seconds) spent in each.
- Crossover policy: one-point crossover, where a random crossover point is selected and the first part from each parent is copied to the corresponding child, and the second parts are copied crosswise.
- Mutation policy: binary mutation, in which randomly one gene is changed.

Applying the proposed genetic algorithm to the ERP domain example with two clusters, the traces distribution obtained the next processes as shown in Fig. 8. In this case, the genetic algorithm seems to balance quite well each cluster considering both entropy distributions. In both cases, each cluster groups two traces. However, as will be seen later in Section 3, it is not the one that offers the best results in terms of entropy.

Hierarchical algorithm The well-known *hierarchical agglomerative clustering* algorithm is based on the work by Ward (1963) as the combination of hierarchical and agglomerative clustering.

As it was previously stated, we use this algorithm with our entropy matrix as the distance matrix. From here, it is possible to decide which grouping criteria the algorithm must follow when creating clusters. Examples of this methods are *single-linkage*, *complete-linkage* (Hubert 1974), *average-linkage* (Murtagh 1983), and *Ward* (Ward 1963). Being the last one, the one used in our solution, since it is one of the most used in practice and it has proven

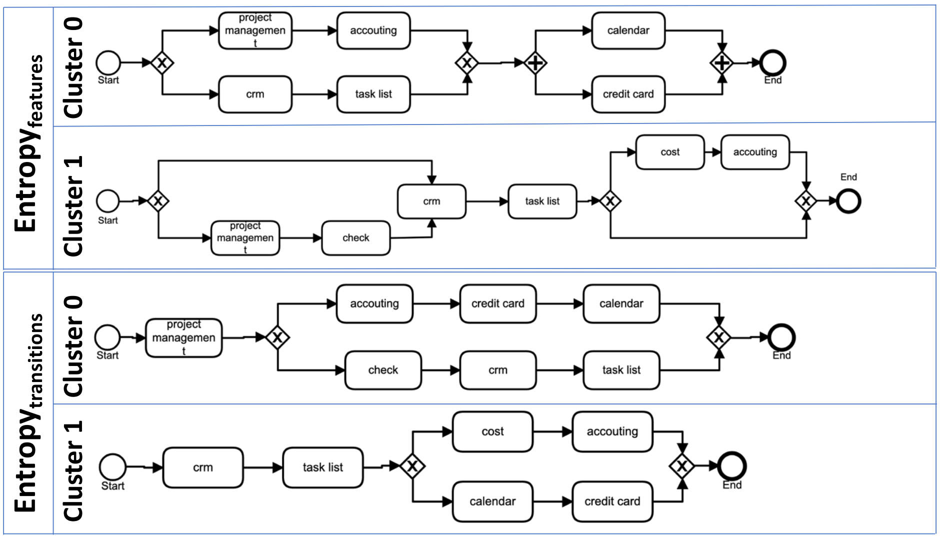


Fig. 8 Processes obtained per clusters and entropies using a genetic algorithm

to be more accurate obtaining the optimal distribution, than the previous ones and avoiding that the partition in groups distorts the original information (Kuiper and Fisher 1975).

On the first hand, *hierarchical clustering* is defined as a procedure to form hierarchical groups of mutually exclusive subsets, each of which has members that are maximally similar concerning the specified characteristics. In the same study, authors define the process as "*assuming we start from n sets, it permits their reduction to* $n - 1$ *mutually exclusive sets by considering the union of all possible* $\frac{n(n-1)}{2}$ *pairs and selecting a union having a maximal value for the objective function*".

On the other hand, *agglomerative clustering* is an algorithm that starts from the assumption that each element constitutes a cluster by itself (singleton) and it successively merges these singletons forming clusters until a stopping criterion is satisfied which is also determined by the objective function.

The elements used in our solution are based on both entropies presented (features and transitions), combined with the objective function Ward's minimum variance method (Ward 1963). This function aims to minimise the sum of the squared differences within all clusters, which means, a variance-minimising approach. Figure 9 shows the obtained dendrogram[2] for the example in Fig. 3 by using Entropy-features. In the mentioned dendrogram, a horizontal line that intersects perpendicularly with two lines, appears, symbolising partitioning into two groups: the first one, corresponding to the first cluster, which contains only one trace, and the second, corresponding to the second cluster, which contains the three remaining traces. The positioning of the horizontal line is determined by the optimal $k\text{-}cluster$ value since this line must be placed at a height where it intersects with as many vertical lines like the $k\text{-}cluster$ value is.

Once the clusters have been generated, it is possible to create a new log for each one of the clusters and to discover the workflows corresponding to them. In the same figure, the

[2] *Dendrogram* that is a branching diagram which represents the arrangement of the clusters produced by the corresponding analyses

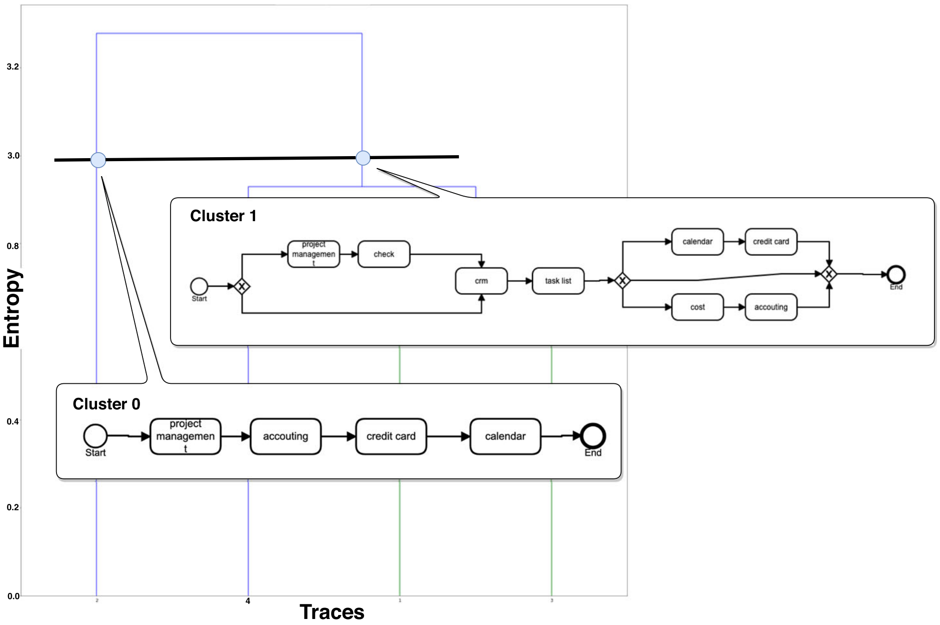


Fig. 9 Clustering for the ERP excerpt using the *Entropy-features* in an agglomerative clustering algorithm

process models obtained according to the clustering division based on entropy-feature and transition have been included, since the same models are obtained using both entropies.

2.2.4 Configuration workflow discovery with process mining

As it could be seen in Fig. 2, once the clusters have been obtained, the next step is to generate the logs for each cluster and execute the process mining on each one of them.

Process mining is a family of techniques based on event logs that can be categorised as process discovery, conformance checking and enhancement (van der Aalst 2016). In this paper, we are focused on the use of *process mining* to analyse the configuration logs for the discovering of configuration workflows based on user experiences. Process discovery in process mining brings together a set of algorithms to generate a workflow process model that covers the traces of activities observed in an organisation (Maruster et al. 2002). The evolution of algorithms during last decades has allowed the discovery of complex models that are able to involve not only the activities executed in the daily work of companies but also the persons who execute them and the used resources.

Process mining is an important topic that has been well received by the enterprises, bringing about the evolution of the research solution tools (e.g., ProM (van Dongen et al. 2005)) to commercial solutions (e.g., Disco[3] and Celonis[4]). This facilitates its applicability to several contexts and areas, although variability has been out of the scope of these techniques before our proposal.

[3] https://fluxicon.com/disco/

[4] https://www.celonis.com/

The purpose of obtaining these configuration workflows is to assist the user who performs the configuration. So that, thanks to this help, it is possible to know what actions must be taken before others, which of them can be carried out in parallel, with which activity should start the configuration, what task should be performed after the current one in order to do it optimally, etc. In this way, following our example, a user who is facing the configuration of an ERP system could know that he could start by configuring the CRM, after that, the task list, followed by the calendar, ending with the credit card.

Process discovery in process mining uses a set of traces similar to the configuration log shown in Fig. 3, to obtain a model that covers the possible traces. Figure 4 shows the process discovered by Disco tool-suite of the example, which covers every possibility configuration trace. The relational patterns among the definition of the features become part of the model. For example, two features can be the first in the traces (*CRM* or *Project management*) or after *CRM* always *Task List* is selected. Figure 4 also shows the number of traces that are represented by each transition as labels of the edges, giving information about the importance of each part of the traces in the obtained model.

2.3 COLOSSI implementation

COLOSSI is supported by the implementation of a framework which is made up of one module for each activity of the process shown in Fig. 2.

1. *Configuration log extractor* is a piece of a software module which takes a set of raw configuration log (including timestamps) in a semi-structured format and returns an XES file. It corresponds to the activity with the same name in Fig. 2.
2. *Configuration log handler* is another piece of software which takes a FM and an XES log as input. First, apply a set of operations over the FM as described in Section 2.2.2. Then, a data cleaning is carried out over the XES log to get a filtered configuration log. The output of this connector is a new XES log with the filtered configuration log. It corresponds to the activity with the same name in Fig. 2.
3. *Cluster generator* is a Python/R module which takes an XES log file which is translated into a matrix. This matrix enables the entropy calculation and the optimal $k\text{-}cluster$ that will be used when some of the clustering algorithms are applied to determine the clusters. A new XES log file is generated for each cluster that composed the final output of this component. This module is represented in Fig. 2 by the activity called Configuration log cluster generator, the different clustering algorithms that can be applied and the generation of new the logs.
4. *Discovery connector* is a piece of software which gets the XES file logs of each cluster and automatically feed the ProM to discover the process models utilizing the *Inductive Miner*. The output of this component is a process model in Petri-net or BPMN format. This module is depicted in Fig. 2 by the Process Discovery activity.

The reason for using ProM in our proposal is because it is the free framework most used by the academy. In it, the Process Mining community adds its contributions as plug-ins, and therefore, it is always up-to-date with new solutions to problems that are under research. On the other hand, it is a very powerful tool that contains all the Process Discovery algorithms, and its construction allows it to be used inside of another software. The selection of the Inductive Miner algorithm is reasoned by the great capacity of this algorithm to fully adjust to the behaviour observed in the log. As explained in Section 3.4, it is one of the most robust algorithms, which produces better results when facing non-synthetic logs, ensuring the correctness of the models obtained. Its parameterization has been adjusted to 100% of fitness, in such a way that all the configuration workflows obtained will completely cover all the traces contained in the log, representing all the possible behaviours that have been collected.

All the resources, thus, configuration logs, the XES files, the workflows discovered, the source code of the COLOSSI framework (i.e., git repository), and a Jupyter notebook that are employed in this work are freely available at[5] http://www.idea.us.es/empiricalsoftware/. The notebook is self-explanatory and allows users to work interactively by executing step-by-step instructions to get the clusters.

3 Evaluation

In this section, we present the evaluation of COLOSSI. Concretely, the evaluation of the algorithms of clustering detailed in Section 2 to different configuration logs obtained from three real scenarios. Moreover, the suitability of each algorithm according to a set of metrics are analysed.

3.1 Experimentation data

In order to analyse the applicability of our example in a configuration real scenario, we propose three different scenarios where the creation of the configuration workflow can be obtained from a set traces: a real ERP (enterprise resource planning) configuration, smart farming, and a computer configurator. Please note that the number of possible products depicted by the models is an interesting metric to understand the usefulness of this approach when performing variability reduction. However, our approach relays on the number of real configurations to obtain the configuration workflows. This is further explained in Section 3.5.

3.1.1 Enterprise resource planning

This dataset (Pereira et al. 2016b) reflects the information of a real ERP variability among a set of configuration logs. The ERP feature model has 1920 features and 59044 cross-tree constraints. Also, the configuration log consists of 35193 event occurrences that represent a total of 170 different configuration traces with an average of 207 features per configuration trace.

3.1.2 Smart farming

This dataset represents several e-commerce transactions from the agribusiness domain (Pereira et al. 2016a). Concretely this model consists of 2008 features. It contains features targeting final customers (around 10%) and business to business (around 90%). Each log consists of a real configuration developed for a concrete user or business having a total of 5749 logs and up to 10^9 possible configuration due to the lack of cross-tree constraints.

[5] https://doi.org/10.5281/zenodo.3574053

3.1.3 Computer configuration

This dataset represents the variability existing in a Dell laptop (Pereira et al. 2018a). It reflects features such a processor or display, among others. Concretely this feature model represents 68 features with seven cross-tree constraints that encode up to 10^9 configurations according to their creators.[6] The configuration log is composed of 42 configurations of such a model.

3.1.4 COLOSSI setup

In this section, we detail each task of the framework presented in Fig. 2.

Configuration log extractor The input data of the configuration examples (i.e., ERP, farming and computers) are represented in CSV files with two elements (columns), the configuration id and the $feature$ that is configured. Note that a feature can appear in one or more traces, but no more than once in the same trace. Then, the timestamp required to extract the traces was taken by the line number in which the features were appearing throughout the file in sequential order. This is, we assume that the timestamps were implicit based on the order of appearance (i.e., line numbers). In all cases, we iterated over all configurations extracting the orders. Finally, we transformed them into a more standard format for traces. Concretely we use in our solution the IEEE Standard for eXtensible Event Stream (XES) 2016. This is a standard to serialise, store and exchange events data, that is commonly used in process mining tools.

Configuration log handler To clean up the set of configurations retrieved by the extractor we decided to consider only valid partial and full configurations. This filtering operation is performed by using the FaMa framework (Felfernig et al. 2018), with the intention of keeping only valid configurations in the log, to avoid introducing noise into the results as a consequence of the use of invalid traces.

After filtering, the valid partial configurations using automated analysis (Galindo et al. 2018), for each case are: for ERP ended up considering 61 configuration traces from the initial set of 170, Farming example initially has 5749 traces and 919 after the filtering, and dell configuration contained the full 42 of the initial set (i.e. all traces were valid).

Determining the number of clusters As commented previously, before applying any grouping algorithm, the optimal number of clusters (i.e., optimal $k\text{-}cluster$) must be determined for each use case and entropy by applying the indicators established in Section 2.2.3.

For the three examples, as it was previously stated, we propose to analyse a range between [0-10] for $k\text{-}cluster$ since the obtained clusters will be used to create configuration workflows later used by humans. In this context of the application and the number of traces of the examples, we consider this range proper on the bases of the results shown in Table 4, that summarises the number of clusters calculated by using each case and entropy.It is important to note that due to the impossibility of computing the entropy matrix for the Smart Farm case with entropy transitions, results for this case will not appear in that table.

As described previously, the distribution of the traces among the cluster is a complex activity, and the best assignment is not a trivial task. For this reason, COLOSSI framework

[6] https://wwwiti.cs.uni-magdeburg.de/~jualves/PROFilE/datasets-download/Dell-Laptop_readme.txt

Table 4 Number of clusters per use case and entropy

Config. Workflow	Entropy	N. of Clusters (k-$cluster$)
ERP	$S_{features}$	5
	$S_{transitions}$	3
Smart Farming	$S_{features}$	6
	$S_{transitions}$	-
Config. Computer	$S_{features}$	2
	$S_{transitions}$	8

provides a set of techniques to be applied once the traces have been generated from the raw data. The objective of each algorithm is the same to distribute the traces among a determined number of clusters to minimise the summing of the entropy of every cluster for a later application of discovery process algorithms for each cluster.

Nevertheless, there are no approach helping to determine the optimal k-$cluster$ in algorithms such as backtracking or genetic, which is why, in our proposal, the optimal number of clusters is always determined using the hierarchical with the entropy matrix. Once an optimal k-$cluster$ value has been found for a case study and an entropy type, this value is used in all grouping algorithms for that case study and that entropy.

3.2 Analysis of clustering

As introduced in previous sections, two different entropies are applied to infer the clustering (i.e., *Entropy-features* and *Entropy-transitions*). The two entropy formulas can help to understand the quality of the future workflow. Thus, a lower value of entropy more quality of the cluster, hence, the workflow has more quality.

Table 5 shows the results obtained for each case and each algorithm taking into account the two entropies defined (feature and transitions). For a better comparison, the metrics associated with the clusters are aggregated as arithmetic means. For the sake of the results, the entropy of features are the best distributions in all the cases except for *Computer Configuration* in the hierarchical algorithm. These results will be confirmed with complexity results obtained with the metrics in Section 3.4.

It is important to emphasise that no results have been obtained for the backtracking algorithm and some of the cases for the entropy of transitions, caused by the huge exponential complexity that the examples imply. Because of this, two drawbacks of our approach have been identified in the application of algorithms and the determination of entropy. Regarding the backtracking algorithm, the use of a complete algorithm requires to explore all the space of solutions hence in most of the cases it requires exponential time depending on the size of the problem in terms of features, traces, and the number of configurations. The high number of features, traces, and configurations in the data used for experimentation made impossible the application of backtracking algorithm in an acceptable time since its executions in the simplest cases took more than 24 hours without results. Also, it is necessary to build an entropy matrix that acts as a distance matrix to determine the number of clusters. Due to the complexity of some scenarios, the creation of the entropy matrix was computationally impossible, disabling the calculation of the k-$cluster$ and consequently, the execution of the clustering algorithms. This is the case of *Smart Farming* where the entropy matrix with entropy-transitions was unapproachable in linear time, taking more than 10 hours.

Table 5 Entropy per algorithm and example

	Greedy Algorithm		Genetic Algorithm		Hierarchical Algorithm	
	Entropy features	Entropy transitions	Entropy features	Entropy transitions	Entropy features	Entropy transitions
ERP	0.2170	0.802	0.4609	0.793	0.1759	0.8585
Smart Farm	0.5137	–	–	–	0.5990	–
Computer Configuration	0.7830	0.9313	0.7264	0.9492	0.2566	0.2352

As previously highlighted, the characteristics and size of the configurations data will condition the achievement of results with clear limitations of resources and time. Therefore, it is necessary to compare the execution times of each algorithm for each example. Table 6 shows the duration in minutes of executions from when the clustering algorithm starts until it reaches a solution. Something important to highlight in these execution times is the evidence that the hierarchical algorithm is much faster than the rest, mostly, because the task that requires the longest computation time is the construction of the distance matrix (in COLOSSI, entropy matrix) and this had previously been done to calculate the optimal $k\text{-}cluster$. In the rest of the algorithms, the entropy matrix does not apply and they build the distances during their executions, this is the reason why their computation time is longer.

3.3 Statistical analysis of results

To check if the running algorithms have an actual impact on the Entropy performance indicators we used the *Null Hypothesis Statistical Test* in which two contrary hypotheses are formulated. On the first hand, the null hypothesis (H0) states that the selected techniques do not influence the obtained results (i.e., the algorithm has no impact on the entropy of the retrieved models). On the other hand, the alternative hypothesis states that the selected algorithm impacts the obtained results significantly (i.e., selecting a greedy or a hierarchical algorithm impacts the entropy obtained).

We decided to fix such hypothesis to understand if our techniques to improve the entropy of the resulting clusters were affected by the technique, and thus, can be useful in variability-aware scenarios. This is, testing this hypothesis intend to check whether the application of techniques coming from other contexts (e.g. process mining) provide meaningful results on variability-intensive systems.

Table 6 Execution times in minutes per algorithm and example

	Greedy Algorithm		Genetic Algorithm		Hierarchical Algorithm	
	Entropy features	Entropy transitions	Entropy features	Entropy transitions	Entropy features	Entropy transitions
ERP	1.76	0.75	105	12	0.016	0.016
Smart Farm	152	–	–	–	0.016	–
Computer Configuration	0.026	0.041	15	15	0.016	0.016

Such executed tests provide a probability value (called *p-value*) which ranges from 0 to 1. The lower the *p-value* of a test, the more likely that the null hypothesis is false, and the alternative hypothesis is correct. It is established that *p-values* under 0.05 or 0.01 are so-called statistically significant, which let us assume that the alternative hypothesis is likely true.

In this analysis, we only checked instances of the greedy and the hierarchical algorithm. This is motivated by the lack of results for all models of the genetic and backtracking algorithms for each case study. Note that this unavailability points out that the results are dependent on the techniques.

The test we relied on for performing the statistical analysis and obtain the *p-values* depends on the properties of the data (Derrac et al. 2011). Concretely, we executed the Wilcoxon test (Wilcoxon 1946) and were not able to prove that our data follow normal distribution so, we had to rely on non-parametric techniques. We executed Friedman's tests for both entropy metrics, obtaining a *p-value* of 0.0455, and a statistics of four in the case of the feature-based entropy. Thus, we have to reject the null hypothesis and then, accept the alternative one. Therefore, for the case of feature-based entropy, the selection of the appropriated algorithm impacts the quality of results. Secondly, we obtained a *p-value* of 0.5637, with a statistic of 0.333 for the case of transition-based entropy's which prevents us from rejecting the null hypothesis.

These results provide two main insights. First, we observe that for the feature-based entropy is heavily dependent on the method used. Second, we can not determine if the transition-based entropy is being affected by the method.

3.4 Analysis of discovered configuration workflows

To evaluate how the application of different clustering algorithms can improve the configuration workflows obtained by COLOSSI, in this section, for each case study and algorithm, we compare the models discovered by using a set of metrics. For each algorithm, entropy and case study, the workflows corresponding to (1) the *filtered* version of the same log including only valid configurations (i.e., after applying *configuration log handler*) and; (2) the set of clusters obtained after applying *cluster generator* explained in Section 2.2.3. The mentioned metrics helps to compare with each other, in terms of complexity and understandability, the configuration workflow of the filtered log and those extracted after the clustering. This means that we measure the quality of the resulting configuration workflow, without taking into account the input data of the feature model.

The analysis is carried out following two different perspectives: (1) the analysis of the discovered configuration workflows and (2) the analysis of the set of configuration traces involved in each cluster used in the process discovery.

First, we highlight that inductive process discovery techniques used by COLOSSI ensure the soundness and correctness of the process models obtained (Leemans et al. 2015). Thus, an analysis of the soundness and correctness of the configuration workflows are unnecessary since processes discovered is always complete, have a proper completion, and have no dead transitions.

However, the complexity of the configuration models are affected by the number of features, the number of configuration traces and the number of transitions. The filtering of the configuration traces or the division of the logs will bring about simpler configuration workflows. Table 7 depicts comparatively the number of features, configuration traces and transitions of the set of configuration logs using in each case study. The data of features,

Table 7 Characteristics of the configuration logs

				Greedy Algorithm		Genetic Algorithm		Hierarchical Algorithm	
		Original	Reduction Ratio	Entropy Features	Entropy Transitions	Entropy Features	Entropy Transitions	Entropy Features	Entropy Transitions
ERP	Features	425	48.66	129.40	253.67	211.40	333.67	146	229.67
	Transitions	2,028	64.20	260.40	863.67	806.40	1,521.33	296.40	462.33
	Traces	61	72.77	12.20	20.33	12.40	20.33	12.40	20.33
Smart Farm	Features	1,420	60.01	723.33	–	–	–	712	–
	Transitions	2,844	8.95	3,394.33	–	–	–	1,794.17	–
	Traces	919	83.33	153.17	–	–	–	153.17	–
Computer Configuration	Features	53	39.68	41.40	22.38	39	30	35	24
	Transitions	205	43.15	174	65.38	151	102.25	106.50	59
	Traces	42	66.86	21	5.25	21	5.25	21	10

transitions, and configuration traces of the clusters have been aggregated per use case, clustering algorithm and type of entropy to show the average for every cluster obtained as a solution after the application of the algorithms of clustering. In addition, the reduction ratio column contains the average rate by which each characteristic has been reduced with respect to the value presented by the original log.

The number of features and configuration traces grouped in the different clusters are decreased in comparison with the original case since they are distributed among the clusters. Regarding the transitions, these values are affected by the entropy and the algorithm used. In general, the number of transitions is reduced, but it could happen, that because the grouped traces are formed by non-common features, the number of transitions can increase, as is the case of smart farming in the greedy algorithm.

In conclusion, clusters reduce the complexity of configuration workflow discovered by reducing the configuration traces involved in the same configuration workflow. However, the question is in what level the quality of the obtained workflow is improved, and which distribution of cluster-entropy works better.

In the literature, several metrics are used to measure how “good” is a design of a business process model (Mendling 2008; Pérez-Castillo et al. 2019; Cardoso 2005). Discovered configuration workflows are also processes with features instead of activities, therefore, these metrics can be adapted to measure the quality of our obtained configuration workflows. The next set of metrics is adapted to measure the understandability and the complexity of the configuration workflows to compare the discovered configuration workflows:

- *Density*: the ratio of transitions divided by the maximum number of possible transitions. The lower the value of density, the higher the understandability.
- *Cyclomatic number* (CC): the number of paths needed to visit all features. The cyclomatic can be seen as a complexity metric, thus, the lower the value of CC, the lower the level of complexity.
- *Coefficient of connectivity* (CNC): the ratio of transitions to features. The greater the value of CNC, the greater the complexity of configuration workflows. Although, the

authors in Mendling (2008) remark that models with the same CNC value might differ in complexity regarding this parameter.

- *Control Flow Complexity* (CFC) enables to measure the complexity in terms of the potential transitions after a split depending on its type. The greater the value of the *CFC*, the greater the overall structural complexity of a workflow.

These four metrics help us to know the complexity and understandability of the configuration workflows from the design perspective and the elements in the model. Nevertheless, these metrics used to measure the quality of the workflow are inconclusive to measure the real usefulness and quality of the discovered workflows applied to the context of the variability management.

As previously mentioned, the uselessness of the quality metrics related to the workflows leads us to define a new custom metric which enables to establish the quality level of the workflow by relating the number of features and their occurrence within the discovered workflow of a cluster. Thus, a metric that enables us to measure how spaghetti is the workflow obtained. Our custom-quality metric is defined as follows:

- Quality ($\Delta_{\equiv}$) measures the difference between the total number of features and the ratio of the sum of the number of times that a feature is selected for each configuration trace and the number of configuration traces.

Formally, given a workflow based on a set of configuration traces (CT) and a set of features ($Features$), the quality can be determined as the following formula:

$$\Delta_{\equiv} = |Features| - \sum_{f \in Workflow} \frac{occurrences(f)}{|CT|} \tag{4}$$

The range of the quality is $[0..|Features|]$, the lower value of quality a better configuration workflow is indicated. The number of features and configuration traces are grouped into the most similar workflow, therefore, it brings about that the quality is near to 0.

To make the application of metric more understandable, we use the first example of Fig. 3 and the *Cluster 2* in Fig. 5. The number included in the rectangle, next to the name of the feature, corresponds to the number of traces within the cluster where the feature appears. Hence, the quality for the *Cluster 2* can be determined by applying the formula as follows:

$$\Delta_{\equiv} = 8 - \left(\frac{3}{3} + \frac{3}{3} + \frac{1}{3} + \frac{1}{3} + \frac{1}{3} + \frac{1}{3} + \frac{1}{3} + \frac{1}{3}\right) \approx 4 \tag{5}$$

The results obtained for these five metrics are shown using box-plot charts represented for each scenario in (Figs. 10, 11 and 12) to compare distribution by metric, algorithm and the type of entropy for each case study. The y axis represents, on a logarithmic scale, the results of each metric for the configuration workflows obtained using a specific algorithm. Detailed values for each case and metrics can be consulted in the Appendix section.

In the case of *ERP* scenario, Fig. 10 shows that the values for *CC* and *CFC* metrics are equal and significantly higher, which implies that the connectivity of the workflow is higher, increasing its complexity. In addition, in the case of the genetic algorithm, it is important to highlight the importance of the type of entropy, which will determine whether the clusters generated after the assignment have similar levels of connectivity or not. For *Density* and *CNC* metrics, all algorithms have similar behaviour regardless of the entropy used. In terms of *Quality*, it is clearly observed how the use of entropy by transitions produces considerably more complex workflows.

In the *Smart Farm* example, the greedy algorithm produces slightly more complex workflows but also more balanced clusters as shown in Fig. 11. Also, the hierarchical clustering

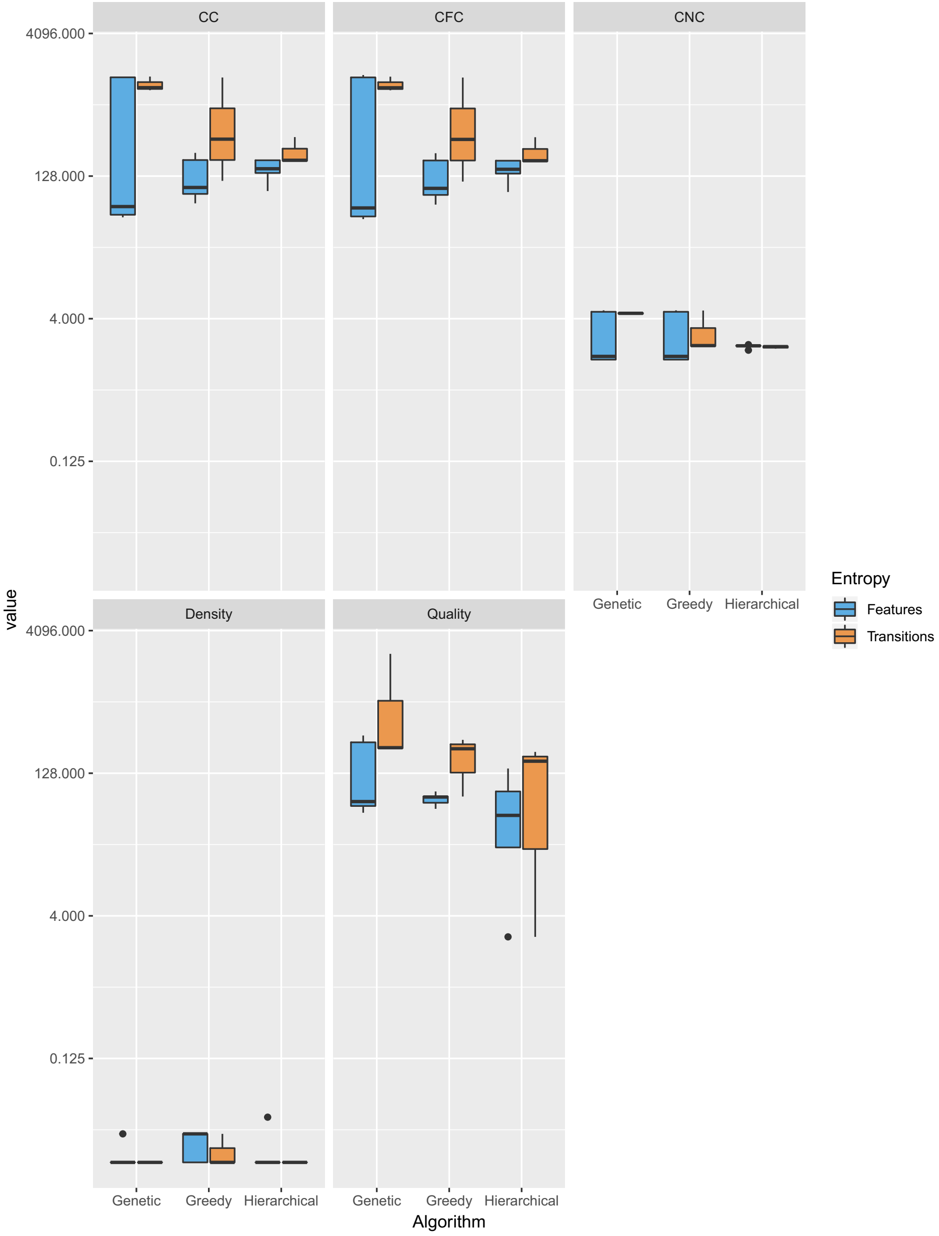


Fig. 10 Metrics distribution of ERP case study

generates less similar clusters, which reduces the complexity and the connectivity, as it is clearly observable for the case of *Quality* metric.

Finally, there are no remarkable differences between the three algorithms for the case of *Computer Configuration*, since they all produce very similar results as shown in Fig. 12. Exclusively, the use of the hierarchical algorithm with transitions entropy could be distinguished as the solutions with less complex configuration workflows. Nevertheless and in

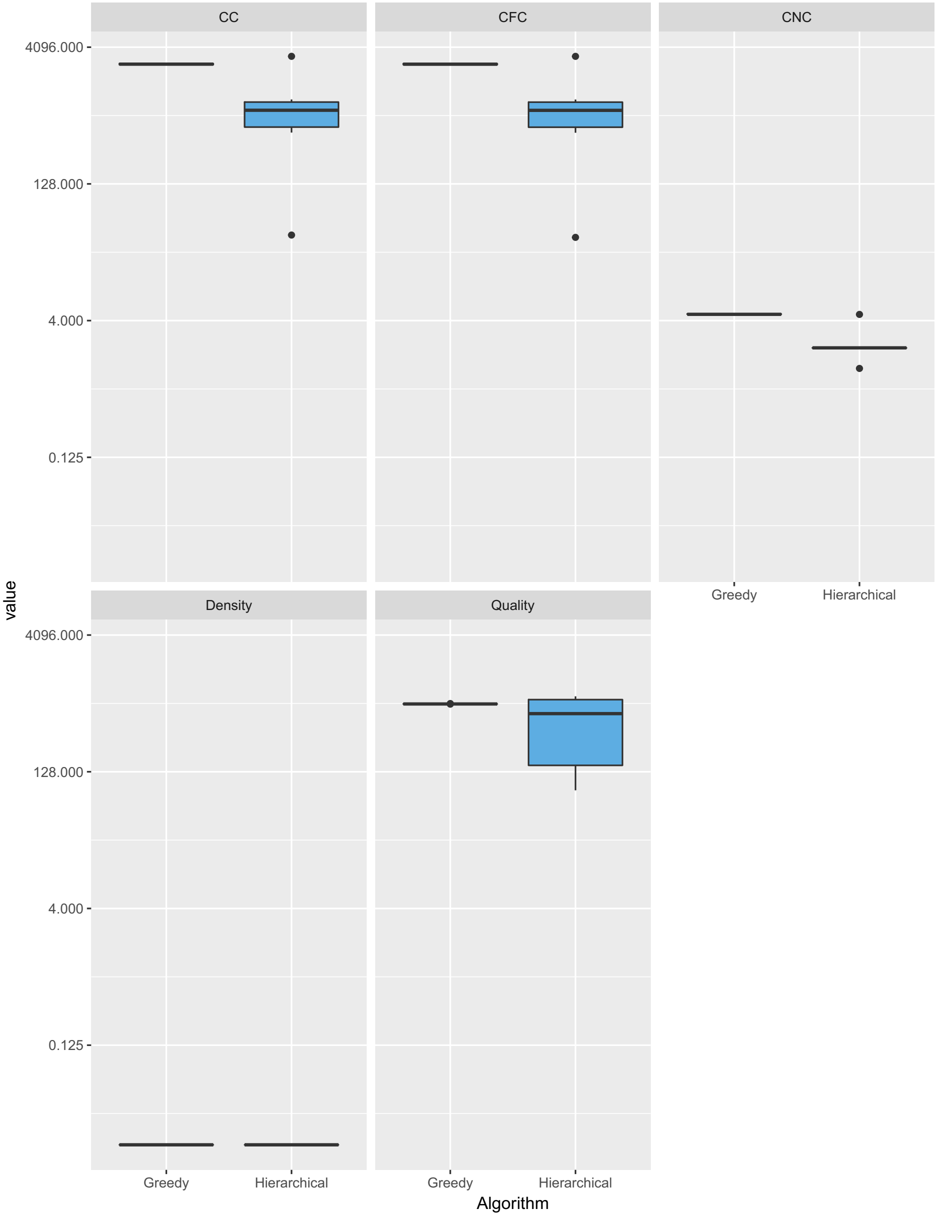


Fig. 11 Metrics distribution of Smart Farm case study

contrary to the other cases, in this particular scenario better general results of *Quality* are obtained using the entropy of transitions.

To compare all the cases with each other, we propose Fig. 13. This figure represents the average values of each metric for each case study in aggregate. In this way, we intend to compare the three examples in terms of complexity, taking a holistic view of the average values obtained by each for each metric. Thus, it is noteworthy that the mean values for three

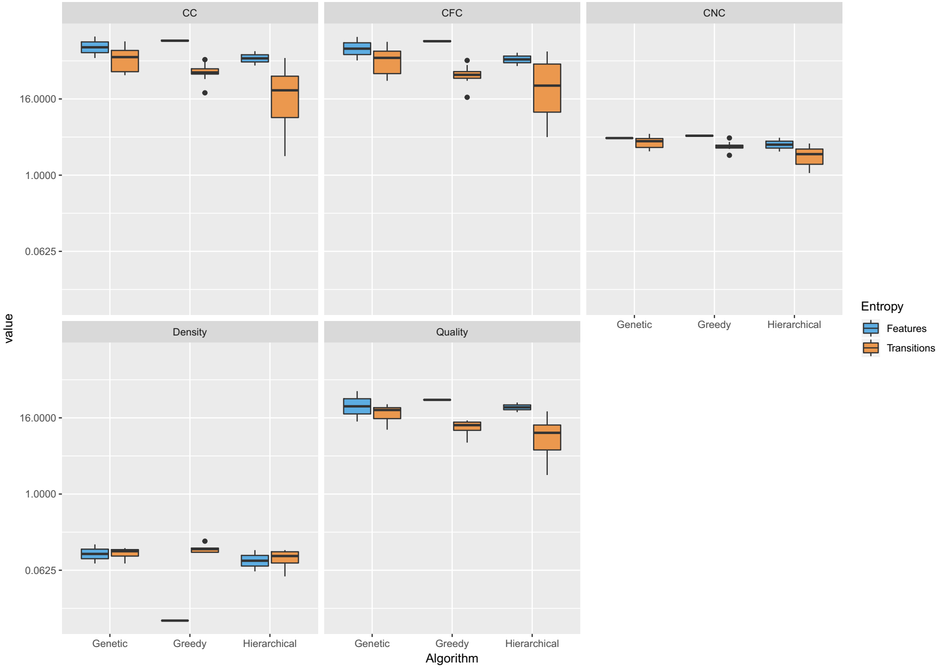


Fig. 12 Metrics distribution of Computer Configuration case study

of the metrics for the *Computer Configuration* case are significantly smaller than in the other cases, which can mean that the workflows obtained from this example are less complex. On the other hand, in light of the results shown in the chart, we can realise that, for these cases, the *Density* and *CNC* metrics do not help to discriminate the real level of complexity of the workflows produced with COLOSSI, since they present the same average value for the three cases. Nevertheless, the *Quality*, *CC* and *CFC* metrics present the highest values for the *Smart Farm* case, which could lead to higher levels of complexity and connectivity. It can also be inferred that *Computer Configuration* may be the most balanced case since all its edges have approximately the same length. In the case of *ERP*, it is remarkable how all its complexity can be focused on its high degree of connectivity, reaching the maximum of our evaluation. The fact that the *Smart Farm* example occupies a larger area between the Quality and CFC metrics denotes that, probably, this case contains a much greater number of features, transitions and splits than the other cases.

3.5 Threats to validity

Although the experiments presented in this paper provide evidence that demonstrates the validity of the proposed solution, in this section, we discuss the different threats to validity that affect the evaluation, derived from the assumptions that we made.

External validity The inputs used for the experiments presented in this paper were either realistic or designed to mimic realistic feature models. However, we do not control the development process and it may have errors and not encode every configuration for all case studies.

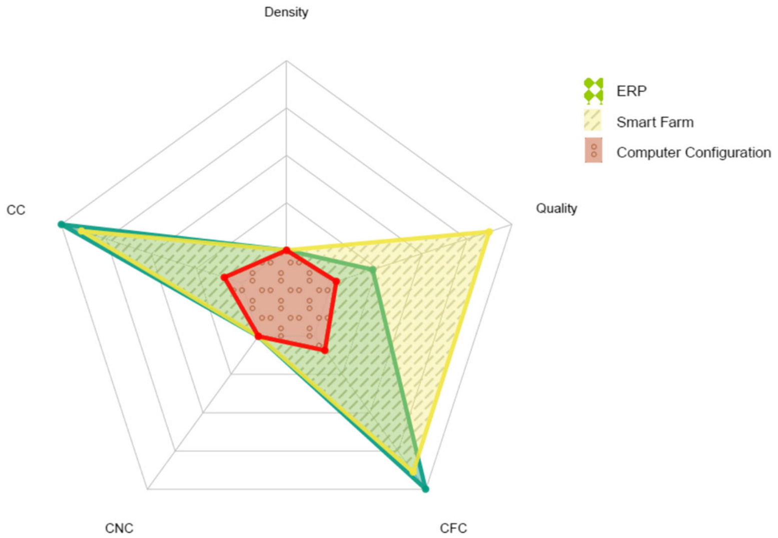


Fig. 13 Average of metrics of the configuration workflows

The major threats to external validity are:

- *Population validity*: the three examples that we used do not represent all configuration traces. Note that all of the models were provided after an anonymisation process. Moreover, the timestamps used to derive the traces were relying on the appearance within the input file without an explicit enumeration. To reduce these threats to validity, we chose some large models that were used in different studies in the literature. Also, we were not directly involved in the development of such models which
- *Ecological validity*: while external validity, in general, is focused on the generalisation of the results to other contexts (e.g., using other models), the ecological validity if focused on possible errors in the experiment materials and tools used. To avoid as much as possible such threats, we relied on previously existing algorithms to perform the process discovery.
- *Limitations depending on the input data*: another external validity problem lies with the shape and size of the input data. As previously stated, one of the important bottlenecks of this solution is the construction of the entropy matrix, since, without the matrix, it will not be possible to determine the optimal $k\text{-}cluster$, and therefore, to apply our solution.
- *Limitations on the use of feature attributes*: as mentioned in Section 2.1, features may contain attributes with more information, which has not been taken into account in our solution. Ignoring this data means that the selection of the same feature with different values in its attributes cannot be differentiated in the resulting workflows. This represents an important weakness of the approach since the use of this information would mean that different paths would be generated in the workflow for the same feature with different values in its attributes. While in our solution, it is considered as the same one.

Internal validity We developed several algorithms and metrics that reveal different properties of the workflows. To mitigate this threat, we have relied on a diversity of approaches. However, there might be characteristics of such workflows that are not revealed and further research should be developed. Also, another major threat to internal validity was the short number of models and configuration available in which we were able to test our techniques. In this case, we tried to cover all the models we found in the literature.

3.6 Examples of discovered configuration workflows

Derived from the high number of implementations and test cases tackled in this paper, every configuration workflow cannot be included in the document. However, for the sake of illustration, two configuration workflows of the ERP are shown in Figs. 14 and 15. Both figures represent the obtained clusters according to the dendrogram built by means of the entropy of features and transitions analysis respectively. In the case of entropy of features (cf., Fig. 14), five clusters are obtained. On the other hand, when the entropy of transition is used, three clusters are derived to split the configuration logs into simpler configuration workflows. In the following subsections, the details about the obtained configuration workflows and clusters are analysed. Moreover, every obtained cluster distribution for each algorithm, case study and type of entropy are available at http://www.idea.us.es/empiricalsoftware/.

4 Related work

This paper combines different research areas, for this reason, this section is structured to cover the main ones: configuration workflows, application for process mining and the variability in process mining.

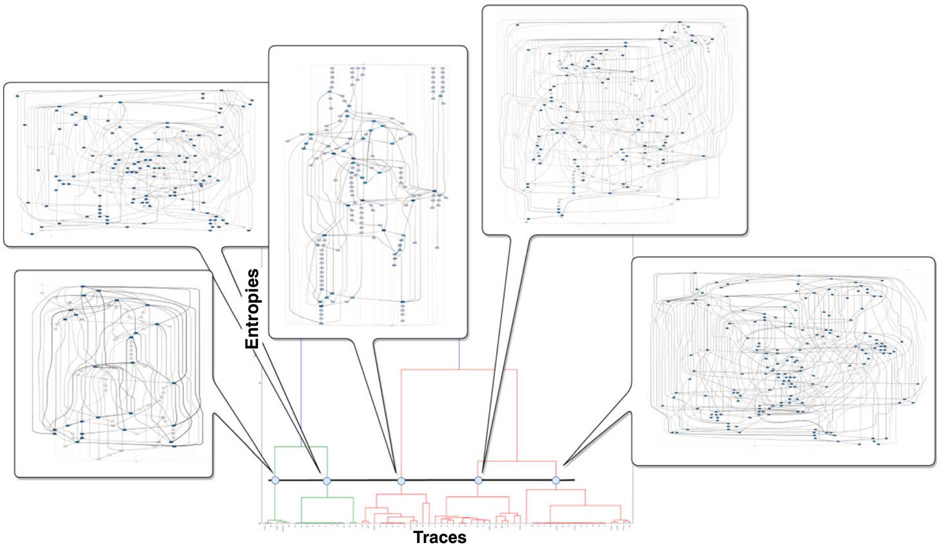


Fig. 14 Clustering for the ERP example using the *Entropy-features*

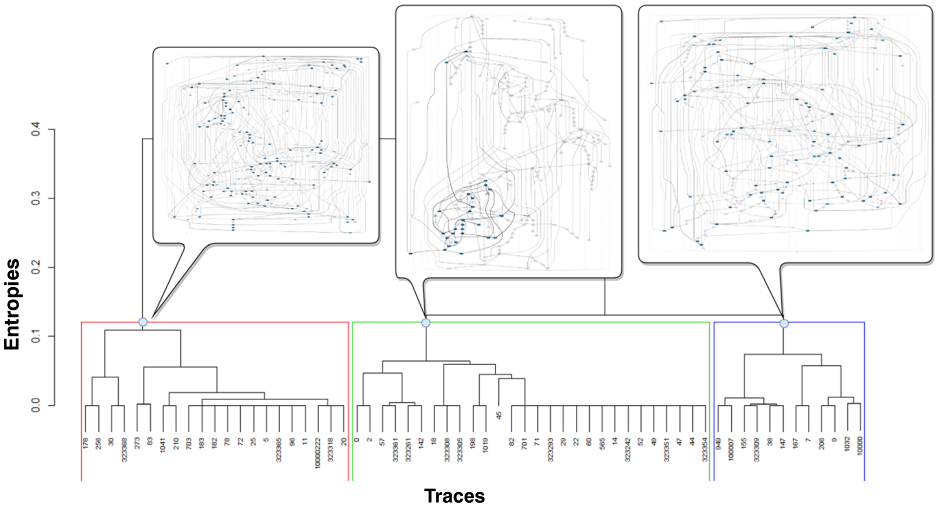


Fig. 15 Clustering for the ERP example using the *Entropy-transitions*

4.1 Configuration workflows

A formal description of configuration workflows is given in Hubaux et al. (2009). However, a configuration workflow is a bit different from our definition. The activity of the configuration workflow can be mapped to more than just a feature as in our case. However, our approach is complementary because in the handling process we can group different features as well. Furthermore, although formal semantics and automated support for configuration workflows are presented, no automated mechanism is developed to automatically generate configuration workflows from existing configuration logs. In that sense, our approach complements theirs.

Different possible feature orders are defined in Galindo et al. (2015). Those orders are used to build web–based configurators hiding the details of the concrete variability model flavours (e.g., OVM, FMs, CVL, etc.). The orders are built from the structure of the variability model. For instance, in the case of FMs, pre-order, post-order or in-order can be used to determine the feature order in which features are presented to the user. COLOSSI differs from this approach because we use as input configuration logs to automatically derive and cluster configuration workflows. Our approach can be complementary to Galindo et al. (2015) because different existing workflows could be also measured using process alignment metrics to determine what is the best feature order to be used.

There exist other approaches (Wang and Tseng 2011, 2014) focused on the field of product configurator design in which configuration workflows has been tackled from the perspective of machine learning. Neither of these two approaches uses clustering or process mining. Both are based on probabilistic estimations aided by a recommendation system that recalculates and guides the user through each step. This means that the user can lose the holistic view of the entire configuration process since from the step n he/she can only know the different possibilities that he/she can do in the step $n + 1$. With our proposal, the user has, from the beginning, an overall perspective of the entire configuration process from start to finish.

4.2 Application of process mining in different contexts

In order to discover the processes followed by users or systems analysing event logs, process mining has been applied in several scenarios. Depending on the scenario, different are the points of view that could be used to discover a process, such as the activities executed, persons involved, the resources used, the location where the actions occur, etc. The versatility of process mining techniques has brought about its application to several scenarios (Dakic et al. 2018), healthcare (Mans et al. 2009; Rozinat et al. 2009; Perimal-Lewis et al. 2016) and IT (Sahlabadi et al. 2014; Măruşter and van Beest 2009; Pérez-Álvarez et al. 2018; Fernández-Cerero et al. 2019) being the most active areas.

The case studies where event logs are produced by human behaviour interactions are especially complex, derived from the free-will capacity of the persons that is not always possible to be modelled. This is the context of this paper, where configuration tasks describe the interaction of users with systems. Previous examples in previous scenarios have been developed, such as Astromskis et al. (2015), to analyse how the users interact with an enterprise resource planning software, or the applicability of software scenarios analysing how the users interact with software to promote improvements about functional specifications or usability aspects (Rubin et al. 2014). Software development has also provided a complex scenario where process mining can provide a mechanism to improve and optimise the known as software process mining (Rubin et al. 2007). However, configurability issue has not been analysed before with process mining.

4.3 High variability in process mining

When there is a high human interaction, as in configuration processes, spaghetti and lasagne processes tend to be obtained. The occurrence of infrequent activities or non-repeated sequence of activities in the analysed log events brings about the necessity to apply frequency-based filtering solutions (Conforti et al. 2017) and other based on the discovery of a chaotic set of activities that can be frequent (Tax et al. 2019).

The infrequence patterns in process discovery are frequently treated as noise (Ly et al. 2012), being removed from the log traces to discover a process that represents the most frequent behaviour (Sani et al. 2017). Different types of filtering can be performed: (i) filtering the events that are not belong to the mainstream behaviour (Conforti et al. 2017; Sani et al. 2017); (ii) integrating the filtering as a part of the discovery (Leemans et al. 2014; Maruster et al. 2006; Weijters and Ribeiro 2011; vanden Broucke and Weerdt 2017); (iii) filtering traces, in an unsupervised (Ghionna et al. 2008) or supervised way (Cheng and Kumar 2015), and; (iv) including a previous steps for clustering the problem, facilitating the discrimination of traces according to different points of view or dividing different types of behaviour (de Leoni et al. 2016; Song et al. 2009; de Medeiros et al. 2007). In Sani et al. (2019) clustering techniques have been used to improve the quality of process models, but not to distribute the traces in different clusters.

To our knowledge, this paper, which is as an extension of Varela-Vaca et al. (2019a), is the first solution for workflow retrieval in SPL-related contexts. It is also an achievement the application of process mining techniques in new fields. This paper aims at promoting synergies between these two areas of study, consider different types of algorithms, metrics or entropies, and they are not oriented towards the consideration of the characteristics of configuration workflows.

Our contribution described in this paper intends to promote synergies between process mining and variability management and software product lines. We consider different types of algorithms, metrics and entropies.

5 Concluding remarks & future work

In this paper, we have coped with the problem of extracting the actual workflows used by SPL configurators by analysing configuration logs. To discover configuration workflows, we decided to rely on process mining techniques. Moreover, we proposed to apply clustering to improve the resulting configuration workflows reducing the complexity and improving their understandability. From our research on configuration workflows, we learned the following important lessons:

1. **Reduce the complexity of the configuration workflows.** We have defined a mechanism based on clustering to divide the configuration logs into smaller configuration groups to facilitate the understanding of the configuration workflows inferred from configuration logs.
2. **Quality measurement.** We have defined a set of metrics adapted from business process literature, to measure the quality of the obtained clusters and configuration workflows.
3. **Improving decisions about configurators.** The clustering creation and the analysis of the obtained configuration workflows provide information to expert users about the features that used to be configured together or sequential lists of features that could be integrated into a single feature. In this way, thanks to the structure of the workflow, the user will know that everything that appears in a parallel can be done simultaneously, while those that appear sequentially will indicate precedence relationships and restrictions.
4. **The selection of the clustering method depends mostly on the input data.** Generally, hierarchical clustering provides good solutions, in terms of quality, velocity and efficiency. However, it has an important bottleneck: the construction of the entropy matrix, which will determine the time and resources necessary to achieve the results. For the cases in which the construction of the entropy matrix becomes very hard, less accurate algorithms, such as greedy or genetic algorithms may be more suitable. The greedy algorithm provides very acceptable results consuming less time, and the genetic algorithm is positioned at an intermediate level for those cases in which results very close to the optimum, are needed. Finally, backtracking ensures optimal entropy minimisation and traces distribution. Unfortunately, the time and resources needed will increase exponentially with the size of the data of the case study. Lamentably, the use of these algorithms without the previous computation of the entropy matrix and the optimal $k\text{-}cluster$, will imply that the number of clusters chosen and the distribution of the traces, probably, will lead to a non-optimal assignment.
5. **The impact of the size and the morphology of the case study data on time and metrics.** Based on the results of the metrics in different examples, and the time and resources that are necessary to get the solutions of each one, there seems to exists a direct relationship among (i) features and the way in which these are related, (ii) the time and resources that are necessary to compute the entropy matrix and the

optimal $k - cluster$, (iii) the time consumed for each algorithm until the clusters are achieved, (iv) and the range of values in which the results of the metrics will be.

In future work, we plan to develop new variability-oriented metrics that can show the impact of the numbers of features within the workflows, trying to incorporate characteristics of the feature models into the clustering and process discovery. Additionally, we would like to incorporate the information present in the attributes of the features in the feature model to achieve more realistic workflows. Moreover, we would like to apply this technique to more scenarios and datasets to complement the validation of our proposal, including in the analysis of other methods to tackle spaghetti processes. Further, we consider interesting to investigate a proper way to obtain the best distribution clusters automatically for a defined number of clusters. In addition, the search of optimum $k - cluster$ is a very difficult task such as shown in our work, therefore it is necessary to find out a way to discover an approximate $k-cluster$ that ensures the achieved assignments not only are correct but they also are potential solutions. We propose to study new formulas of seeking $k - cluster$ based on the structure (i.e., number of task and/or gateways) of initial workflows before clustering. From our point of view, it is also relevant to propose multiple uses of the resulting workflows to help in different areas such as reverse engineering or SPL testing. Another proposal is to incorporate incorrect configuration to analyse the misalignment with the expected feature model.

Acknowledgements This work has been partially by the Ministry of Science and Technology of Spain through ECLIPSE (RTI2018-094283-B-C33) and OPHELIA (RTI2018-101204-B-C22) projects; the TASOVA network (MCIU-AEI TIN2017-90644-REDT); and the Junta de Andalucía via METAMORFOSIS projects, the European Regional Development Fund (ERDF/FEDER), and the MINECO Juan de la Cierva postdoctoral program.

Appendix : Quality metrics results

This appendix contains in the Tables 9, 10, 11, 12, and 13, the metric data represented in Figs. 10, 11, 12. To facilitate the interpretation of the data, the values of the metrics have been normalised in each metric, so that, all the results are between 0 and 1, allowing comparisons to be made. In addition, Table 8 is included to show the metric values for the original logs. With this, it can be seen how, in most cases, their values are closer to 0 after clustering, meaning that the resulting configuration workflows are also less complex. Still, it is important to note that it is very difficult to determine a generalisation regarding this data, since they are too domain-specific.

Table 8 Metrics for the initial logs of each case study

Case Study	Density	CC	CNC	CFC	Quality
ERP	0.05	0.49	0.96	0.49	0.26
Smart Farm	0	0.43	0.24	0.43	1
Computer Configuration	0.39	0.04	0.73	0.04	0.03

Table 9 Metrics for ERP case with entropy-features

	Algorithm	Density	CC	CNC	CFC	Quality
Cluster 1	Greedy	0.05	0.05	0.26	0.05	0.09
	Hierarchical	0.16	0.02	0.24	0.02	0.05
	Genetic	0.05	0.43	0.95	0.43	0.36
Cluster 2	Greedy	0.11	0.01	0.16	0.01	0.08
	Hierarchical	0.05	0.04	0.20	0.04	0.16
	Genetic	0.05	0.01	0.10	0.01	0.07
Cluster 3	Greedy	0.11	0.02	0.24	0.02	0.07
	Hierarchical	0.05	0.05	0.26	0.05	0.09
	Genetic	0.11	0.01	0.13	0.01	0.06
Cluster 4	Greedy	0.11	0.02	0.20	0.02	0.08
	Hierarchical	0.05	0.05	0.25	0.05	0.002
	Genetic	0.05	0.01	0.08	0.01	0.05
Cluster 5	Greedy	0.05	0.06	0.28	0.06	0.06
	Hierarchical	0.05	0.04	0.25	0.04	0.02
	Genetic	0.05	0.43	1	0.45	0.31

Table 10 Metrics for ERP case with entropy-transitions

	Algorithm	Density	CC	CNC	CFC	Quality
Cluster 1	Greedy	0.11	0.03	0.24	0.03	0.08
	Hierarchical	0.05	0.10	0.24	0.10	0.24
	Genetic	0.05	0.33	0.91	0.33	0.27
Cluster 2	Greedy	0.05	0.09	0.26	0.09	0.26
	Hierarchical	0.05	0.05	0.22	0.05	0.19
	Genetic	0.05	0.31	0.88	0.31	0.26
Cluster 3	Greedy	0.05	0.43	0.99	0.43	0.33
	Hierarchical	0.05	0.05	0.25	0.05	0.002
	Genetic	0.05	0.44	0.93	0.44	0.37

Table 11 Metrics for Smart Farm case with entropy-features

	Algorithm	Density	CC	CNC	CFC	Quality
Cluster 1	Greedy	0.05	0.80	0.92	0.80	0.82
	Hierarchical	0	0.33	0.23	0.33	0.12
	Genetic	–	–	–	–	–
Cluster 2	Greedy	0.05	0.84	0.96	0.84	0.83
	Hierarchical	0	0.25	0.25	0.25	0.90
	Genetic	–	–	–	–	–

Table 11 (continued)

	Algorithm	Density	CC	CNC	CFC	Quality
Cluster 3	Greedy	0.05	0.82	0.93	0.82	0.82
	Hierarchical	0	0.14	0.24	0.14	0.46
	Genetic	–	–	–	–	–
Cluster 4	Greedy	0.05	0.81	0.94	0.81	0.82
	Hierarchical	0	0.25	0.24	0.25	0.92
	Genetic	–	–	–	–	–
Cluster 5	Greedy	0.05	0.80	0.94	0.80	0.82
	Hierarchical	0.05	1	0.94	1	1
	Genetic	–	–	–	–	–
Cluster 6	Greedy	0.05	0.83	0.96	0.83	0.82
	Hierarchical	0.05	0.01	0.02	0.01	0.09
	Genetic	–	–	–	–	–

Table 12 Metrics for Computer Configuration case with entropy-features

	Algorithm	Density	CC	CNC	CFC	Quality
Cluster 1	Greedy	0.55	0.03	0.77	0.03	0.03
	Hierarchical	0.72	0.02	0.73	0.02	0.02
	Genetic	0.88	0.02	0.71	0.02	0.01
Cluster 2	Greedy	0.61	0.04	0.85	0.04	0.03
	Hierarchical	0.33	0.01	0.33	0.01	0.03
	Genetic	0.44	0.04	0.74	0.04	0.04

Table 13 Metrics for Computer Configuration case with entropy-transitions

	Algorithm	Density	CC	CNC	CFC	Quality
Cluster 1	Greedy	0.77	0.01	0.46	0.01	0.01
	Hierarchical	0.33	0.02	0.47	0.02	0.02
	Genetic	0.77	0.01	0.43	0.009	0.01
Cluster 2	Greedy	0.77	0.01	0.46	0.01	0.01
	Hierarchical	0.27	0.02	0.36	0.02	0.01
	Genetic	0.66	0.03	0.89	0.03	0.03
Cluster 3	Greedy	0.66	0.005	0.25	0.005	0.007
	Hierarchical	0.72	0.01	0.54	0.01	0.01
	Genetic	0.61	0.02	0.70	0.02	0.02
Cluster 4	Greedy	0.72	0.01	0.59	0.01	0.01
	Hierarchical	0.61	0.002	0.11	0.002	0.005
	Genetic	0.44	0.01	0.34	0.01	0.02

Table 13 (continued)

	Algorithm	Density	CC	CNC	CFC	Quality
Cluster 5	Greedy	0.66	0.01	0.42	0.01	0.01
	Hierarchical	0.5	0	0	0	0
	Genetic	0.77	0.02	0.79	0.02	0.02
Cluster 6	Greedy	1	0.02	0.73	0.02	0.01
	Hierarchical	0.55	0.001	0.07	0.001	0.002
	Genetic	0.72	0.02	0.73	0.02	0.02
Cluster 7	Greedy	0.66	0.01	0.44	0.01	0.01
	Hierarchical	0.66	0.08	0.34	0.008	0.01
	Genetic	0.5	0.02	0.5	0.02	0.02
Cluster 8	Greedy	0.77	0.009	0.39	0.009	0.01
	Hierarchical	0.72	0.004	0.22	0.004	0.006
	Genetic	0.72	0.01	0.41	0.01	0.01

References

Alférez M, Acher M, Galindo JA, Baudry B, Benavides D (2019) Modeling variability in the video domain: language and experience report. Softw Qual J 27(1):307–347

Astromskis S, Janes A, Mairegger M (2015) A process mining approach to measure how users interact with software: an industrial case study. In: Proceedings of the 2015 international conference on software and system process. ICSSP 2015. ACM, New York, pp 137–141

Augusto A, Conforti R, Dumas M, Rosa ML, Maggi FM, Marrella A, Mecella M, Soo A (2019) Automated discovery of process models from event logs: review and benchmark. IEEE Trans Knowl Data Eng 31(4):686–705. https://doi.org/10.1109/TKDE.2018.2841877

Baker FB, Hubert LJ (1975) Measuring the power of hierarchical cluster analysis. J Am Stat Assoc 70(349):31–38

Ball GH, Hall DJ (1965) Isodata a novel method of data analysis and pattern classification. Tech. rep. Stanford Research Inst, Menlo Park

Bosch J (2018) The three layer product model: an alternative view on spls and variability. In: Proceedings of the 12th international workshop on variability modelling of software-intensive systems, VAMOS 2018, Madrid, Spain, February 7–9, 2018, p 1. https://doi.org/10.1145/3168365.3168366

Caliński T, Harabasz J (1974) A dendrite method for cluster analysis. Commun Stat-Theory Methods 3(1):1–27

Cardoso J (2005) Control-flow complexity measurement of processes and weyuker's properties. In: 6th International enformatika conference, vol 8, pp 213–218

Cheng H, Kumar A (2015) Process mining on noisy logs—can log sanitization help to improve performance? Decis Support Syst 79:138–149. https://doi.org/10.1016/j.dss.2015.08.003

Conforti R, Rosa ML, ter Hofstede AHM (2017) Filtering out infrequent behavior from business process event logs. IEEE Trans Knowl Data Eng 29(2):300–314. https://doi.org/10.1109/TKDE.2016.2614680

Dakic D, Stefanovic D, Cosic I, Lolic T, Medojevic M (2018) Business application: a literature review. In: 29th DAAAM international symposium on intelligent manufacturing and automation. https://doi.org/10.2507/29th.daaam.proceedings.125

Davies DL, Bouldin DW (1979) A cluster separation measure. IEEE Trans Pattern Anal Mach Intell (2):224–227

de Leoni M, van der Aalst WMP, Dees M (2016) A general framework for correlating, predicting and clustering dynamic behavior based on event logs. Inf Syst 56:235–257. https://doi.org/10.1016/j.is.2015.07.003

de Medeiros AKA, Guzzo A, Greco G, van der Aalst WMP, Weijters AJMM, van Dongen BF, Saccà D (2007) Process mining based on clustering: a quest for precision. In: Business process management workshops, BPM 2007 international workshops, BPI, BPD, CBP, ProHealth, RefMod, semantics4ws, Brisbane, Australia, September 24, 2007, Revised Selected Papers, pp 17–29. https://doi.org/10.1007/978-3-540-78238-4_4

De Weerdt J, vanden Broucke S, Vanthienen J, Baesens B (2013) Active trace clustering for improved process discovery. IEEE Trans Knowl Data Eng 25(12):2708–2720

Derrac J, García S, Molina D, Herrera F (2011) A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evol Comput 1(1):3–18

Duda RO, Hart PE et al (1973) Pattern classification and scene analysis, vol 3. Wiley, New York

Dunn JC (1974) Well-separated clusters and optimal fuzzy partitions. J Cybern 4(1):95–104

Durán A, Benavides D, Segura S, Trinidad P, Ruiz-Cortés A (2017) Flame: a formal framework for the automated analysis of software product lines validated by automated specification testing. SOSYM 16(4):1049–1082. https://doi.org/10.1007/s10270-015-0503-z

Felfernig A, Walter R, Galindo JA, Benavides D, Erdeniz SP, Atas M, Reiterer S (2018) Anytime diagnosis for reconfiguration. J Intell Inf Syst 51(1):161–182. https://doi.org/10.1007/s10844-017-0492-1

Fernández-Cerero D, Varela-Vaca ÁJ, Fernández-Montes A, Gómez-López MT, Alvárez-Bermejo JA (2019) Measuring data-centre workflows complexity through process mining: the google cluster case. J Supercomput. https://doi.org/10.1007/s11227-019-02996-2

Ferreira DR, Alves C (2011) Discovering user communities in large event logs. In: Daniel F, Barkaoui K, Dustdar S (eds) Business process management workshops—BPM 2011 international workshops, Clermont-Ferrand, France, August 29, 2011, Revised Selected Papers, Part I, Springer, Lecture Notes in Business Information Processing, vol 99, pp 123–134. https://doi.org/10.1007/978-3-642-28108-2_11

Frey T, Van Groenewoud H (1972) A cluster analysis of the d2 matrix of white spruce stands in saskatchewan based on the maximum-minimum principle. J Ecol 60(3):873–886

Galindo J, Turner H, Benavides D, White J (2014a) Testing variability-intensive systems using automated analysis: an application to android. Softw Qual J 1–41. https://doi.org/10.1007/s11219-014-9258-y

Galindo JA, Alférez M, Acher M, Baudry B, Benavides D (2014b) A variability-based testing approach for synthesizing video sequences. In: International symposium on software testing and analysis, ISSTA '14, San Jose, CA, USA—July 21–26, 2014, pp 293–303

Galindo J, Dhungana D, Rabiser R, Benavides D, Botterweck G, Grünbacher P (2015) Supporting distributed product configuration by integrating heterogeneous variability modeling approaches. Inf Softw Technol 62(1):78–100

Galindo JA, Benavides D, Trinidad P, Gutiérrez-Fernández AM, Ruiz-Cortés A (2018) Automated analysis of feature models: Quo vadis? Computing 101:387–433

Ghionna L, Greco G, Guzzo A, Pontieri L (2008) Outlier detection techniques for applications. In: Foundations of intelligent systems. Springer, Berlin, pp 150–159

Grabusts P et al (2011) The choice of metrics for clustering algorithms. In: Proceedings of the 8th international scientific and practical conference, vol 2, pp 70–76

Greco G, Guzzo A, Pontieri L, Sacca D (2006) Discovering expressive process models by clustering log traces. IEEE Trans Knowl Data Eng 18(8):1010–1027

Halkidi M, Vazirgiannis M, Batistakis Y (2000) Quality scheme assessment in the clustering process. In: European conference on principles of data mining and knowledge discovery. Springer, pp 265–276

Hartigan JA (1975) Clustering algorithms, 99th, John Wiley & Sons, Inc., USA

Hompes BFA, Verbeek HMW, van der Aalst WMP (2015) Finding suitable activity clusters for decomposed process discovery. In: Ceravolo P, Russo B, Accorsi R (eds) Data-driven process discovery and analysis. Springer International Publishing, Cham, pp 32–57

Hompes BFA, Buijs JCAM, van der Aalst WMP, Dixit PM, Buurman J (2017) Detecting changes in process behavior using comparative case clustering. In: Ceravolo P, Rinderle-Ma S (eds) Data-driven process discovery and analysis. Springer International Publishing, pp 54–75

Hubaux A, Classen A, Heymans P (2009) Formal modelling of feature configuration workflows. In: Proceedings of the 13th international software product line conference, Carnegie Mellon University, Pittsburgh, PA, USA, SPLC '09, pp 221–230. http://dl.acm.org/citation.cfm?id=1753235.1753266

Hubaux A, Heymans P, Schobbens PY, Deridder D, Abbasi E (2013) Supporting multiple perspectives in feature-based configuration. SOSYM 12(3):641–663. https://doi.org/10.1007/s10270-011-0220-1. http://www.scopus.com/inward/record.url?eid=2-s2.0-84879788174&partnerID=40&md5=dee1ff6a27f859c32d424a1528d81ada

Hubert L (1974) Approximate evaluation techniques for the single-link and complete-link hierarchical clustering procedures. J Am Stat Assoc 69(347):698–704

Hubert LJ, Levin JR (1976) A general statistical framework for assessing categorical clustering in free recall. Psychol Bull 83(6):1072

Jain AK, Murty MN, Flynn PJ (1999) Data clustering: a review. ACM Comput Surv 31(3):264–323

Kobren A, Monath N, Krishnamurthy A, McCallum A (2017) A hierarchical algorithm for extreme clustering. In: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining. KDD '17. ACM, New York, pp 255–264

Krzanowski WJ, Lai Y (1988) A criterion for determining the number of groups in a data set using sum-of-squares clustering. Biometrics 44(1):23–34

Kuiper FK, Fisher L (1975) 391: a Monte Carlo comparison of six clustering procedures 777–783. Biometrics 31(3):777–783

Lebart L, Morineau A, Piron M (2000) Statistique exploratoire multidimensionnelle, Dunod, Paris, France

Leemans SJJ, Fahland D, van der Aalst WMP (2014) Discovering block-structured process models from incomplete event logs. In: Petri Nets, Springer, Lecture Notes in Computer Science, vol 8489, pp 91–110

Leemans SJJ, Fahland D, van der Aalst WMP (2015) Scalable process discovery with guarantees. In: Gaaloul K, Schmidt R, Nurcan S, Guerreiro S, Ma Q (eds) Enterprise, business-process and information systems modeling. Springer International Publishing, Cham, pp 85–101

Lettner M, Rodas-Silva J, Galindo JA, Benavides D (2019) Automated analysis of two-layered feature models with feature attributes. J Comput Lang 51:154–172

Ly LT, Indiono C, Mangler J, Rinderle-Ma S (2012) Data transformation and semantic log purging for process mining. In: CAiSE, Springer, Lecture notes in computer science, vol 7328, pp 238–253

MacKay DJC (2002) Information theory inference & learning algorithms. Cambridge University Press, New York

Makanju A, Brooks S, Zincir-Heywood AN, Milios EE, Safavi-Naini R (2008) Logview: visualizing event log clusters. In: Korba L, Marsh S (eds) Sixth annual conference on privacy, security and trust, PST 2008, October 1–3, 2008. IEEE Computer Society, Fredericton, pp 99–108. https://doi.org/10.1109/PST.2008.17

Makanju A, AN Zincir-Heywood, Milios EE (2009) Clustering event logs using iterative partitioning. In: IV JFE, Fogelman-Soulié F, Flach PA, Zaki MJ (eds) Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining, Paris, France, June 28–July 1, 2009. ACM, pp 1255-1264. https://doi.org/10.1145/1557019.1557154

Mans RS, Schonenberg MH, Song M, van der Aalst WMP, Bakker PJM (2009) Application of process mining in healthcare—a case study in a dutch hospital. In: Fred A, Filipe J, Gamboa H (eds) Biomedical engineering systems and technologies. Springer, Berlin, pp 425–438

Măruşter L, van Beest NRTP (2009) Redesigning business processes: a methodology based on simulation and techniques. Knowl Inf Syst 21(3):267. https://doi.org/10.1007/s10115-009-0224-0

Maruster L, Weijters AJMM, van der Aalst WMP, van den Bosch A (2002) Process mingin: discovering direct successors in process logs. In: Discovery Science, 5th international conference, DS 2002, Lübeck, Germany, November 24–26, 2002, Proceedings, pp 364–373. https://doi.org/10.1007/3-540-36182-0_37

Maruster L, Weijters AJMM, van der Aalst WMP, van den Bosch A (2006) A rule-based approach for process discovery: dealing with noise and imbalance in process logs. Data Min Knowl Discov 13(1):67–87

McClain JO, Rao VR (1975) Clustisz: a program to test for the quality of clustering of a set of objects. JMR. J Market Res (pre-1986) 12(000004):456

Mendling J (2008) Metrics for business process models. Springer, Berlin, pp 103–133

Milligan GW (1980) An examination of the effect of six types of error perturbation on fifteen clustering algorithms. Psychometrika 45(3):325–342

Milligan GW (1981) A monte carlo study of thirty internal criterion measures for cluster analysis. Psychometrika 46(2):187–199

Murtagh F (1983) A survey of recent advances in hierarchical clustering algorithms. Comput J 26(4):354–359. https://doi.org/10.1093/comjnl/26.4.354. http://oup.prod.sis.lan/comjnl/article-pdf/26/4/354/1072603/26-4-354.pdf

Pereira JA, Matuszyk P, Krieter S, Spiliopoulou M, Saake G (2016a) A feature-based personalized recommender system for product-line configuration. In: Proceedings of the international conference on generative programming: concepts and experiences. ACM, pp 120–131

Pereira JA, Matuszyk P, Krieter S, Spiliopoulou M, Saake G (2016b) A feature-based personalized recommender system for product-line configuration. In: Proceedings of the international conference on generative programming: concepts and experiences. ACM, pp 120–131

Pereira JA, Schulze S, Figueiredo E, Saake G (2018a) N-dimensional tensor factorization for self-configuration of software product lines at runtime. In *Proceedings of the 22nd International Systems and Software Product Line Conference - Volume 1 (SPLC '18)*. Association for Computing Machinery, New York, NY, USA, 87–97. https://doi.org/10.1145/3233027.3233039

Pereira JA, Matuszyk P, Krieter S, Spiliopoulou M, Saake G (2018b) Personalized recommender systems for product-line configuration processes. Comput Lang Syst Struct 54:451–471

Pérez-Álvarez JM, Maté A, López MTG, Trujillo J (2018) Tactical business-process-decision support based on kpis monitoring and validation. Comput Ind 102:23–39

Pérez-Castillo R, Fernéndez-Ropero M, Piattini M (2019) Business process model refactoring applying ibuprofen. An industrial evaluation. J Syst Softw 147:86–103

Perimal-Lewis L, Teubner D, Hakendorf P, Horwood C (2016) Application of process mining to assess the data quality of routinely collected time-based performance data sourced from electronic health records by validating process conformance. Health Inform J 22(4):1017–1029

Ratkowsky D, Lance G (1978) Criterion for determining the number of groups in a classification Vol. 44, No. 1, pages 23-34

Rodas-Silva J, Galindo JA, García-Gutiérrez J, Benavides D (2019) Selection of software product line implementation components using recommender systems: an application to wordpress. IEEE Access 7:69226–69245

Rohlf FJ (1974) Methods of comparing classifications. Annu Rev Ecol System 5(1):101–113

Rozinat A, de Jong ISM, Günther CW, van der Aalst WMP (2009) Process mining applied to the test process of wafer scanners in ASML. IEEE Trans Syst Man Cybern Part C 39(4):474–479

Rubin V, Günther CW, van der Aalst WMP, Kindler E, van Dongen BF, Schäfer W (2007) Process mining framework for software processes. In: Wang Q, Pfahl D, Raffo DM (eds) Software process dynamics and agility. Springer, Berlin, pp 169–181

Rubin VA, Mitsyuk AA, Lomazova IA, van der Aalst WMP (2014) Process mining can be applied to software too! In: Proceedings of the 8th ACM/IEEE international symposium on empirical software engineering and measurement. ESEM '14. ACM, New York, pp 57:1–57:8

Sahlabadi M, Muniyandi R, Shukur Z (2014) Detecting abnormal behavior in social network websites by using a process mining technique. J Comput Sci 10(3):393–402. https://doi.org/10.3844/jcssp.2014.393.402

Sani MF, van Zelst SJ, van der Aalst WMP (2017) Improving process discovery results by filtering outliers using conditional behavioural probabilities. In: Business process management workshops—BPM 2017 international workshops, Barcelona, Spain, September 10–11, 2017, Revised Papers, pp 216–229. https://doi.org/10.1007/978-3-319-74030-0_16

Sani MF, Boltenhagen M, van der Aalst W (2019) Prototype selection based on clustering and conformance metrics for model discovery. https://arxiv.org/pdf/1912.00736.pdf

Schobbens P, Heymans P, Trigaux J, Bontemps Y (2007) Generic semantics of feature diagrams. Comput Netw 51(2):456–479. https://doi.org/10.1016/j.comnet.2006.08.008

She S, Lotufo R, Berger T, Wasowski A, Czarnecki K (2010) The variability model of the linux kernel. In: VAMOS, vol 10, pp 45–51

Song M, Günther CW, van der Aalst WMP (2008) Trace clustering in process mining. In: Ardagna D, Mecella M, Yang J (eds) Business process management workshops, BPM 2008 international workshops, Milano, Italy, September 1–4, 2008. Revised Papers, Springer, Lecture Notes in Business Information Processing, vol 17, pp 109–120. https://doi.org/10.1007/978-3-642-00328-8_11

Song M, Günther CW, van der Aalst WMP (2009) Trace clustering in. In: Ardagna D, Mecella M, Yang J (eds) Business Process Management Workshops. Springer, Berlin, pp 109–120

Tax N, Sidorova N, van der Aalst WMP (2019) Discovering more precise process models from event logs by filtering out chaotic activities. J Intell Inf Syst 52(1):107–139. https://doi.org/10.1007/s10844-018-0507-6

Thüm T, Apel S, Kästner C, Schaefer I, Saake G (2014) A classification and survey of analysis strategies for software product lines. ACMCS 47(1). https://doi.org/10.1145/2580950

Valencia-Parra A, Ramos-Gutiérrez B, Varela-Vaca AJ, López MTG, Bernal AG (2019a) Enabling process mining in aircraf manufactures: extracting event logs and discovering processes from complex data. In: Proceedings of the industry forum at BPM 2019 co-located with 17th international conference on business process management (BPM 2019), Vienna, Austria, September 1–6, 2019, pp 166–177

Valencia-Parra Á, Varela-Vaca ÁJ, Gómez-López MT, Ceravolo P (2019b) CHAMALEON: framework to improve data wrangling with complex data. In: Proceedings of the 40th international conference on information systems, ICIS 2019, Munich, Germany, December 15–18, 2019

van der Aalst WMP (2011) Analyzing "spaghetti processes". Springer, Berlin

van der Aalst WMP (2016) Process mining–data science in action, 2nd edn. Springer, Berlin

van Dongen BF, de Medeiros AKA, Verbeek HMW, Weijters AJMM, van der Aalst WMP (2005) The prom framework: a new era in process mining tool support. In: Applications and theory of Petri nets 2005, 26th international conference, ICATPN 2005, Miami, USA, June 20–25, 2005, Proceedings, pp 444–454. https://doi.org/10.1007/11494744_25

vanden Broucke SKLM, Weerdt JD (2017) Fodina: a robust and flexible heuristic process discovery technique. Decis Support Syst 100:109–118. https://doi.org/10.1016/j.dss.2017.04.005

Varela-Vaca AJ, Gasca RM (2013) Towards the automatic and optimal selection of risk treatments for business processes using a constraint programming approach. Inf Softw Technol 55(11):1948–1973

Varela-Vaca ÁJ, Galindo JA, Ramos-Gutiérrez B, Gómez-López MT, Benavides D (2019a) Process mining to unleash variability management: discovering configuration workflows using logs. In: Proceedings of the 23rd International Systems and Software Product Line conference, SPLC 2019, Volume A, Paris, France, September 9–13, 2019, pp 37:1–37:12

Varela-Vaca ÁJ, Gasca RM, Ceballos R, Gómez-López MT, Torres PB (2019b) Cyberspl: a framework for the verification of cybersecurity policy compliance of system configurations using software product lines. Applied Sciences 9(24). https://doi.org/10.3390/app9245364. https://www.mdpi.com/2076-3417/9/24/5364

Wang Y, Tseng MM (2011) Adaptive attribute selection for configurator design via shapley value. Artif Intell Eng Des Anal Manuf 25(2):185–195. https://doi.org/10.1017/S0890060410000624

Wang Y, Tseng M (2014) Attribute selection for product configurator design based on gini index. Int J Prod Res 52(20):6136–6145. https://doi.org/10.1080/00207543.2014.917216

Ward JH Jr (1963) Hierarchical grouping to optimize an objective function. J Am Stat Assoc 58(301):236–244

Weijters AJMM, Ribeiro JTS (2011) Flexible heuristics miner (FHM). In: CIDM. IEEE, pp 310–317

Wilcoxon F (1946) Individual comparisons of grouped data by ranking methods. J Econ Entomol 39(2):269–270

XES (2016) IEEE Standard for eXtensible Event Stream (XES) for achieving interoperability in event logs and event streams. IEEE Std 1849-2016 pp 1–50. https://doi.org/10.1109/IEEESTD.2016.7740858

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

Affiliations

Belén Ramos-Gutiérrez[1] · **Ángel Jesús Varela-Vaca**[1] · **José A. Galindo**[1] ·
María Teresa Gómez-López[1] · **David Benavides**[1]

Ángel Jesús Varela-Vaca
ajvarela@us.es

José A. Galindo
jagalindo@us.es

María Teresa Gómez-López
maytegomez@us.es

David Benavides
benavides@us.es

[1] Data-Centric Computing Research Hub (IDEA), Universidad de Sevilla, Seville, Spain

3.2.3 Self-adaptative troubleshooting for to guide resolution of malfunctions in Aircraft Manufacturing

Published in IEEE Access, Vol. 9 (9380188), pp. 42707-42723, 2021, IEEE.

- ***Authors**: Belén Ramos Gutiérrez, María Teresa Gómez López, Diana Borrego, Rafael Ceballos, Rafael M. Gasca, Antonio Barea.*
- ***DOI**: `https://doi.org/10.1109/ACCESS.2021.3066253`.*
- ***Rating**: Q2 (JIF'21 3.476).*

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 11, 2021, accepted March 11, 2021, date of publication March 17, 2021, date of current version March 23, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3066253

Self-Adaptative Troubleshooting for to Guide Resolution of Malfunctions in Aircraft Manufacturing

BELÉN RAMOS-GUTIÉRREZ[1], MARÍA TERESA GÓMEZ-LÓPEZ[1], DIANA BORREGO[1], RAFAEL CEBALLOS[1], RAFAEL M. GASCA[1], AND ANTONIO BAREA[2]

[1]Departamento de Lenguajes y Sistemas Informáticos, Escuela Técnica Superior de Ingeniería Informática, Universidad de Sevilla, 41012 Sevilla, Spain
[2]Airbus Defense and Space San Pablo Sur, 41020 Sevilla, Spain

Corresponding author: Belén Ramos-Gutiérrez (brgutierrez@us.es)

This work was supported by the Project (RTI2018-094283-B-C33), funded by: FEDER/Ministry of Science and Innovation - State Research Agency.

ABSTRACT The increasing complexity of systems and the heterogeneous origin of the possible malfunctions bring about the necessity of redefining the troubleshooting processes. Troubleshooting comprises the set of steps for the systematic analysis of the symptoms after the detection of a malfunction. The complexity of certain systems, such as aircraft, means the origin of that malfunction can be any of several reasons, where diagnosis techniques support engineers in determining the reason for the unexpected behaviour. However, derived from the high number of components involved in an aircraft, the list of possible fault origins can be extremely long, and the analysis of every element on the list, until the element responsible is found, can be very time-consuming and error-prone. As an alternative, certain input/output signals can be read to prevent the substitution of a correctly functioning component, by validating its behaviour in an indirect way. In order to optimise the actions to perform, we have identified the relevant parts of the model to propose a troubleshooting process to ascertain the signals to read and the components to substitute, while striving to minimise the action cost in accordance with a combination of structural analysis, the probability of malfunction associated to the components, and the cost associated to each extra signal read and component substituted. The proposal has been validated in a system taken from a real scenario obtained in collaboration with the Airbus Defence and Space company. A statistical analysis of the degree of improvement of the troubleshooting process has also been included.

INDEX TERMS Decision-making process, model-based diagnosis, multi-objective function, troubleshooting.

I. INTRODUCTION

Fault diagnosis provides mechanisms to isolate the element responsible for a malfunction in accordance with a set of observations [1], [2]. However, signal values provided by monitoring systems are not always sufficient to isolate a suitable subset of possible components responsible for a malfunction, since a great number of root causes can be involved. Frequently, probability and cost are included in sorting the list of possible causes, although the number of possibilities can be extremely high. One option for the reduction of the number of candidates involves improving the observation of the system by developing a set of new readings of signals in the system [3], [4]. Troubleshooting is the reasoning process for the determination of what part of a system is causing a malfunction in accordance with a set of actions that guide the diagnoser to isolate the origin of misbehaviour. It is fundamental both to decrease the time taken to solve the problem and reduce the cost associated to reading signals of the system and to the component substitutions. Both aspects are especially important in contexts where there are numerous possible combinations of malfunctions. Aeronautic organisations can include a high number of components that are interconnected, such as the distribution of sensors to monitor

The associate editor coordinating the review of this manuscript and approving it for publication was Fatih Emre Boran.

the assembly process [5], and the Cyber-Physical Systems interaction with the rest of the software used in the company [6], [7] for troubleshooting and maintenance purposes [8].

The detection, isolation, and repair of components during aircraft assembly and in-service processes are very problematic [9], derived from the complexity of the systems, for both the number of components and the density of relations between them. The growing complexity in the systems and the heterogeneity of the failure nature, makes necessary to tackle the problem isolation improving the troubleshooting mechanisms [10]. During the aircraft assembly and in-service maintenance processes, several malfunctions might be found, which is the reason why a number of tests need to be applied to prevent the propagation of faults to other phases of the assembly. Both detection and resolution of possible malfunctions during the tests are fundamental in reducing the number of incidences before the aircraft is ready, thereby enhancing the product and process quality. Typically, the phases involved in the methodologies used in troubleshooting in systems such as aircraft include [11]: visual inspection, operational evaluation, problem classification, problem isolation, problem location, problem resolution, and a final operational evaluation to ensure the resolution of the problem. In other more specific cases, such as in-service support, the troubleshooting methodologies are complex and frequently, hand the responsibility of decision-making to the experts, instead of defining an ad-hoc set of steps for each case where both probability and cost are taken into account [12]. In complex contexts, such as aeronautic scenarios, for an observed malfunction there are several possible root causes, owing to the high connectivity between the components. This is why the visual or manual inspections, following a set of predefined steps, do not always constitute the optimal combination of activities to isolate the malfunction. This is why, it is considered interesting to create customised troubleshooting for each observed malfunction that has been adapted to the observations at each moment.

In this paper, we propose a framework to support the troubleshooting process after the detection of a malfunction in order to isolate the component responsible from a sorted list by automating the steps as much as possible, and guiding the operator minimising the cost of detection and that of the component substitutions. The method is based on the incorporation of the fault probability, substitution cost, the cost of the reading and the structural analysis, thereby creating a customised solution of each malfunction and sequence of signal. Since, for each element of the model, both probability and substitution cost must be combined, it is necessary to incorporate a function to optimise multiple objectives.

In order to validate our proposal, we have developed a solution that has been tested on the company Airbus Defence and Space. The concept of this proposal started as part of the context of the European project *Clean Sky 2*, and contributes to one of the tasks aligned with the key societal challenge for smart, green and integrated transport defined in Horizon 2020. The *Clean Sky 2* project built a full-scale in-flight demonstrator of innovative architectures and configurations to contribute toward advances in environmental and economic performance and to bring crucial competitiveness benefits to European industry. Through further initiatives, the final proposal was released and validated by the Airbus Defence and Space Manufacturing Engineering Transversal Systems department. A statistical analysis has been incorporated to demonstrate how the use of our proposal can reduce the troubleshooting effort.

The paper is organised as follows: Section II introduces the models of problems where our proposal can be applied. Section III presents the methodology proposed in this paper to be applied to the defined model. Section IV shows step by step, and in great detail, the different actions that would be carried out and the final result that is achieved with our proposal. Section V lays out the evaluation of our proposal and the provided advantages in a real scenario, as is the aircraft assembly process. Section VI shows the software architecture of our proposal. Section VII summarises the previous proposals in the area, and the reason why our solution implies a step forward in the literature. Finally, conclusions are drawn and future work is outlined.

II. TROUBLESHOOTING MODEL DESCRIPTION

The systems where troubleshooting techniques can be applied are those formed by a set of components interconnected by links. Both components and links can fail, but their main difference is that the links can be read to obtain information about the system behaviour. Those links can be automatically readable by computers (without any extra-cost) or through human action (implying cost and time). When an incorrect behaviour is triggered, it is necessary to find the elements responsible for the malfunction and to perform its substitution. In this section, the model of the systems supported by our proposal are formalised and an example is introduced which allows us to detail and explain the proposed methodology. Figure 1 shows a model, consisting of 18 components connected through 26 links, that represents a simplified model from a particular aircraft system. The same model can include various operational situations or configurations, called *operational modes* which perform specific operational scenarios, thereby activating a subset of their components depending on the mode.

Formalising these ideas, in our proposal, a **system model** involved in a troubleshooting process is described as the tuple $\langle E, L, OM, pf \rangle$, where E is a set of component elements, L represents the link elements that connect the component elements, OM is a set of operational modes where different parts of the systems are involved according to the operations executed in each case, and pf is the priority function which describes the importance between substitution cost, reading cost and malfunction probability of the elements to be sorted in the troubleshooting process. Each part of the system is detailed in the following subsections.

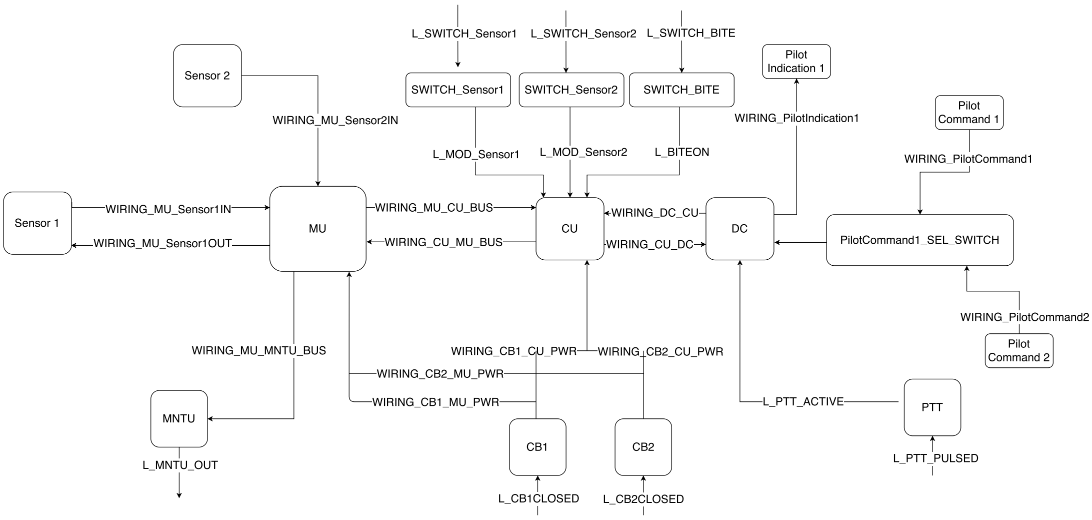


FIGURE 1. System example for troubleshooting.

A. COMPONENT ELEMENTS (E)

A system model consists of a set of component elements, each of which is described by an identifying name. The component elements of the example in Figure 1 are shown in Table 1.

TABLE 1. Component elements.

COMPONENT ELEMENT NAMES		
MU	CU	DC
PilotCommand1	PilotCommand2	PilotIndication1
CB1	CB2	Sensor1
MNTU	Sensor2	PTT
PilotCommand1_SEL_SWITCH	SWITCH_Sensor2	SWITCH_Sensor1
SWITCH_BITE	SWITCH_CB1	SWITCH_CB2

B. LINK ELEMENTS (L)

The links represent elements that connect component elements, and describe the inputs and outputs of the link. Each link is represented by the tuple ⟨*Name*: String, *Origin*: Component, *Destination*: Component, *Domain*: Integer/Real/Boolean, *Observability*: Boolean⟩. The *name* represents the link in a univocal way, *origin* is the component that produces the value that is assigned to a *variable* associated to the input or output of the link, *destination* is the component that receives the value of the variable transported by the link, and the *observability* is associated to the automatic readability of the link. The links can also fail, and therefore be substituted as a consequence of the detection of a malfunction, but the main difference with the components is that the values that are read in the links offer information regarding the complete behaviour of the element, since only one output is possible in a link.

According to the value of the *observability* (*true* or *false*) described in the previous tuple, the input or output variables of the links are defined as *known* or *readable* variables. In the model, the non-observable variables (i.e., internal to components) are not included since the components are presented as a black box.

- **Known variable**. When *observability* is *true*, the value of the variable associated to the link is known by the automatic monitorisation of the system, and its reading does not entail any extra effort, and the reading *cost* is equal to 0. Furthermore, in a particular way, if a link does not have an origin or destination component, then the *observability* of its associated variable is mandatorily *true* (it represents an input or output variable).
- **Readable variables**. When *observability* is *false*, the reading of the variables requires human intervention, thereby entailing an extra effort that requires labour time, which involves a reading *cost* greater than 0.

Link elements of the example in Figure 1 are shown in Table 2. In this example, links without origin correspond to input variables, and links without destination represent output signals.

C. OPERATIONAL MODE (OM)

Frequently, a single system can be utilised to support several different behaviours. Therefore, and depending on the specific operational mode in which the system is operating, on detecting a malfunction, it is not necessary to analyse all the elements of the system, but only those involved in that mode. This is especially important in the assembly process, where different parts of the system are involved in each test. An operational mode represents a state of the system model

TABLE 2. Link elements for the example.

NAME	ORIGIN	DESTINATION	DOMAIN	OBSERVABILITY
WIRING_MU_Sensor1IN	Sensor1	MU	*Boolean*	*False*
WIRING_MU_Sensor2IN	Sensor2	MU	*Boolean*	*False*
WIRING_CB1_MU_PWR	CB1	MU	*Boolean*	*False*
WIRING_CB2_MU_PWR	CB2	MU	*Boolean*	*False*
WIRING_CB1_CU_PWR	CB1	CU	*Boolean*	*False*
WIRING_CB2_CU_PWR	CB2	CU	*Boolean*	*False*
WIRING_CU_DC	CU	DC	*Boolean*	*False*
WIRING_DC_CU	DC	CU	*Boolean*	*False*
WIRING_PilotCommand2	PitloCommand2	DC	*Boolean*	*False*
L_PTT_ACTIVE	PTT	DC	*Boolean*	*False*
L_MOD_Sensor1	SWITCH_Sensor1	CU	*Boolean*	*False*
L_MOD_Sensor2	SWITCH_Sensor2	CU	*Boolean*	*False*
L_BITEON	SWITCH_BITE	CU	*Boolean*	*False*
WIRING_MU_CU_BUS	MU	CU	*Boolean*	*False*
WIRING_CU_MU_BUS	CU	MU	*Boolean*	*False*
WIRING_MU_MNTU_BUS	MU	MNTU	*Boolean*	*True*
WIRING_MU_Sensor1OUT	MU	Sensor1	*Boolean*	*True*
WIRING_PilotIndication1	DC	PilotIndication1	*Boolean*	*True*
WIRING_PilotCommand	PilotCommand1	DC	*Boolean*	*True*
L_CB1CLOSED		SWITCH_CB1	*Boolean*	*True*
L_CB2CLOSED		SWTICH_CB2	*Boolean*	*True*
L_SWITCH_Sensor2		SWITCH_Sensor2	*Boolean*	*True*
L_SWITCH_Sensor1		SWITCH_Sensor1	*Boolean*	*True*
L_SWITCH_BITE		SWITCH_BITE	*Boolean*	*True*
L_PTT_PULSED		PTT	*Boolean*	*True*
L_MNTU_OUT	MNTU		*Boolean*	*True*

where the system performs a specific operational scenario, for which the part of the model and a specific set of the components and link elements are active in each operation. It is described by the tuple $\langle name, E_{om}, L_{om}\rangle$ which represents the *name* of the operational mode, a subset of the components $E_{om} \subseteq E$, and a subset of the links $L_{om} \subseteq L$. Depending on the OM, different values for the probabilities of fault, reading cost, and substitution cost are assigned to each element (both components and links). Moreover, since the element links can be observed, they also have an observational cost that depends on the OM, derived from the difficulty to access a link according to the stage of the assembly. In summary, the E_{om} and L_{om} are described as the tuples: $E_{om} = \langle E, prob, sub\_cost\rangle$ and $L_{om} = \langle L, prob, sub\_cost, obs\_cost\rangle$.

Figure 2 details the part of the simplified aircraft system involved in the Operational Mode called *BITE*.

In order to ascertain whether the observed values of the known variables correspond to a correct behaviour, each OM must include a set of *Observations of Correct Behaviours*, described as follows.

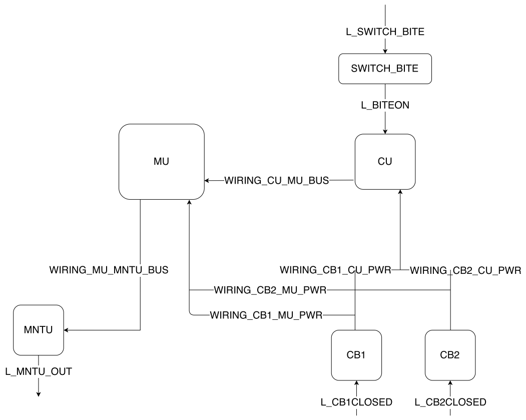


FIGURE 2. Simplified aircraft system in BITE operational mode.

1) OBSERVATIONS OF CORRECT BEHAVIOURS

Detecting correct or incorrect behaviour of a system can be carried out by observing the known variables. To this end, it is necessary to have previously determined the tuples of observations that represent correct behaviours. Each correct behaviour is identified by a tuple of values assigned to the known variables of the links in a system that does not present incorrect behaviour. When the known variables are not equal to any of the tuples that define correct behaviours, it is assumed that the observation corresponds to an incorrect behaviour.

Table 3 describes the *observations of correct behaviours* of the known variables for the BITE Operational Mode of the example. Since the domain of the variables for the example is *Boolean*, they can take value 0 or 1.

D. PRIORITY FUNCTION (PF)

As mentioned before, the selection of a component as being the element that is probably responsible for a malfunction derives from: a structural analysis of the relation of

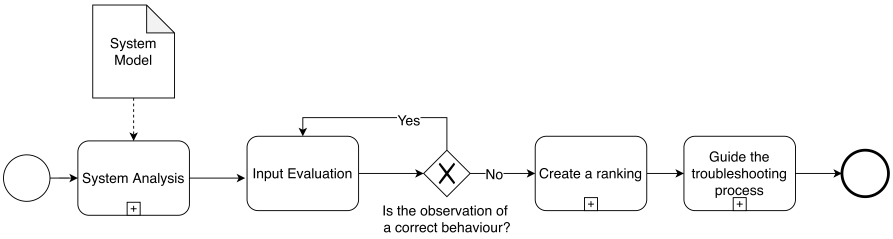

System Model
System Analysis
Input Evaluation
Yes
No
Is the observation of a correct behaviour?
Create a ranking
Guide the troubleshooting process

FIGURE 3. Steps of the methodology.

TABLE 3. Correct valuation for known variables in BITE operational mode.

KNOWN VARIABLES	CORRECT BEHAVIOUR
WIRING_MU_MNTU_BUS	1
L_CB1_CLOSED	1
L_CB2_CLOSED	1
L_SWITCH_BITE	1
L_PTT_PULSED	1
L_MNTU_OUT	1

the components (described by using the component and link elements), the probability of failure of the components (*probability*), and the costs of substitution and observation (*cost*). Once the involved components are determined by structural analysis, and in order to sort them into a ranking, a priority function is applied to each component by combining its *probability* and *costs* in a multi-objective function. The priority value obtained is the employed for sorting the components into order. One example of a priority function could be:

$$\sin(\frac{\pi}{2} \cdot E_{probability}) \cdot [0.5 \cdot E_{subsCost} + 0.5 \cdot E_{obsCost}]$$

where $E_{probability}$ is the probability of fault in the elements (both links and components) for the activated OM; $E_{subsCost}$ is the substitution cost of the element for the selected OM; and $E_{obsCost}$ is the observation cost. Furthermore, links have an associated observability cost (obs_cost), while, for components, $E_{obsCost}$ is the highest reading cost of all output links of the component. For this function, the domain of the results is in the range between 0 and 100. Therefore, for an element, a result close to 100 indicates that it can potentially be responsible for the failure, while if the result is close to 0, that it means that the element is almost definitely not responsible for the failure. Accordingly, if two elements obtain similar values from the priority function, it indicates that they both share roughly the same chance of being responsible for the failure. By applying the priority function to each element, a sorted ranking of elements is obtained, from highest to lowest probability. The proposed methodology explains how this interpretation can guide the engineer (Section III).

III. TROUBLESHOOTING PROCESS METHODOLOGY

Since the complexity of the systems can generate a very long list of possible root causes, it would be highly unprofitable to verify each and every possibility. This is the scenario where the operators must decide what is the best option, frequently following predefined guides designed by the engineers. To facilitate this process, our proposal customises the steps of the troubleshooting process according to the observations. This is why the proposed troubleshooting guide is needed to helping the decision making regarding on the most appropriate components to replace/analyse or the variables to read in order to obtain a more accurate and optimal diagnosis.

The methodology has four steps (as depicted in Figure 3): (1) system analysis; (2) system monitoring via input values; (3) creation of a ranking of elements according to the priority function; and (4) the execution of the guiding process to find the root of the fault, with a lower cost based on the highest probability.

The process starts a system analysis in accordance with the formalisation in Section II, which provides a structural analysis for the relations between the elements and defines how a malfunction affects the observations, as explained in Subsection III-A. In accordance with the observations at runtime, the system may be diagnosed, and a ranking is created that considers the probabilities of failure and their costs of substitution and observation by using the priority function (as explained in Subsection III-B). The troubleshooting guiding process enables the component responsible for the malfunction to be found while minimising the costs, as detailed in Subsection III-C.

In the following subsections, these steps are laid out in detail, and the example introduced in Section II is employed as an illustration.

A. SYSTEM ANALYSIS

In order to ascertain those elements which are potentially responsible for an observed malfunction (possible root cause), a structural analysis of the dependencies between the elements is necessary. The analysis process starts with the model of the system as input, which is decomposed (1) and analysed according to the dependencies between the components and links (2).

1) MODEL DECOMPOSITION

According to the formalisation, a component can have multiple inputs and outputs and therefore our proposal

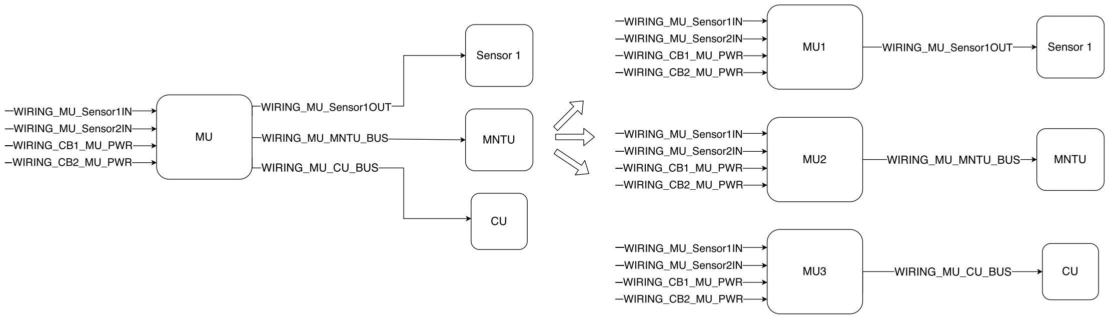


FIGURE 4. **MIMO component MU transformed into three MISO components.**

decomposes the system when necessary. Specifically, decomposition focuses on component elements with multiple outputs, the so-called MIMO (Multiple Input-Multiple Output) components. These MIMO components make exoneration tasks more complex, since a malfunction in a MIMO component can produce correct and incorrect outputs at the same time. Generally, in classic diagnosis, the correct observation of the output of a component exonerates it from the responsibility of a malfunction. However, if there are MIMO components in the model, this exemption becomes more complex during the troubleshooting process.

In order to apply exoneration according to a correct observation, we need to transform MIMO components into MISO (Multiple Input-Simple Output) components [13]. The transformation of a MIMO component E creates as many new MISO components as outputs of E, all of which contain the same input link elements as E. For the example, in Figure 1, there are 5 MIMO components: $\{DC, MU, CB1, CB2, CU\}$. In Figure 4, the component MU is transformed into three new components $MU1$, $MU2$, and $MU3$, one for each output of the MU original component.

With this decomposition, a MIMO component can only be exonerated if every single one of its MISO components are exonerated. Thus, situations like the following could occur: the output WIRING_MU_Sensor1IN is exonerated due to the correct reading of a variable. This leads to the MISO component $MU1$ being exonerated, but the entire MU component cannot be exonerated, since its other two outputs could still be incorrect. Subsequently if the MU component has to be analysed, then the analysis begins by reading its outputs that yet to be exonerated yet (that is, the exoneration of its remaining MISOs are sought).

2) DEPENDENCY ANALYSIS

There are many techniques that obtain the structural dependencies of a system [14]. This proposal focuses on the creation of a fault signature matrix [2], [15] that represents the set of variables that would be affected when each specific element fails. In the field of diagnosis, the use of fault signature matrices facilitates the isolation of components depending on the observations. For example, if the observation of a variable (output of an element E) denotes that E works correctly, then this implies that all the elements whose outputs are inputs of E also work correctly. These can therefore be removed from the list of possible root causes, thereby propagating the exoneration to related components.

After the model decomposition phase, the fault signature matrix is created to represent the relation between the element link variables, which can be observed in order to ascertain whether if the process is working correctly, and the elements of the system that can fail. Table 4 shows a part of the signature matrix for the system in BITE Operational Mode presented in Figure 2, to make the proposal understandable. The signature matrix includes only the MISO components involved in the specific operating model under analysis. Thus, for instance, components $MU1$ and $MU3$ do not appear, because they do not operate in this mode. Likewise, link elements that do not have an element as their origin never appear in the matrix, since if they were included, their column would always be empty, which means that under no circumstances reading them can help to exonerate any element.

In order to obtain the fault signature matrix [16], the structural relations in the system model are analysed using the information regarding its inputs and outputs. To enable the representation of which variables are affected when an element fails, a signature matrix has one row per element (component or link), and one column per known or readable variable of the system. When an X is located in a cell which is in the row of element E and the column of variable v, this means that if element E fails, then the value of v is affected by this fault. It is equivalent to the existence (in the model) of a path between the output of element E and the link variable v. However, the reverse implication is not possible, since an element E can work correctly while v is incorrect, due to another element implication. For the variable point of view, if v is incorrect, at least one of the elements with an X in its column must be working incorrectly. The signature matrix also includes the variables associated to the input value of the link element, such as L_BITEON_(I). As described above, a link element is an element with only one input and

TABLE 4. Signature matrix for BITE OM system.

	LINK ELEMENT VARIABLES								
ELEMENTS	L_BITEON	L_BITEON_(I)	WIRING_CB1 _CU_PWR	WIRING_CB2 _CU_PWR	WIRING_CU _MU_BUS	WIRING_CB1 _CU_PWR	WIRING_CB2 _MU_PWR	WIRING_MU _MNTU_BUS	L_MNTU_OUT
SWITCH_BITE	X	X						X	X
L_BITEON	X							X	X
CU					X			X	X
WIRING_CB1 _CU_PWR			X					X	X
WIRING_CB2 _CU_PWR				X				X	X
CB11			X		X			X	X
CB12						X		X	X
CB21				X	X			X	X
CB22							X	X	X
WIRING_MU _CU_BUS					X			X	X
MU2								X	X
WIRING_CB1 _MU_PWR						X		X	X
WIRING_CB2 _MU_PWR							X	X	X
WIRING_MU _MNTU_BUS								X	X
MNTU									X
L_MNTU_OUT									X

TABLE 5. Ranking of elements for an incorrect input tuple in BITE OM.

KNOWN VARIABLES	INCORRECT BEHAVIOUR
WIRING_MU_MNTU_BUS	**0**
L_CB1_CLOSED	1
L_CB2_CLOSED	1
L_SWITCH_BITE	1
L_PTT_PULSED	1
L_MNTU_OUT	**0**

(a) Tuple of incorrect values for known variables

ELEMENT	PRIORITY FUNCTION VALUE
WIRING_MU_MNTU_BUS	58.337
CB1	57.896
CB2	57.896
WIRING_CB1_MU_PWR	47.895
WIRING_CB2_MU_PWR	47.895
MU	8.916
MNTU	8.603

(b) Ranking of possible root cause elements

one output. The column of an input link element is equal to the column of the element link variables without an X in the element link. These variables are not included in the model since they are derived from the link element, and no such extra information is necessary.

B. RANKING CREATION

During the assembly and testing processes, the system functionalities are continuously monitored to verify the correct behaviour, to make a diagnosis if needed. If certain known variables present incorrect values, this means that the read input tuple does not match the operational mode. It therefore becomes necessary to formulate a hypothesis regarding the malfunction, and if the diagnosis activity begins. Depending on the specific known variables that are correct or incorrect, it is possible to isolate the potential elements involved in troubleshooting, since probably not all the elements that belong to the OM can be responsible for the detected malfunction. This will depend on the structural analysis (used to create the signature matrix) and according to the specific incorrect known variables. For example, Table 5(a) describes a tuple of known variable values that does not match with the correct behaviour in the OM, and therefore a diagnosis process should be executed. When an incorrect tuple is detected, the elements involved in the OM must be ordered, such as is shown in Table 5(b). The troubleshooting process starts analysing this list and the signature matrix.

It is important to note that MISO components never appear in the ranking, only MIMOs appear. This is because when a component has to be replaced, it must be completely replaced, since its division into MISOs is not physical, but logical. In this way, the MISOs intervene in the subsequent guiding process only to facilitate exemptions, while only the MIMO components appear in the ranking.

Furthermore, it is important to highlight that, as mentioned above, if two or more elements appear in the ranking with the same value for the priority function, then this indicates that they have the same value obtained using the priority function. In these circumstances, the operator can decide which is to be analysed first.

C. GUIDING THE TROUBLESHOOTING PROCESS

In classic approaches, and making use of a ranking of potential elements to be replaced, the troubleshooting starts by replacing the first element in that ranking (*WIRING_MU_MNTU_BUS* in the previous example), since it has the greatest value according to the priority function. Therefore, if the problem is solved, the diagnosis process is complete. In contrast, if the malfunction persists, the diagnosis process follows the order established in the ranking for the replacement of elements one by one until the origin of the malfunction is found.

However, due to the potential complexity of the models, the potential root causes can be very numerous and very

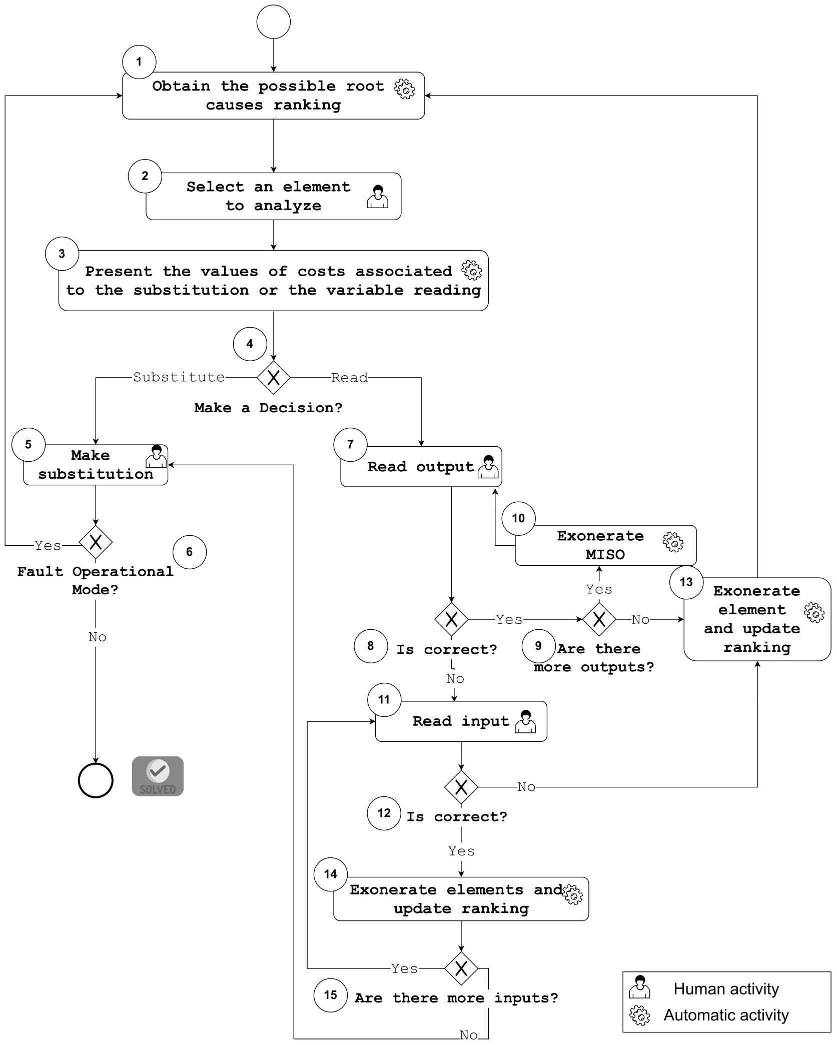

1 Obtain the possible root causes ranking
2 Select an element to analyze
3 Present the values of costs associated to the substitution or the variable reading
4 Make a Decision?
Substitute
Read
5 Make substitution
6 Fault Operational Mode?
Yes
No
SOLVED
7 Read output
8 Is correct?
Yes
No
9 Are there more outputs?
Yes
No
10 Exonerate MISO
11 Read input
12 Is correct?
No
Yes
13 Exonerate element and update ranking
14 Exonerate elements and update ranking
15 Are there more inputs?
Yes
No
Human activity
Automatic activity

FIGURE 5. Stages of the guiding algorithm.

expensive to fix or replace, therefore, it may not be feasible to replace or repair the elements without being 100% certain that they are really causing the problem. That is why in the troubleshooting process, our proposal includes, the readings of readable variables to isolate the problem in an optimal way, and to prevent unnecessarily replacing correct elements whenever possible.

During the process, the steps presented schematically in Figure 5 are proposed as a guide, and they serve as support to explain, in greater detail, the different stages that make up the guiding process until the component responsible for the malfunction is detected.

According to the ranking **(Step 1)**, such as that shown in Table 5 (b), the operator must decide which element to analyse **(Step 2)**. The algorithm suggests analysing the elements in the order that they appear in the ranking; however, if the operator decides to analyse another element instead of that which the ranking order proposes, then the ignored element is relocated at the end of the list for future consideration. The exoneration or blame of the components can be determined in a direct (by means of their replacement) or an indirect way (reading readable variables) **(Step 3)**.

The information provided for each element to make the decision is composed of three types of costs:

- **Element substitution cost:** the cost of replacing the component/link element.
- **Cost of ascertaining whether the element is working correctly:** if the element is a component, then it might have one or more outputs, which are readable or known variables. In the calculation of this cost, we must differentiate between two scenarios: (1) the outputs of the component are correct, or (2) the component outputs are incorrect and all its inputs are correct. Consequently, this cost is equal to the sum of observation costs of the outputs plus the sum of the observation costs of the inputs. Similarly, with link elements, if the variable associated with the link is correct, then the cost of ascertaining that it is correct is equal to the cost of reading its variable. However, if the variable is incorrect, then the cost of knowing that the link element is correct is equal to the cost of reading its variable with an incorrect result plus the cost of reading the input, also with an incorrect result.
- **Cost of ascertaining whether the element is not working correctly:** This is the cost of finding out that its outputs are incorrect, plus the cost of reading all its inputs to verify that they are correct.

Based on this information, the operator has to decide (**Step 4**) between substituting the element or, if it is possible, analysing the element indirectly by reading variables. If the operator decides to substitute (**Step 5**), the element must be replaced and the persistence of the fault should be checked (**Step 6**). If the malfunction disappears, then the troubleshooting process ends since the responsible for the failure has been found. On the other hand, if the fault is still present, then the information regarding the performed substitution is recorded and the troubleshooting process continues once the ranking is updated accordingly.

For the example, it is decided to analyse the element by reading variables; the process starts reading the output of the element **(Step 7)** to exonerate it and the other elements related to that variable in the signature matrix (**Step 13**). If the variable is incorrect (**Step 8**) the inputs of the element must be read (**Step 11**) to ascertain whether they are incorrect or not (**Step 12**). The casuistry is as follows:

- If at least one input variable is incorrect, then the component under study can be exonerated since its incorrect output has been caused by another faulty component (**Step 13**).
- If an input is correct (**Step 15**), then the components related to it in the signature matrix can be exonerated (i.e., those that have an X in the column of the input variable).
- If every input is correct (**Step 16**), then it is implied that the component produces an incorrect output from correct inputs, and therefore it is possibly the component the possible responsible for the malfunction.
- If an output is correct (**Step 8**) and the element is MIMO, then only the MISO element can be exonerated **(Step 10)**, unless it is the last MISO component (from that MIMO element) that remains to be exempted, which would then exonerate the entire MIMO component, as well as those related to the output variable in the signature matrix **(Step 13)**.

However, it is important to note that although all readings are saved during execution, if an element is replaced, then these read variables are reset, since every time an element changes in the system, the system also changes and, therefore, previous readings are no longer reliable.

IV. GUIDING PROCESS FOR THE EXAMPLE

For a better understanding of the previous algorithm, an example of the trace of the troubleshooting is detailed below, where the element responsible for the malfunction is MU, although this is unknown before the troubleshooting process starts. Figure 6 shows the BITE OM diagram, which includes costs of the substitution of components and costs of reading variables, and has been labelled with the phases described in the algorithm to facilitate its understanding.

Details of the development of the guiding process for a specific scenario are now given, and include decisions, readings made, substitutions, etc.

1) Starting from the incorrect input tuple shown in Table 5(a), the ranking shown in Table 5(b) is obtained (**Step 1**). *WIRING_MU_MNTU_BUS* is the first element, a link, and the first candidate to be analysed according to the priority function results (**Step 2**). The system shows the three costs associated with the analysis of that element **(Step 3)**:
   - **Substitution cost**: 1,000.
   - **Cost of ascertaining whether *WIRING_MU_MNTU_BUS* is correct**:
     - **Best-case scenario**: 100. The cost of reading the output variable is correct, implies that the element is correct.
     - **Worst-case scenario**: 200. The cost of reading the output *WIRING_MU_MNTU_BUS* and input *WIRING_MU_MNTU_BUS_(I)* variables, to ascertain whether the component is working correctly (input and output incorrect) or incorrectly (input correct and output incorrect).
   - **Cost of ascertaining whether *WIRING_MU_MNTU_BUS* is incorrect**: 200. This cost is equal to ascertaining that it is correct in the worst-case scenario.

   Based on these three costs, the operator must decide whether to choose whether the replacement of the element or to perform the readings of a variable (**Step 4**). For the example, the decision is to read (**Step 7**), since it costs 200 instead of the 1,000 of the substitution. The algorithm then asks if the read variable is correct (**Step 8**). In this case, it is not correct (**Step 8**) and its input must be read (**Step 11**). The input of the link (*WIRING_MU_MNTU_BUS_(I)*) is incorrect (**Step 12**), and therefore, the link element is correct

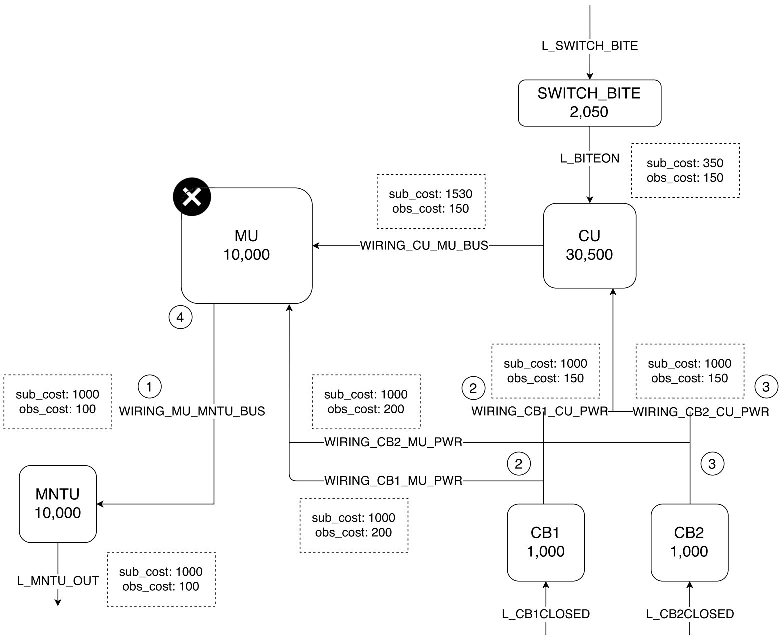


FIGURE 6. Troubleshooting example trace diagram.

TABLE 6. Ranking after first analysis.

ELEMENT	PRIORITY FUNCTION VALUE
CB1	57.896
CB2	57.896
WIRING_CB1_MU_PWR	47.895
WIRING_CB2_MU_PWR	47.895
MU	8.916
MNTU	8.603

and can be exonerated (**Step 13**). The ranking is then updated and the process continues (**Step 1**). Table 6 shows the new resulting ranking.

2) Following on with the example, the next element to be analysed is *CB1* (**Step 2**). The system shows the three costs associated to its analysis:
   - **Substitution cost**: 1,000.
   - **Cost of ascertaining whether *CB1* is correct**:
     - **Best-case scenario**: 350. This component is MIMO and is divided into two MISO components,(*CB11* and *CB12*). The cost of reading each output correctly (*WIRING_CB1_CU_PWR_(I)* and *WIRING_CB1_MU_PWR_(I)*) is 150 and 200 respectively, and the total cost their sum.
     - **Worst-case scenario**: 350. The worst-case scenario in ascertaining whether this component is correct is equal to the sum of reading its outputs, being incorrect, plus the reading of its input variable (*L_CB1_CLOSED*), being incorrect. Since *L_CB1_CLOSED* is a known variable, then its observation cost is 0 and the total cost is equal to 350.
   - **Cost of ascertaining whether *CB1* is incorrect**: 350. This cost is equal to the cost of ascertaining whether it is correct in the worst-case scenario.

   This time, when these costs are shown to the operator (**Step 3**), the option chosen is that of analysing by reading (**Step 4**), and the analysis begins by reading the outputs of element *CB1* (**Step 7**). These turn out to be correct (**Step 8**) and, therefore, the entire MIMO component can be exonerated, because both its MISO components have been exonerated (**Steps 10 and 13**). In addition, as result of the correct readings, two actions are carried out: (1) Elements are sought whose outputs are related to the correct variable (there is an X in the column of the variables *WIRING_CB1_CU_PWR_(I)* and *WIRING_CB1_MU_PWR_(I)*, and in the row of the potential components to be exonerated in the fault signature matrix), which, in this case, there are none; and (2) the correct reading of these variables is stored in the system so that it is taken into account in case they are needed in the future. However, the malfunctioning element has yet to be identified, and hence it is necessary to update the ranking and continue the search (**Step 1**). Table 7 shows the new ranking.

3) Now, the first element in the ranking is *CB2* (**Step 2**). The system displays the three related costs (**Step 3**):

TABLE 7. Ranking after the second analysis.

ELEMENT	PRIORITY FUNCTION VALUE
CB2	57.896
WIRING_CB1_MU_PWR	47.895
WIRING_CB2_MU_PWR	47.895
MU	8.916
MNTU	8.603

TABLE 8. Updated ranking after the third analysis.

ELEMENT	PRIORITY FUNCTION VALUE
WIRING_CB1_MU_PWR	47.895
WIRING_CB2_MU_PWR	47.895
MU	8.916
MNTU	8.603

- **Substitution cost**: 1,000.
- **Cost of ascertaining whether *CB2* is correct**:
  - **Best-case scenario**: 350. This component is MIMO and is divided into two MISO components (*CB21* and *CB22*). The cost of reading each output correctly(*WIRING_CB2_CU_PWR_(I)* and *WIRING_CB2_MU_PWR_(I)*), is 150 and 200 respectively.
  - **Worst-case scenario**: 350. The sum of reading its outputs (350), whereby at least one is incorrect, plus the reading of its input variable (*L_CB2_CLOSED*), which is also incorrect. Since *L_CB2_CLOSED* is a known variable, its observation cost is 0 and the total cost is equal to 350.
- **Cost of ascertaining whether *CB2* is incorrect**: 350. This cost is the same as the cost of the worst-case scenario.

By using this information, the operator chooses to analyse *CB2* by readings (**Step 4**), and the study begins by reading the its outputs (**Step 7**), which turn out to be correct (**Step 8**) and, therefore, the entire MIMO component can be exonerated, because both its MISO components have been exonerated (**Steps 10 and 13**). The two tasks that are executed after the correct readings are carried out in the same way as in the previous component. Finally, this is not the faulty component, and therefore it becomes necessary to update the ranking and continue the search (**Step 1**). Table 8 shows the new ranking.

4) *WIRING_CB1_MU_PWR* is the next element of the ranking to be analysed (**Step 2**). The system provides the following costs (**Step 3**):
   - **Substitution cost**: 1,000.
   - **Cost of ascertaining whether *WIRING_CB1_MU_PWR* is correct**:
     - **Best-case scenario**: 200. The cost of reading the output variable of the link that is correct.
     - **Worst-case scenario**: 200. The cost of reading the variable of the link that is incorrect plus the cost of reading its input variable (*WIRING_CB1_MU_PWR_(I)*), also incorrect. However, the reading of the variable *WIRING_CB1_MU_PWR_(I)*has already been carried out with a correct result and therefore it does not need to be read again.
   - **Cost of ascertaining whether *WIRING_CB1_MU_PWR* is incorrect**: 200. This cost is equal to the cost of reading the variable of the link, and ascertaining that it is incorrect, plus the cost of reading the variable that is the input of the link *WIRING_CB1_MU_PWR_(I)*, that is correct, making a total of 400. However, the reading of the variable *WIRING_CB1_MU_PWR_(I)* has already been carried out with a correct result, and therefore, it does not need to be read again, and no extra cost is incurred.

   Based on the information given, the operator must decide whether to choose to replace the element or to perform the readings of a variable (**Step 4**). The decision is to read (**Step 7**). Following on, the algorithm asks whether the read variable is correct (**Step 8**). In this case, it is correct and since it is a link, there are no more outputs (**Step 9**), and therefore the link element must be exonerated **(Step 13)**. In the new scenario, the fault persists, and therefore the ranking is updated and the process continues (**Step 1**).

TABLE 9. Updated ranking after the fifth analysis.

ELEMENT	PRIORITY FUNCTION VALUE
MU	8.916
MNTU	8.603

5) *WIRING_CB2_MU_PWR* is the next element of the ranking to be analysed (**Step 2**). Following the same reasoning as in the previous element, the system shows the following costs (**Step 3**):
   - **Substitution cost**: 1,000.
   - **Cost of ascertaining whether *WIRING_CB2_MU_PWR* is correct**:
     - **Best-case scenario**: 200.
     - **Worst-case scenario**: 200.
   - **Cost of ascertaining whether *WIRING_CB2_MU_PWR* is incorrect**: 200.

   The operator decides to analyse the link elements by readings again (**Step 7**). The reading is correct (**Step 8**) and since it is a link, there are no more outputs (**Step 9**), therefore, and the link element must also be exonerated (**Step 13**). The ranking is updated as shown in Table 9.

6) *MU* is the next item on the ranking to be tested (**Step 2**). The system provides the following costs (**Step 3**):
   - **Substitution cost**: 10,000.
   - **Cost of ascertaining whether *MU* is correct**:
     - **Best-case scenario**: 0. This component is MIMO, but in this mode, only one of its MISOs

is operative: *M2*, whose output is the variable *WIRING_MU_MNTU_BUS_(I)*. The reading cost of this variable is 100. Nevertheless it has been read previously, making the total cost for the MIMO element equal to 0.

  - **Worst-case scenario**: 150. This cost is equal to the cost of reading its output incorrectly, which is 0, plus the sum of the observation costs of its three inputs (*WIRING_CB1_MU_PWR, WIRING_CB2_MU_PWR, WIRING_CU_MU_BUS*), which must all be correct. Taking into account that the first two have already been exonerated, this cost is equal to the observation cost of *WIRING_CU_MU_BUS*.

- **Cost of ascertaining whether *MU* is incorrect**: 150. The cost is the same as for the worst-case finding that it is correct.

This time, again, the operator decides to analyse by reading variables (**Step 7**). As before, the process starts reading the output variables of the component, which, as previously analysed, presents an incorrect result **(Step 8)**. This leads to reading the inputs **(Step 11)**. This component has three inputs, two of which have already been read. When *WIRING_CU_MU_BUS* is verified, it is correct too, implying that the component is receiving correct information and producing erroneous information. Therefore, this component would be a potential cause of the failure, and the algorithm indicates to the operator that it should be replaced (**Step 5**). After the replacement, the fault disappears and the search process ends, since *MU* is responsible for the failure.

The convenience of applying our proposal instead of the classical proposals, where no partial readings are included, is laid out in the following section.

V. EVALUATION OF THE PROPOSAL

The effectiveness of our algorithm depends on the location in the ranking of the element responsible for the malfunction for each test. If the element responsible is the first one, then our algorithm incurs time and cost reading various variables. Otherwise, if the element responsible is deep within the list, our algorithm reduces the diagnosis cost drastically. To measure this effectiveness while taking into account the probability that an element appears in the different positions of the ranking, the troubleshooting guiding algorithm has been validated using a simplified aircraft system model of a real environment in Airbus Defence and Space factories. In order to verify the validity of the proposal, Subsection V-A shows how the algorithm has been tested in three different Operational Modes by simulating the possible element responsible for the malfunction for each mode and examining how the guidance of our algorithm could affect the decisions. This verification has been carried out to compare the benefits and disadvantages of our methodology versus the traditional methodology. Subsection V-B lists the conclusions drawn in the validation, the advantages of our approach, and those cases where its use can introduce major benefits.

A. VALIDATION OF THE PROPOSAL

In this section, our approach is tested on three different OMs from the same simplified aircraft system. This implies that, in each OM, certain elements are involved, while others are not, and therefore the lists of possible root causes change with each test. For illustrative reasons, and to be able to go into greater detail, give a full description of the tests for one OM, although the obtained results of the other two are also included in the paper. The details of the results of the other two tests are shown on the web.[1]

For every test, results are presented in accordance with two approaches: (1) classic, replacing elements in the order of the cost-probability relationship given in the ranking; and (2) our proposal, based on the reading of variables.

The tests are carried out, by using a defined ranking to determine the costs of discovering of the faulty element. These costs depend on the type of approach used and the location in the ranking of the element responsible. To gather every cost, both approaches and the positions of the element are studied for a later analysis. Table 10 contains the information regarding the results achieved for the OM BITE2. First, the ELEMENT column represents the ranking of possible root causes identified during the troubleshooting process, sorted in the order from highest to lowest according to the priority function. The column CLASSIC APPROACH contains the results of costs incurred in ascertaining whether an element is responsible for the failure or not, following the classic methodology of substituting elements according to the ranking order. In the classic approach, the cost of substituting the i-th component includes the cost of substituting every (i-1)-th previous element that turned out to be correct (the best-case or the worst-case scenario, depending on the case) and the element involved in each tuple. Regarding our approach, the set of columns PROPOSED APPROACH contains the same information with the exception that, in our proposal, the costs differ depending on whether a scanned element is working properly and its outputs are correct (lowest cost to ascertain whether it is correct) or reading every inputs and outputs, that must be performed when the output is incorrect and it is necessary to ascertain if the inputs are correct. When an incorrect element is detected, it is necessary to include the replacing cost to the cost of the column to determine the culprit. Moreover, both for correct and incorrect elements, the cost of analysing the element of the i-th row includes the cost of ascertaining that the (i-1)-th previous elements are also correct. Finally, the column SAVINGS contains the percentages of cost reduction of our proposal compared to the classic proposal in the analysis of each element (the percentage includes the substitution cost of each element plus the cost

[1] `http://www.idea.us.es/ts2020/`.

TABLE 10. Comparison of costs and steps of traditional and proposed solutions in BITE2 mode.

ELEMENTS	SUBSTITUTION COST	CLASSIC APPROACH	PROPOSED APPROACH		SAVINGS (%)
		Cost to know that is correct or not	Lowest Cost to know if it's correct	Greatest Cost to know if it's correct or culprit	
CB2	1,000	1,000	350	350	-35%
CB1	1,000	2,000	700	700	15%
WIRING_CB2_CU_PWR	1,000	3,000	850	850	38.33%
WIRING_CB1_CU_PWR	1,000	4,000	1,000	1,000	50%
WIRING_CU_MU_BUS	1,530	5,530	1,150	1,150	51.53%
CU	30,500	36,030	1,300	1,750	10.49%
MU	10,000	46,030	1,400	1,950	74.03%

TABLE 11. Costs per probability of failure and savings in BITE2 mode.

ELEMENTS	PROBABILITY	PROBABILITY in 100 tests	CLASSIC APPR. COST	PROPOSED APPR. COST
CB2	30	19.23	19,230.76	25,960.50
CB1	30	19.23	38,461.53	32,691
WIRING_CB2_CU_PWR	30	19.23	57,692.30	35,575.50
WIRING_CB1_CU_PWR	30	19.23	76,923.07	38,460
WIRING_CU_MU_BUS	20	12.85	70,897.43	34,438
CU	8	5.12	184,769.23	165,120
MU	8	5.12	236,051.28	61,184

to determine that this element is responsible, represented in the column with the highest costs in the proposed approach).

In the first test, we found a small ranking, in which only a few elements of the system intervene, as illustrated in Table 10. It can be observed that the costs are drastically reduced when the elements analysed are not responsible for the failure. When the defective element is at the top of the ranking, our proposal entails higher costs than the classic approach. However, the cost decreases significantly when the position of the defective item is lower in the ranking, and costs can be halved or reduced even more in certain cases.

If we focus on the column of Table 10 that shows the cost of determining whether the element is incorrect, the cost when each component fails can be ascertained. However, the analysis must include the number of times that the each element is actually the responsible for the malfunction. To evaluate our proposal, the probability information associated to each component is used. In the example, the probability is assessed in the range [0..100], where 0 represents the least probability of being responsible for the malfunction. In order to gain a more general idea of the advantages of applying our proposal, 100 tests are simulated, which take into account the different probabilities associated to each element. Table 11 includes the probability of each element where the summatory of the probability is an impossible 156 in this case. In order to obtain the column (Probability in 100 tests), a proportional adjustment is carried out. Based on the column probability in 100 test, we multiplied this value by the cost to solve the problem for each element, comparing the two approaches (columns Classic appr. cost and Proposed appr. cost). For the 100 elements, the total cost for the classic approach is $\sum_{i=1}^{Elements} probabilityIn100Tests_i * ClassicAppr.Cost_i = 684,025.64$, and $\sum_{i=1}^{Elements} probabilityIn100Tests_i *$ $ProposedAppr.Cost_i = 393,717.94$, thereby saving $290,307.69$ €, which implies saving of 42.44% in the 100 tests.

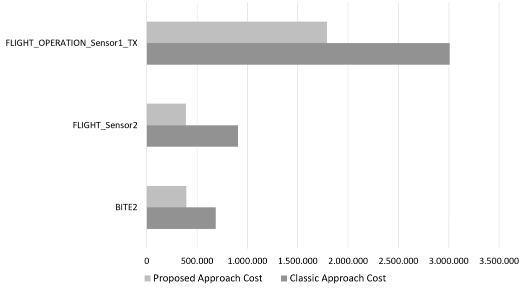


FIGURE 7. Total costs obtained for 100 tests with the two approaches.

B. CONCLUSIONS OF THE VALIDATION

Based on the observations collected in the previous subsection, four general behaviours can be concluded:

1) For elements in the first positions of the ranking, there are no major differences between the traditional solutions and our algorithm. In fact, costs remain the same or even increase using our approach, as well as the number of steps needed to reach the solution.
2) In the average cases, that is, those in which the elements are in the central positions of the ranking, the use of our proposal compared to the traditional approach enables costs to be reduced, and attains a greater reduction in the lower ranking positions.
3) For the last elements of the ranking, all costs are reduced by at least 50%. Those cases in which certain elements are extraordinarily expensive, should be borne in mind, since the costs of our proposal are also lower than the costs of the traditional methodology.
4) By carrying out a statistical analysis of the results, we have reached the conclusion that, assuming that each failure was repeated 100 times, and taking into

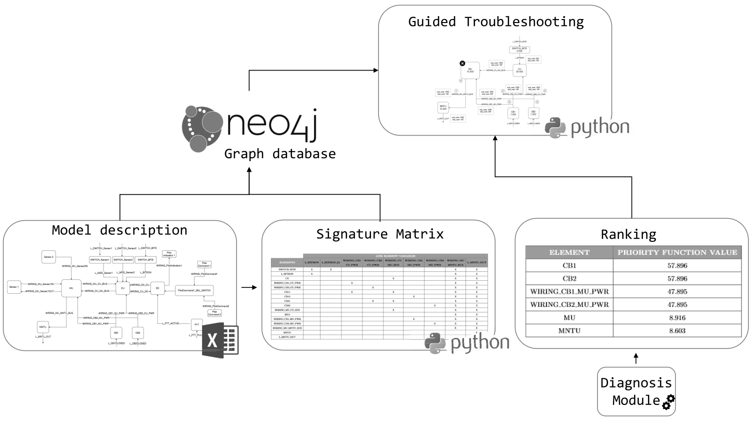


FIGURE 8. System architecture for guided troubleshooting.

account the probability of failure of each element, our proposal produces significant savings: 42.44%, 57.26%, and 40.54%. Figure 7 shows the large differences between the total costs for the approaches for each of the three tests.

VI. IMPLEMENTATION DETAILS

Since the solution is oriented towards guiding troubleshooting during the assembly processes, it is crucial that the implementation can be executed not only on computers, but also on tablets or other mobile devices. For this reason, the set of combined technologies is oriented towards the optimisation of time and resources, since the main aim is to achieve an efficient and fast response system, with a suitable performance and minimum resource consumption.

As shown in Figure 8, based on the description of the system model and the structural relationships between elements, the troubleshooting system needs a mechanism to describe the dependencies for the later isolation process. Hence, we propose the use of labelled graphs for modelling the problem representation, whereby it is possible to use a graph database to store and query these graphs, as well as storing the signature matrix that is used during the execution of the guided algorithm. On the other hand, the diagnosis module provides the ranking of possible root causes of failure, and, by using these possible causes, our system allows the operator to execute the guiding process to ascertain the real element responsible for the malfunction.

In order to achieve the aforementioned objectives, our graph database has been designed as follows:

- **Nodes**. We have used four different types of nodes: **elements**, **link variables**, **operational modes**, and **signature matrices**. For each node, the graph database only stores its name.
- **Edges**. Three types of edges are included:
  - *Operational-mode structural relationships*: These represent the structural relations that are established between the elements and variables of the system in the different operational modes. Thus, if, in a particular operational mode, a set of elements and variables must necessarily be connected and either active or inactive, then the edges linking them will appear in the graph. To represent this relationship, each edge that links an element and a variable has the name of an operational mode and contains additional information in the way of attributes, such as: (1) the type of relationship between the element and the variable (input, output, bidirectional); and (2) a value that represents the state in which the interface of the element must necessarily be (0: inactive; 1: active).
  - *Operational-mode costs and probabilities*: Each element has an edge to each operational mode. These edges specify their substitution cost and failure probability as an attribute. Similarly, each variable also has an edge to each operational mode, which contains the cost of observation of the variable as an attribute.
  - *Correspondence between the operational mode and its signature matrix*: These are edges that link the operational-mode nodes with the signature-matrix nodes. In this respect, it is possible to determine which operational mode is associated with which signature-matrix failure.

Figure 9 shows an example of a graph database of the BITE mode.

Our proposal is developed by using Python to create the logic of the algorithm described in Section III and by employing **Neo4j** as a graph database. **Neo4j** is a No-SQL graph-oriented database that is employed to store information

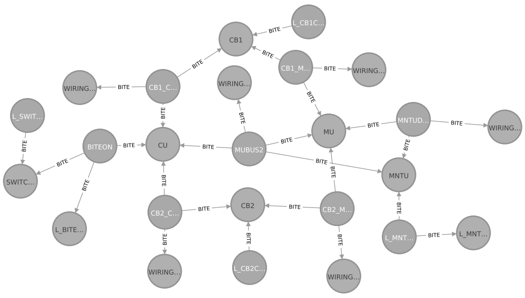


FIGURE 9. Example of a BITE mode graph.

related to the data model. Thanks to the structural relationships established between the elements of the studied subsystem, it can be modelled using a graph topology. For this type of information structuring, this type of database gives us significant advantages over an SQL solution, since the queries made on the data model imply adjacency relationships, which are as important as the data itself. In addition, due to the features of the algorithm, most of the queries require the navigation through different levels of depth, and this is always more efficient if graphs are used instead of SQL "*join*" operations. Moreover, it should be taken into account that for this small simplified problem, our database supports more than 100 nodes and 1, 600 edges.

On the other hand, the query language used by this database favours simplicity, which is an important fact to take into account, since the lighter the logic used by the system, the greater the chances of using it on different platforms and devices. Details of the implementation, more tests, and an example of the applicability of our proposal is available in the web.[2]

VII. RELATED WORK

In previous work, automated troubleshooting has brought significant benefits for quality assurance in manufacturing or network systems [17]. On these systems, failures can occur and various attempts could be made to repair them in the best way possible. Furthermore, on the aircraft manufacturing or similar complex systems, faults could propagate or affect numerous other systems, which leads to a great quantity of work to solve the fault. Our work has been motivated by the limitation of available information to perform the best fault troubleshooting process during the manufacturing. In these cases, many hypotheses have be to considered and the troubleshooting process could require a high number of steps to solve a determined failure. The selection of previous studies that are more closely related to our paper includes topics such as sequential fault diagnosis [18], interactive troubleshooting and troubleshooting using Artificial Intelligence techniques.

The MyAID application in [19] provide workers with interactive troubleshooting of industrial machines. It relies on hypermedia information systems, and shows step-by-step instructions using multimedia material. However, this study fails to specify how the application minimises the time and cost in the search for causes of failure.

Another set of studies related to fault diagnosis and troubleshooting for aircraft systems use Artificial Intelligence techniques. Reference [20] uses a system based on the combination of case-based reasoning and fault tree analysis, and provides a more precise fault diagnosis and obtains technical support for maintenance. The troubleshooting process uses the specific symptoms as input, and the output is a troubleshooting guide tree of similar cases identified. Another paper [21], presents an intelligent decision system for fault diagnosis of aircraft. The C4.5 Algorithm is employed to obtain the best decision trees from the training data available and Principal Component Analysis (PCA) to decrease the dimension of the input data. The results show high correct fault detection rates, low missed detection, and also obtains the false alarm rates. In this case it is necessary to make available a complete and proper dataset regarding the faults for the construction of a model for troubleshooting problems. However, this is not always possible. Other previous work proposes to optimising fault troubleshooting processes that obtain the highest efficacy. The efficacy could depend on various criteria, such as fault probability, cost, and time. The input of this work is a list of suspected components that have been identified as possible causes of failure, and the output is the optimal ordering of troubleshooting tasks to solve the fault in the system. For example, Liu [22] proposes a utility function based on the probability of components suspected of failing and the time required to verify each component in order to obtain the optimal ordering. Furthermore, a mathematical proof is given such that the ordering obtained minimises the mean troubleshooting time, cost, or a combination of the two. This work only takes into account the list of suspected components, while our work is richer because the variables associated with these components are also considered in order to optimise the troubleshooting process.

In reference [23], another approach towards effective troubleshooting decisions is proposed. It is based on Bayesian Networks and the Multi-criteria Decision Approach (MCDA). The efficacy of the troubleshooting process depends on fault probability, cost, time, and risk of repair action. The approach ensures a cost-saving, highly efficient, and low-risk troubleshooting selection in each step, where different alternatives are considered with regard to the previous criteria. An automobile engine startup failure is used as a case study, but fails to show any validation of the results obtained. In [24], the authors propose a framework to detect anomalies in aircraft systems during flight. However, it does not provide a systematic methodology for the resolution of the identified failures.

Recently, a set of new topics have appeared related to automated fault troubleshooting, such as Self-Healing [25], Smart Troubleshooting [10], and Smart Maintenance [26], [27]. These concepts include frameworks, methodologies and related tools. Furthermore, the modelling analysis, and

[2] `http://www.idea.us.es/ts2020/`.

recovery of information from failures is proposed in an automatic or (semi)automatic way. These could provide very interesting alternatives to addressing the problem of the effective fault troubleshooting process.

Finally, some other proposals have been developed to support both monitoring and troubleshooting using machine learning [28], [29], but they did not take into account the change of the observation in the model to reduce time and cost. However, to the best of our knowledge, our solution is the first to automatically create and adapt the troubleshooting problem according to observations, by optimising its solution taking into account the cost of signal reads and component substitutions.

VIII. CONCLUSION AND FUTURE WORK

A proper troubleshooting process, which minimises the time and cost of analysis in the search for causes of failure, is crucial for the daily operation of manufacturing companies. This is especially relevant with troubleshooting processes that work with systems whose components are expensive and whose complexity requires a very tricky analysis that can be extended widely in time, and can cause significant losses and delays. In this paper, we focus on the usefulness of reading variables and the interactivity of the troubleshooting guiding the process to improve its performance. Our solution is based on a graph-oriented approach that provides the user with the necessary information to help improve decision-making in the search for the element responsible for the failure. To this end, we have assumed the existence of 5 essential aspects: (1) a model that contains all the structural relationships between the different elements of the system, which can be interpreted as a graph; (2) the cost and probability associated with the failure of components, which will be used for ranking the causes ordered; (3) the ability to read values in intermediate elements (links); (4) the possibility of transforming a model with MIMO elements into MISOs; and, (5) the capability of being able to exonerate elements through observations. With all this information, it is possible to generate an algorithm capable of following a certain order, indicating the application of actions that can be carried out on each element of the ranking, and the costs that his/her decision entails, as well as acting accordingly, exonerating, blaming or isolating the different elements and variables that explain the malfunctioning of the system. The algorithm has been tested in a real scenario, and the degree of cost reduction that can be achieved is analysed.

Although, in some cases, our solution may seem more tedious than the traditional methodology, our validation and the fact that the experts participate in the project demonstrate that costs are reduced and that the unnecessary substitution of very expensive elements may easily disappear.

As an extension to this paper, we continue to work on improving the ranking order to ensure the optimal way to analyse the system. Furthermore, we are investigating how to include 3 types of multiple failure: more than one single simultaneous failure; failures where there is more than one element responsible for the fault; and failures that produce multiple variables with incorrect states. Consequently, we are also analysing the best way to include information from this multiplicity in the ranking prioritisation function. Finally, we are working on the analysis of decision-making and results obtained in each execution, using machine learning, in an effort to improve the weights given for probability in an automated way and based on real data.

REFERENCES

[1] M. Cordier, P. Dague, M. Dumas, F. Lévy, J. Montmain, M. Staroswiecki, and L. Travé-Massuyès, "A comparative analysis of AI and control theory approaches to model-based diagnosis," in *ECAI 2000*, W. Horn, Ed. Berlin, Germany: IOS Press, Aug. 2000, pp. 136–140.

[2] R. Ceballos, M. T. Gómez-López, R. M. Gasca, and C. D. Valle, "A compiled model for faults diagnosis based on different techniques," *AI Commun.*, vol. 20, no. 1, pp. 7–16, 2007.

[3] D. Borrego, M. T. Gómez-López, and R. M. Gasca, "Minimizing test-point allocation to improve diagnosability in business process models," *J. Syst. Softw.*, vol. 86, no. 11, pp. 2725–2741, Nov. 2013.

[4] L. Travé-Massuyès, T. Escobet, and R. Milne, "Model-based diagnosability and sensor placement application to a frame 6 gas turbine subsystem," in *Proc. 17th Int. Joint Conf. Artif. Intell. (IJCAI)*, B. Nebel, Ed. Seattle, WA, USA: Morgan Kaufmann, 2001, pp. 551–556.

[5] S. Du and L. Xi, "Fault diagnosis in assembly processes based on engineering-driven rules and PSOSAEN algorithm," *Comput. Ind. Eng.*, vol. 60, no. 1, pp. 77–88, Feb. 2011.

[6] E. Frontoni, J. Loncarski, R. Pierdicca, M. Bernardini, and M. Sasso, "Cyber physical systems for industry 4.0: Towards real time virtual reality in smart manufacturing," in *Augmented Reality, Virtual Reality, and Computer Graphics*, L. T. D. Paolis and P. Bourdot, Eds. Cham, Switzerland: Springer, 2018, pp. 422–434.

[7] A. Tedesco, M. Gallo, and A. Tufano, "A preliminary discussion of measurement and networking issues in cyber physical systems for industrial manufacturing," in *Proc. IEEE Int. Workshop Meas. Netw. (M N)*, Sep. 2017, pp. 1–6.

[8] U. Wetzker, I. Splitt, M. Zimmerling, C. A. Boano, and K. Römer, "Troubleshooting wireless coexistence problems in the industrial Internet of Things," in *Proc. IEEE Int. Conf. Comput. Sci. Eng. IEEE Int. Conf. Embedded Ubiquitous Comput., 15th Int. Symp. Distrib. Comput. Appl. Bus. Eng.*, Paris, France: IEEE Computer Society, Aug. 2016, p. 98.

[9] H. Warnquist, J. Kvarnström, and P. Doherty, "A modeling framework for troubleshooting automotive systems," *Appl. Artif. Intell.*, vol. 30, no. 3, pp. 257–296, Mar. 2016.

[10] M. Caporuscio, F. Flammini, J. Thornadtsson, N. Khakpour, and P. Singh, "Smart-troubleshooting connected devices: Concept, challenges and opportunities," *Future Gener. Comput. Syst.*, vol. 111, pp. 681–697, Oct. 2020.

[11] N. Papakostas, P. Papachatzakis, V. Xanthakis, D. Mourtzis, and G. Chryssolouris, "An approach to operational aircraft maintenance planning," *Decis. Support Syst.*, vol. 48, no. 4, pp. 604–612, Mar. 2010.

[12] R. Kannan, S. S. Manohar, and M. S. Kumaran, "Nominal features-based class specific learning model for fault diagnosis in industrial applications," *Comput. Ind. Eng.*, vol. 116, pp. 163–177, Feb. 2018.

[13] Y. Jiang, B. An, M. Huo, and S. Yin, "Design approach to MIMO diagnostic observer and its application to fault detection," in *Proc. IECON-44th Annu. Conf. IEEE Ind. Electron. Soc.*, Washington, DC, USA, Oct. 2018, pp. 5377–5382.

[14] M. Blanke, M. Kinnaert, J. Lunze, M. Staroswiecki, and J. Schrder, *Diagnosis and Fault-Tolerant Control*, 2nd ed. Berlin, Germany: Springer, 2010.

[15] J. C. Chan and J. A. Abraham, "A study of faulty signatures using a matrix formulation," in *Proc. Int. Test Conf.*, Sep. 1990, pp. 553–561.

[16] D. Jung, "A generalized fault isolability matrix for improved fault diagnosability analysis," in *Proc. 3rd Conf. Control Fault-Tolerant Syst. (SysTol)*, Barcelona, Spain, Sep. 2016, pp. 519–524.

[17] T. Ogata, A. Takeuchi, S. Fukuda, T. Yamada, T. Ochi, K. Inoue, and J. Ota, "Characteristics of skilled and unskilled system engineers in troubleshooting for network systems," *IEEE Access*, vol. 8, pp. 80779–80791, 2020.

[18] M. Vomlelová and J. Vomlel, "Troubleshooting: NP-hardness and solution methods," *Soft Comput.—Fusion Found., Methodologies Appl.*, vol. 7, no. 5, pp. 357–368, Apr. 2003.

[19] V. Villani, N. Battilani, G. Lotti, and C. Fantuzzi, "Myaid: A troubleshooting application for supporting human operators in industrial environment," *IFAC-PapersOnLine*, vol. 49, no. 19, pp. 391–396, 2016.

[20] X. P. Yu, Q. Li, and X. Hu, "Aircraft fault diagnosis system research based on the combination of CBR and FTA," in *Proc. 1st Int. Conf. Rel. Syst. Eng. (ICRSE)*, Oct. 2015, pp. 1–6.

[21] S. A. Z. Wang, J.-L. Zarader, and K. Youssef, "A decision system for aircraft faults diagnosis based on classification trees and PCA," in *Intelligent Autonomous Systems 12*. Berlin, Germany: Springer, 2013, pp. 411–422.

[22] J. Liu, "Optimal task ordering for troubleshooting systems faults," in *Proc. IEEE Aerosp. Conf.*, Mar. 2005, pp. 3709–3714.

[23] Y. Huang, Y. Wang, and R. Zhang, "Fault troubleshooting using Bayesian network and multicriteria decision analysis," *Adv. Mech. Eng.*, vol. 6, Jan. 2014, Art. no. 282013.

[24] H. Lee, G. Li, A. Rai, and A. Chattopadhyay, "Real-time anomaly detection framework using a support vector regression for the safety monitoring of commercial aircraft," *Adv. Eng. Informat.*, vol. 44, Apr. 2020, Art. no. 101071.

[25] A. Asghar, H. Farooq, and A. Imran, "Self-healing in emerging cellular networks: Review, challenges, and research directions," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 1682–1709, 3rd Quart., 2018.

[26] Y. Liu, Y. Wu, and Z. Kalbarczyk, "Smart maintenance via dynamic fault tree analysis: A case study on Singapore MRT system," in *Proc. 47th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw. (DSN)*, Jun. 2017, pp. 511–518.

[27] M. Ashjaei and M. Bengtsson, "Enhancing smart maintenance management using fog computing technology," in *Proc. IEEE Int. Conf. Ind. Eng. Eng. Manage. (IEEM)*, Dec. 2017, pp. 1561–1565.

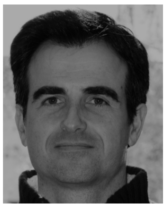
[28] A. D'Alconzo, P. Barlet-Ros, K. Fukuda, and D. Choffnes, "Machine learning, data mining and big data frameworks for network monitoring and troubleshooting," *Comput. Netw.*, vol. 107, pp. 1–4, Oct. 2016.

[29] C. Zhang, Y. He, B. Du, L. Yuan, B. Li, and S. Jiang, "Transformer fault diagnosis method using IoT based monitoring system and ensemble machine learning," *Future Gener. Comput. Syst.*, vol. 108, pp. 533–545, Jul. 2020.

BELÉN RAMOS-GUTIÉRREZ is currently a Software Engineering and Technology Ph.D. Student with the University of Seville. Her research interests include process mining, data extraction, and optimisation for process mining techniques; and troubleshooting and decision support systems in industrial environments. She is also working as a Predoctoral Researcher with the University of Seville and collaborate on projects involving industrial aspects related to logistics-port and aeronautical environments. Her goal is to improve and automate the extraction and processing of heterogeneous data from industrial environments for exploitation with process mining techniques.

MARÍA TERESA GÓMEZ-LÓPEZ is currently pursuing the Ph.D. degree in computer science. She is also a Lecturer with the University of Seville and the Head of the IDEA Research Group. Her research interests include model-based diagnosis in business processes and data management in big data environment. She has led several private and public research projects and has published more than 20 impact articles (DSS, IS, DKE, IST, and so on). She was nominated as a member of several Program Committees (BPM, ER, EDOC, CAISE Doctoral Consortium, and so on). She has been reviewing for several international journals. She has been invited speaker at various conferences and summer schools.

DIANA BORREGO received the Ph.D. degree in computer science. She is currently a Lecturer with the University of Seville and a member of the IDEA Research Group. Her research interests include the verification and diagnosis of business processes through automatic reasoning for their quality improvement. Her works have appeared in international conferences and journals, including *Information and Software Technology*, *Data & Knowledge Engineering*, the *Journal of Systems and Software*, and *Computers in Industry*.



RAFAEL CEBALLOS received the M.Sc. and Ph.D. degrees from the Department Computer Languages and Systems, University of Sevilla, in 2002 and 2011, respectively. He is currently working as an Assistant Professor with the University of Sevilla. He is the author and coauthor of many book chapters, conference papers, and impact journals articles (*Applied Sciences*, *Information and Software Technology*, and *Data & Knowledge Engineering*). He has been awarded in CAEPIA-05 Doctoral Consortium. He was an Invited Speaker with the International Summer School on Fault Diagnosis of Complex Systems in 2019. His research interests include business processes and data management, model-based diagnosis, software testing, and cybersecurity.

RAFAEL M. GASCA received the Ph.D. degree in computer science, in 1998. He is currently a Professor with the University of Seville, Spain, where he has been responsible for the Quivir Research Group since 1999. His research interests include fault diagnosis, cybersecurity technologies, and business process management systems. He has been involved in European and Spanish research projects and has published numerous articles in *Computer Science* journals and international conferences.

ANTONIO BAREA has been working with Airbus since 2007. He is currently the responsible for the Avionics & Mission Systems Technologies Industrialization as a part of the Manufacturing Engineering. He has led many industrial innovations to improve the quality and reduction of costs for system testing process in the aerospace industrial facilities, which are being published or under patent review, such as the Mini Multi Interface Box Simulator (MMIBS), the Prow Radar Obstacle Simulator for Testing (PROST), Flight Recorder Industrial & Integral Data Analyzer (FRIIDA), and the latest, the Automatic Troubleshooting System (ATS).

3.2.4 Process mining to unleash variability management: discovering configuration workflows using logs

Published in Proceedings of the 23rd international systems and software product line conference, SPLC 2019, volume A, Paris, France, September 9-13, 2019, pp. 1–12. ACM, 2019

- ***Authors**: Ángel Jesús Varela Vaca, José Á. Galindo, Belén Ramos Gutiérrez, María Teresa GómezLópez, David Benavides.*

- ***DOI**: `https://doi.org/10.1145/3336294.3336303`.*

- ***Rating**: Class 2 in Ranking SCIE.*

Process Mining to Unleash Variability Management: Discovering Configuration Workflows Using Logs

Ángel Jesús Varela-Vaca, José A. Galindo, Belén Ramos-Gutiérrez
María Teresa Gómez-López and David Benavides
Universidad de Sevilla
Seville, Spain
{ajvarela,jagalindo,brgutierrez,maytegomez,benavides}@us.es

ABSTRACT

Variability models are used to build configurators. Configurators are programs that guide users through the configuration process to reach a desired configuration that fulfils user requirements. The same variability model can be used to design different configurators employing different techniques. One of the elements that can change in a configurator is the configuration workflow, i.e., the order and sequence in which the different configuration elements are presented to the configuration stakeholders. When developing a configurator, a challenge is to decide the configuration workflow that better suites stakeholders according to previous configurations. For example, when configuring a Linux distribution, the configuration process start by choosing the network or the graphic card, and then other packages with respect to a given sequence. In this paper, we present COnfiguration workfLOw proceSS mIning (COLOSSI), an automated technique that given a set of logs of previous configurations and a variability model can automatically assist to determine the configuration workflow that better fits the configuration logs generated by user activities. The technique is based on process discovery, commonly used in the process mining area, with an adaptation to configuration contexts. Our proposal is validated using existing data from an ERP configuration environment showing its feasibility. Furthermore, we open the door to new applications of process mining techniques in different areas of software product line engineering.

CCS CONCEPTS

• **Software and its engineering** → **Software product lines**.

KEYWORDS

variability, configuration workflow, process mining, process discovery, clustering

ACM Reference Format:
Ángel Jesús Varela-Vaca, José A. Galindo, Belén Ramos-Gutiérrez and María Teresa Gómez-López and David Benavides. 2019. Process Mining to Unleash Variability Management: Discovering Configuration Workflows Using Logs. In *23rd International Systems and Software Product Line Conference - Volume A (SPLC '19), September 9–13, 2019, Paris, France.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3336294.3336303

1 INTRODUCTION

Variability models such as Feature Models (FMs) [22] describe commonalities and variabilities in Software Product Lines (SPLs) and are used along all the SPL development process. After an FM is defined, products can be configured and derived. In the configuration and derivation process, users select and deselect features using a *configurator.* A configurator [21][19] is a software tool that presents configuration options to the users in different stages. An example of a configurator tool is KConfig [58] where developers can configure the Linux kernel with more than 12.000 configuration options.

An important aspect of a configurator is to determine the *configuration workflow* [28], i.e., the order in which features and options are presented to configuration stakeholders. For instance, when configuring the Linux kernel using KConfig [58], there can be different user configuration profiles depending on interests or skills. The configuration workflow used by a configurator can impact the user experience in the configuration process. Therefore, selecting a well suited configuration workflow is a challenge. Up to now – to the best of our knowledge– the selection of a configuration workflow is made either intuitively or following the structure and properties of a variability model [21, 66].

In this paper, we present COLOSSI, an approach that takes a feature model and a set of existing configuration logs and automatically retrieves configuration workflows. A configuration log is a set of configurations performed in the past in a given domain taking into account a configuration order. Our solution relies on *process mining* [3] techniques. Process mining is a well established area of business process management that uses different techniques to extract business processes from traces of execution. In our approach, we conceptually map a business process model to a configuration workflow and traces to configuration logs making possible to reuse process mining techniques to infer configuration workflows.

Although using process mining can automatically retrieve configuration workflows, the results can be difficult to interpret to domain engineers in order to build a configurator. This is mainly due to the fact that, very often, mined processes are "spaghetti-like" models in which the same activity needs to be duplicated [62]. To illustrate the difficulty, Figure 1 shows the result of directly applying process mining techniques to the ERP system presented in [46] and detailed in Section 3.

The simplification of spaghetti processes is an open problem in the process mining domain [3]. Variability models have special characteristics that can help to guide the discovery process. To overcome this difficulty, our solution adapts a clustering algorithm

SPLC '19, September 9–13, 2019, Paris, France

ACM ISBN 978-1-4503-7138-4/19/09…$15.00
https://doi.org/10.1145/3336294.3336303

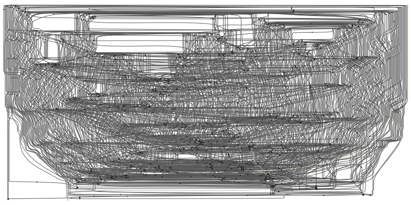


Figure 1: Spaghetti process of the ERP presented in [46].

solution that instead of retrieving a single–complicated configuration workflow, is able to cluster the configurations according to different metrics. Our solution takes information from the variability model as input and retrieves less complex configuration workflows that can assist the development of better configurators.

COLOSSI is validated using an ERP case study taken from [46]. Results show that the metrics of the retrieved configuration workflows are improved by a 99% in the best case and 81% in the worst case with respect to the spaghetti like first solution.

The contributions of this paper are as follows:

- An automated technique based on process mining to select a configuration workflow that better fits the configuration logs according to a set of metrics.
- A validation of the proposal using a realistic ERP scenario.
- An available implementation that can be applied to other datasets.

The remainder of this paper is organised as follows: Section 2 details the solution and concepts that grounds our proposal; Section 3 presents empirical results from analysing COLOSSI; Section 4 presents the related work and Section 5 presents concluding remarks and lessons learned.

2 COLOSSI: CONFIGURATION WORKFLOW PROCESS MINING SOLUTION

In order to create a configuration workflow, a feature model and a configuration log must be combined. Figure 2 shows an overview of the COLOSSI approach. Using the configuration log, it is possible to apply process mining techniques to derive a valid configuration workflow representing all the possible paths defined in the configuration logs. It is likely that the resulting workflow follows the so-called spaghetti-style [62] and it is therefore difficult to understand and manipulate. Nevertheless, it is important to remark that it can be already exploited by process mining automated tools to extract metrics, perform simplification over the workflow as well as many additional analysis. Also, any generated configuration workflow can be already used to build automatically a configurator.

In addition, to the usage of process mining techniques, we propose handling and clustering methods to reduce and group similar configuration traces according to some properties. Those clusters can then be used again as input of process mining techniques to obtaining a set of configuration workflows depending on the observed behaviour of the configuration logs. Those workflows will obtain better metrics with respect to the original complex workflows of step ①. Our conjecture is that the resulting configuration workflows of step ② will better guide the domain engineers in the construction of a configurator as well as the analysis mentioned previously.

Following, we describe the details of the different elements of COLOSSI.

2.1 Inputs

A feature model is an arranged set of features that describes variability and commonality using features and relationships among them. [18, 57]. FMs describe all the potential combinations of features. Figure 3 shows an excerpt of a feature model of the ERP domain where features are arranged in a tree–like structure and different relationships are established among them. FMs can be used to build configurators that are pieces of software that guide the configuration process while selecting and deselecting features. An example of a configurator is KConfig, a tool that helps configuring the Linux kernel. As an FM can define a configuration space defined by all the possible feature combinations, it can also define different possible configuration workflows that can be derived using the same FM.

COLOSSI takes as input a FM and a *configuration log*. To define a configuration log, we use some definitions that are used in process mining area to define events and traces and we map those definitions to define a configuration log.

An event log is a multiset of traces:

Definition 2.1. (Event Log). Let L be an event log $L = [\tau_1, \cdots, \tau_m]$ as a multiset of traces τ_i.

A trace is a tuple with an identifier and a sequence of events that occurred at some point in time:

Definition 2.2. (Trace). Let τ be a trace $\tau = \langle case\_id, \mathcal{E} \rangle$ which consists of a *case_id* which identifies the case, and a sequence of events $\mathcal{E} = \{\varepsilon_1, \cdots, \varepsilon_n\}$, ε_i occurring at a time index i relative to the other events in $\mathcal{E}$.

An event occurrence is a 3–tuple with an identifier of an activity that occurred at some timestamp and that can have additional information:

Definition 2.3. (Event occurrence). Let ε be an event occurrence $\varepsilon = \langle activity\_id, timestamps, others \rangle$ which is specified by the identity of an activity which produces it and the timestamps. It can store more information (i.e., states, labels, resources, etc.)

In COLOSSI, we conceptually map elements from the feature modelling domain to the process mining domain as shown in Table 1. Concretely, an event log is conceptually a configuration log. A trace is an ordered configuration, i.e., a *configuration trace*, thus, it is a set of selected features that follow a given order. Finally, an event occurrence is a feature. Additionally, a feature can have more information like attributes associated with this feature such as preferences, metrics or the like.

2.2 Configuration logs extractor

A configuration log is composed of a set of configuration traces where each configuration trace encodes not only the features of a configuration but the timestamps indicating when each feature

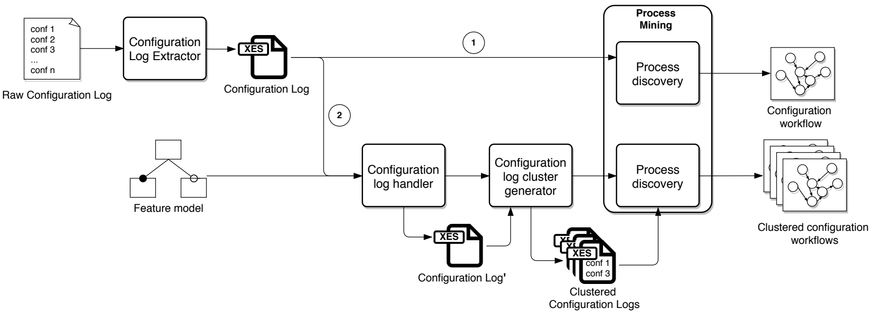
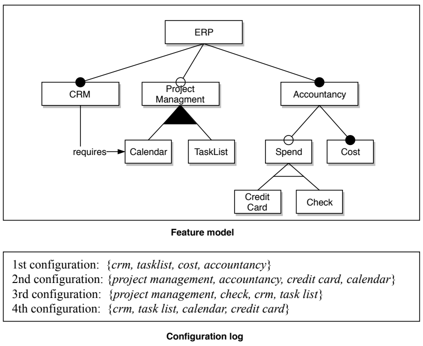


Figure 2: COLOSSI solution overview.



1st configuration: {*crm, tasklist, cost, accountancy*}
2nd configuration: {*project management, accountancy, credit card, calendar*}
3rd configuration: {*project management, check, crm, task list*}
4th configuration: {*crm, task list, calendar, credit card*}

Figure 3: ERP domain based example.

Process Mining	Product Line
Event log	Configuration log
Trace	Configuration trace
Event occurrence	Feature

Table 1: Mapping concepts.

was selected. In a raw configuration log, we can find a diversity of meta-information among the selected or deselected features. Moreover, this meta-information can be presented in a unstructured or structured fashion.

In this first step, we take as input a raw configuration log and output a set of configuration traces. Therefore, we need to *i*) search for the meta-information encoding the timestamps for each feature. Note that this might not be explicit and can be provided using other mechanisms (e.g., line numbers in a plain text format); *ii*) use this meta-information to represent the feature selection order, and; *iii*) store the set of configuration traces in a format that can be used throughout the configuration workflow retrieval process (e.g., XES serialization [1]). After this, we end up with a set of configuration traces that represent the selection order used by the configurator users. However, there might be non-valid configurations and other erroneous configurations w.r.t domain information.

2.3 Configuration logs handler

At this step, the configuration log might contain non-valid configurations, erroneous partial selection of features among other domain-related errors such as those depicted in [6]. To remove clutter and noise out of the workflows, users might prefer to remove such information from the configuration log. This cleaning step consist on removing wrong selection of features (a.k.a non-valid partial configurations) as well as generate metrics that can be latter exploited to optimise the workflow retrieval process. For example, the use of atomic-sets to complete partial configurations.

Depending on the expected workflow usage, domain engineers have to define the meaning of a valid configuration and the metrics to rely on. For example, a SPL engineer might consider only configurations with complete assignments of features to develop a configuration while other might find interesting to consider full assignments (i.e., to configure only the variability part of the product line, keeping aside the common parts).

2.4 Process mining - discovery

Process mining is a family of techniques based on event logs that can be categorised as process discovery, conformance checking and enhancement [63]. In this paper, we are focused on the use of *process mining* to analyse the configuration logs for the discovering of configuration workflows based on the user experiences. Process discovery in process mining brings together a set of algorithms to generate a workflow process model that covers the traces of activities observed in an organisation [40]. The evolution of algorithms during last decades has allowed the discovery of complex models that are able to involve not only the activities executed in the daily work of companies, but also the persons who execute them and the used resources.

Process mining is an important topic that has been well received by the enterprises, bringing about the evolution of the research solution tools (e.g., ProM [64]) to commercial solutions (e.g., Disco™and Celonis™). This facilitates its applicability to several contexts and areas, although variability has been out of the scope of these techniques before this paper.

Process discovery in process mining uses a set of traces similar to the configuration log shown in Figure 3, to obtain a model that covers the possible traces. Figure 4 shows the process discovered by Disco tool-suite, which covers every possibility configuration trace. The relational patterns among the definition of the features become part of the model. For example, two features can be the first in the traces (*CRM* or *Project management*) or after *CRM* always *Task List* is selected. Figure 4 also shows the number of traces that are represented by each transition, giving information about the importance o each part of the traces in the obtained model.

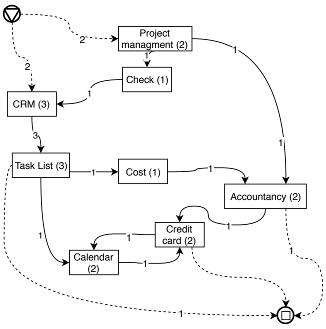


Figure 4: Process discovered for configuration log of ERP domain based example.

In the framework proposed in this paper (Figure 2), *Process mining - process discovery* module enables to read an event log and generates a process model that fits these traces. In the case of the variability context, a *configuration log* is read and a *configuration workflow* is obtained using the same techniques used for classical process mining.

2.5 Configuration logs cluster generator

Configuration processes have a high degree of variability, specially when the configuration order is defined by human decisions. The application of process discovery in this type of scenarios tends to produce spaghetti-like processes, being necessary to apply a pre-processing. Configurability contexts are specially variable in relation to the executed activities derived from the high human intervention, thereby, we propose to divide the traces into subsets, to model different profiles of users and avoiding the discovery of non-user understandable processes. In these contexts, where process discovery is used to infer spaghetti-like processes, frequently clustering techniques such as a pre-processing step are applied [26]. To adapt the solution to configuration tasks, we propose the division of the configuration traces into multiple clusters before the application of a process discovery. This division lets to discover configuration workflows with more quality. This section describes what a cluster is and the metric (e.g., entropy) used to divide the traces among them. In following sections, we describe how quality is measured and how the clusters can be created.

Being L a configuration log composed of a set of configuration traces (i.e., $[\tau_1, \cdots, \tau_m]$), a cluster is a subset of configuration traces from L that complies certain properties.

Definition 2.4. (Cluster of Configuration Traces). A cluster of configuration logs, $c = [\tau_i, \cdots, \tau_j] \subseteq L$, where $\forall \tau_k \subseteq L$, $\exists c \mid \tau_k \in$ c and $\nexists$ c' ≠ c where $\tau_k \in$ c'.

The distribution of configuration traces between various clusters depends on the purpose of the practitioners. In our case, the goal is to group the more similar configuration traces. In this paper, the meaning of 'similar' is related to both features and transitions involved in the logs. For this reason, we adapted the classical information entropy metric [37] by introducing two different custom entropy metrics for clustering in the configuration context:

- *Entropy-features* ($S_{features}$) of a cluster: a metric which measures the similarity between a set of traces according to the features that belong to the same cluster. Thus, it is the ratio between the number of features that do not appear in all configuration traces ($features_{nat}$) and the number of different features in all the configuration traces ($features_{diff}$):

$$S_{features} = \frac{|features_{nat}|}{|features_{diff}|} \tag{1}$$

- *Entropy-transitions* ($S_{transitions}$) of a cluster: a metric which measures the similarity between a set of traces according to the transitions that belong to the same cluster. Thus, it is the ratio between the transitions that do not appear in all configuration traces ($transitions_{nat}$) and the number of different transitions in all the configuration traces ($transitions_{diff}$):

$$S_{transitions} = \frac{|transitions_{nat}|}{|transitions_{diff}|} \tag{2}$$

In order to illustrate the calculation of entropies, the $S_{features}$ and $S_{transitions}$ for the *Cluster 1* and *Cluster 2* of Figure 5 are determined in Table 2.

	Entropy-features	*Entropy- transitions*
Cluster 1	$\frac{0}{4} = 0$	$\frac{0}{3} = 0$
Cluster 2	$\frac{6}{8} = 0, 75$	$\frac{6}{15} = 0, 4$

Table 2: Entropies for the clusters of the Figure 5.

Note that the range of the entropy is [0..1]. The values of entropy that are close to 0 represent more similar traces, whilst when they are close to 1 represent that there are different features involved in the traces of the cluster. The best configuration of clusters obtained from a set of configurations traces is the one that minimize the summation of the entropy of all clusters obtained. The challenge is how to obtain the best configuration of clusters as a pre-processing of a process discovery.

In order to find out the best configuration traces divided into clusters, minimising the entropy of the resulting clusters, different algorithms for clustering can be used. In accordance with [30] *clustering* provides an unsupervised classification of patterns (observations, data items, or feature vectors) into groups (clusters). *Clustering* [31] brings together a large set of algorithms that can be classified in different ways according to the point of view necessary in the case of study. We propose the use of the well-known *hierarchical agglomerative clustering* algorithm base on the work in [69] as the combination of hierarchical and agglomerative clustering.

On the first hand, *hierarchical clustering* is defined as a procedure to form hierarchical groups of mutually exclusive subsets, each of which has members that are maximally similar with respect to the specified characteristics [69]. In the same study, authors define the process as: assuming we start from *n* sets, it permits their reduction to $n - 1$ mutually exclusive sets by considering the union of all possible $\frac{n(n-1)}{2}$ pairs and selecting a union having a maximal value for the objective function.

On the other hand, *agglomerative clustering* is an algorithm that starts from the assumption that each element constitutes a cluster by itself (singleton) and it successively merges these singletons together forming clusters until a stopping criterion is satisfied which is also determined by the objective function.

The characteristics used in our solution is based on both entropies presented (features and transitions), combined with the objective function the Ward's minimum variance method [69]. This function aims to minimise the sum of the squared differences within all clusters, which means, a variance-minimising approach. Figure 5 shows the obtained dendrogram[1] for the example in Figure 3 by using Entropy-features.

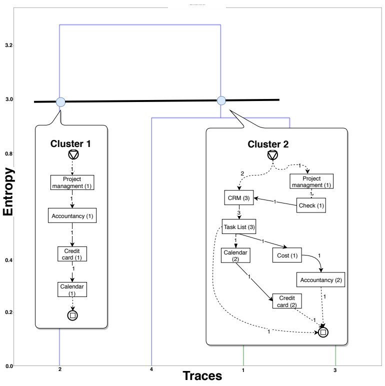


Figure 5: Clustering for the ERP excerpt using the *Entropy-features.*

It is well-known that every clustering algorithm builds a distance matrix during its execution based on a given methodology (euclidean, manhattan, etc.). However, *hierarchical agglomerative clustering* is an exception, since it can be carried out from the distance matrix itself. We consider the entropy matrix as the distance matrix, this leads us to decide for this method of clustering, which can be performed also using other methods, such as *single-linkage*, *complete-linkage*, *average-linkage*, and *Ward*.

Whenever a clustering process is executed, one of the first problems to deal with is to decide which is the optimal number of clusters. Many studies carried out about the discovery of the optimum number of clusters, therefore, we decided to study a significant number of them to choose by voting the number of clusters that most indicators had selected as optimal. In this regard, 17 different indicators [4, 5, 9, 14, 16, 17, 20, 24, 25, 29, 32, 33, 42, 44, 45, 50, 51] are used as reference to choose the best number of clusters adapted to our scenario.

By using these indicators with the different clustering methods mentioned above, all of them selected a too large number of clusters as the optimal solution. In addition, the values of the indicators themselves are in some of the cases too dissimilar.

For all the methods the range [0-10] is proposed to determine the number of clusters. In the *single-linkage* method always retrieved the maximum (i.e., 10). For the *average-linkage* retrieved always an average of 2 clusters. Both methods help to bound the limits of number of clusters between 2 and 10. Only with the *complete-linkage* and *Ward* methods, more homogeneous, similar and an assumable number of clusters as results were obtained for the dendrograms showed in Figures 6 and 7.

Finally, by making use of dendrograms, it is observed that applying the *Ward* method, the samples becomes better distributed among the clusters, thereby, generating more differentiated clusters and better structured dendrograms. This fact determines the *Ward's* method as the best option for our approach.

2.6 Leveraging the results of COLOSSI

The COLOSSI approach can be used in different scenarios to leverage process mining in variability management. One of the scenarios presented in this paper is the building of configurators. However, we envision other areas where process mining can be used to automate different tasks. Next, we describe those scenarios, also related to software product lines, from our experience and perspective:

- **Configurator building**. Up to now, configurators building is performed using manual mechanisms or, at most, using the information present in the variability model (e.g., tree traversal in feature models). With COLOSSI, we open the door to use existing configuration logs to build configurators. This novel approach can open the door to new ways of assisting configurators builders by using the generated configuration workflow to optimise configurators.
- **Data analysis**. From the generated configuration workflow it is possible to perform many analysis in terms of graph metrics. Deadlocks identifications, misalignment analysis, metrics extraction –to just mention a few– are areas where process mining techniques can be useful.
- **Testing**. From the data extracted in the former item, it could be possible to define new sampling techniques [61] that can improve the identification of bugs or feature interactions in existing product lines.
- **Variability reduction**. One of the challenges for companies that develop software product lines is variability reduction [8]. While variability is a must in a software product line approach, it is always difficult to find a trade off between a high degree of variability and a systematic management of such a variability. In this context, experts claim for techniques and tools to reduce variability while preserving configurability. Process mining techniques presented in this paper can be a

[1] *Dendrogram* is a branching diagram which represents the arrangement of the clusters produced by the corresponding analyses

first step towards defining tools to assist in the decision of variability reduction.

- **Reverse engineering**. One of the inputs used when reverse engineering feature models are configurations (a.k.a. product matrix). We envision that the techniques described in this paper can be used in reverse engineering of variability models. For instance, the generated configuration workflow can be analysed to better guide reverse engineering algorithms

3 EVALUATION

In this section we present the evaluation of COLOSSI. The evaluation consists of the application of the framework detailed in Section 2 to a configuration log obtained from a real scenario. The possible clusters derived from the application of the defined entropies are analysed.

3.1 Experimentation data

In order to analyse the applicability of our example in a configuration real scenario, we used the raw information from [46]. The used data include a configuration model representing a real ERP, as well as a raw configuration log. The ERP feature model has 1920 features and 59044 cross-tree constraints. Also, the configuration log is formed of 35193 event occurrences that represent a total of 170 different configuration traces with an average of 207 features per configuration trace.

3.2 Framework application

In this section we detail each task of the framework presented in Figure 2.

3.2.1 Configuration Log Extractor. The input data of the configuration of the ERP is represented in a CSV file with two elements, the configuration id and the feature that is configured. Note that a feature can appear in one or more traces, but no more than once in the same trace. Then, the timestamp required to extract the traces was taken by the line number in which the features where appearing throughout the file in a sequential order. This is, we assume that the timestamps were implicit based on the order of appearance (i.e., line numbers) then, transformed them into a more standard format for traces. Concretely we use in our solution the IEEE Standard for eXtensible Event Stream (XES) [1]. This is a standard to serialise, store, exchange events data and it is commonly used in process mining techniques.

3.2.2 Configuration Log Handler. To clean up the set of configurations retrieved by the extractor we decided to consider only valid partial and full configurations. This filtering operation is performed by using the FaMa framework [7]. After filtering, the valid partial configurations using automated analysis [6], we ended up considering 61 configuration traces from the initial set of 170.

3.2.3 Configuration Log Cluster Generator and Process Discovery. Figure 6 and 7 represent the obtained clusters according to the dendrogram built by means of the entropy of features and transitions analysis respectively. In the case of feature entropy (Figure 6), five clusters are obtained. On the other hand, when the transition entropy is used, three clusters are derived to split the configuration logs into simpler configuration workflows. In the following subsections, the details about the obtained configuration workflows and clusters are analysed.

3.3 Analysis of Results

In order to evaluate how the application of clustering can improve the configuration workflows obtained by COLOSSI, in this section, we compare the models discovered: (1) the *original* configuration logs obtained from the initial raw configuration log (cf. Section 3.1); (2) the *filtered* version of the same log including only valid configurations (i.e., after applying *configuration log handler*), and; (3) two set of clusters based on the proposed entropies (features and transitions explained in Section 2.5).

The analysis is carried out following two different perspectives: (1) the analysis of the discovered configuration workflows and (2) the analysis of the set of configuration traces involved in each cluster used in the process discovery.

3.3.1 Analysis of discovered configuration workflows. First, highlight that inductive process discovery techniques used by COLOSSI, ensure the soundness and correctness of the process models obtained [35]. Thus, an analysis of the soundness and correctness of the configuration workflows are unnecessary since processes discovered is always complete, have a proper completion, and have no dead transitions.

However, the complexity of the configuration models is affected by the number of features, the number of configuration traces and the number of transitions. Obviously, the filtering of the configuration traces or the division of the logs will bring about simpler configuration workflows. Figure 8 depicts in a comparative way the number of features, configuration traces and transitions of the set of configuration logs using in each scenario: original configuration log, filtered configuration log only with valid traces, the 5 clusters obtained by using entropy-features, and the 3 clusters obtained by using entropy-transitions.

Regarding general parameters, there is an enormous difference between the original, filtered version, and both that use clustering. The original configuration workflow contains 1652 features and 3330 transitions, whilst the filtered version has one less magnitude order of features and transitions. The clusters seem similar to the filtered version, however, each cluster contains, at least, less than half regarding features. In case of the transitions, the filtered version reached four times fewer transitions than the original. However, clusters reached in the worst case 500 transitions less than the filtered and more than one third fewer transitions in the best case. Due to the clusters group a set of similar traces, the number of traces is intrinsically smaller regarding the configuration traces of the original and filtered configuration workflow.

In conclusion, clusters enable to reduce the complexity of configuration workflow discovered by reducing the configuration traces involved in the same configuration workflow. However, the question is what level the quality of the obtained workflow is improved, and which distribution of cluster-entropy works better.

In literature, several metrics are used to measure how "good" is a design of a business process model [10, 43, 48]. Discovered configuration workflows are also processes with features instead of activities, therefore, these metrics can be adapted to measure

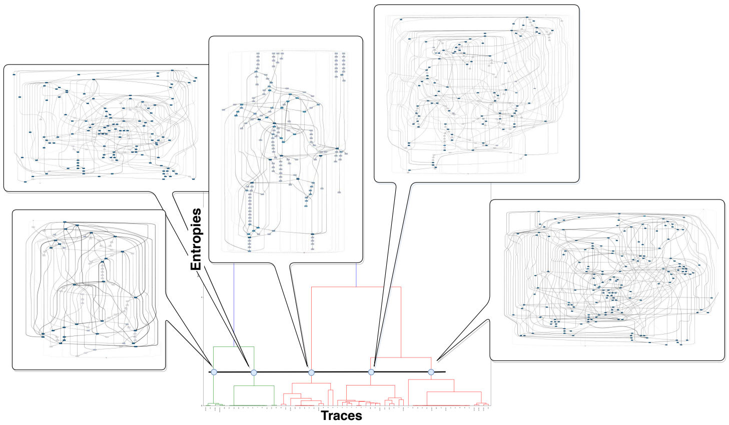


Figure 6: Clustering for the ERP example using the *Entropy-features.*

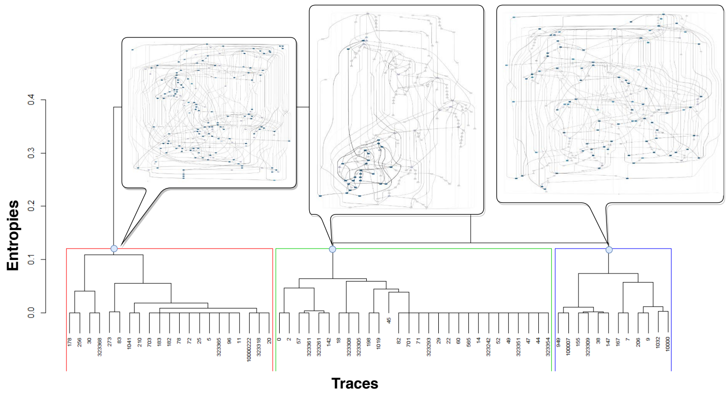


Figure 7: Clustering for the ERP example using the *Entropy-transitions.*

the quality of our obtained configuration workflows. The next set of metrics is adapted to measure the understandability and the complexity of the configuration workflows to compare the four discovered configuration workflows:

- *Density*: the ratio of transitions divided by the maximum number of possible transitions. The lower the value of density, the higher the understandability and the lower complexity.
- *Cyclomatic number* (*CC*): the number of paths needed to visit all features. The cyclomatic can be seen as a complexity metric, thus, the lower the value of *CC*, the lower the level of complexity.
- *Coefficient of connectivity* (*CNC*): the ratio of transitions to features. The greater the value of *CNC*, the greater the complexity of configuration workflows. Although, the authors in [43] remark that models with the same *CNC* value might differ in complexity regarding this parameter.
- *Control Flow Complexity* (*CFC*) enables to measure the complexity in terms of the potential transitions after a split depending on its type. The greater the value of the *CFC*, the greater the overall structural complexity of a workflow.

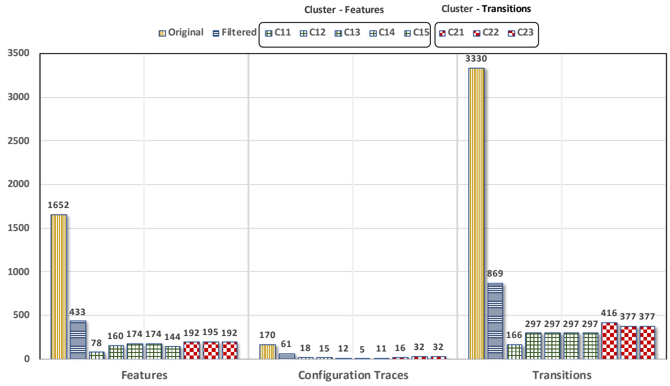


Figure 8: Characteristics of the configurations logs.

Table 3 shows the results of these metrics obtained from the discovered configuration workflows. As explained, for every metric the lower value they take, the better the understandability and less complexity. We shall highlight that the metrics associated with the different clusters are aggregated as arithmetic means for a better comparison.

Config. Workflow	*Density*	*CC*	*CNC*	*CFC*
Original	0,00123	1679	2,016	1677
Filtered	0,00464	437	2,069	434
Cluster-Features	0,01469	125,8	1,892	118,4
Cluster-Transitions	0,00632	118,8	1,213	156

Table 3: Metrics of the configuration workflows.

It is interesting to note how configuration workflow for the original has the lowest density in comparison with the others. This is because the number of features is so high and it compensates the largest number of transitions. However, it has the highest complexity compared to *CC* and *CFC*. However, no conclusions can be achieved with regard to the complexity based on *CNC*. Therefore, although the density is the lowest, the other three metrics demonstrate that the workflow for the original is very complex and misunderstood. On the other hand, the workflow for the filtered version has the highest density, thus, it is the less understandable regarding to this metric. However, the complexity shown by the other metrics demonstrates that it is more understandable and less complex than the original configuration workflow.

Comparing the workflow for the filtered version with both cluster versions, we can conclude that both clusters are more understandable due to lower values related to the four complexity metrics, (i.e., density, *CC*, *CNC* and *CFC*).

In fact, these four metrics help us to know the complexity and understandability of the configuration workflows from the design perspective and the elements in the model. Nevertheless, these metrics used to measure the quality of the workflow are inconclusive to measure the real usefulness and quality of the discovered workflows applied to the context of the variability management.

3.3.2 *Analysis of clustering.*

As introduced in previous sections, two different entropies are applied to infer the clustering (i.e., *Entropy-features* and *Entropy-transitions*). The two entropy formulas can help to understand the quality of the workflow. Thus, a lower value of entropy more quality of the cluster, hence, workflow has more quality. In this regard, Table 4 gives the values regarding the number of clusters and the entropy for each configuration workflow. In this case, the metrics associated with the clusters are aggregated as arithmetic means for better comparison. It is important to highlight that the clusters group a less number of configuration traces, the entropy of the clusters (as mean) are less in both cases than the original and the filtered solution.

Config. Workflow	*N. of Clusters*	*Entropy features*	*Entropy transitions*	*Quality* $(\Delta_{\equiv})$
Original	1	1	0,021	1598,84
Filtered	1	1	0,027	365,18
Cluster-Features	5	0,188	-	59,02
Cluster-Trans.	3	-	0,0052	115,037

Table 4: Comparison of the number of clusters, entropy and quality.

In order to compare the clusters, the entanglement metric is determined as shown in Figure 9. The entanglement indicates the relation between two dendrograms charts, thus, two different distributions of clusters. The range of the entanglement is $[0..1]$. The entanglement values closer to 0 are better than to 1. Thus, the entanglement helps to understand how similar are the dendrograms, thereby, how similar clusters are with the independence of the entropy. In this case, the entanglement is $0,38$ which indicates that both clusters are very similar, in other words, the entropy used to perform the clusters reach similar cluster distributions in this case.

As previously mentioned, the uselessness of the quality metrics related to the workflows leads us to define a new custom metric which enables to establish the quality level of the workflow by relating the number of features and their occurrence within the discovered workflow of a cluster. Thus, a metric that enables us

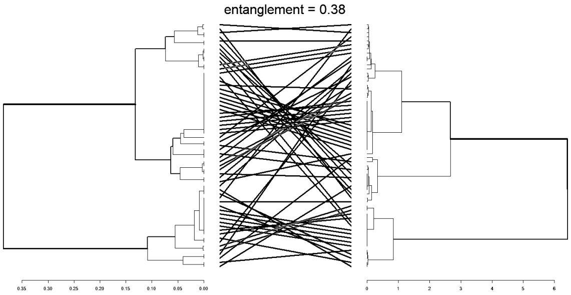


Figure 9: Comparative of entanglement between clusters (*Entropy-features* (right) and *Entropy-transitions* (left)).

to measure how spaghetti is the workflow obtained. Our custom-quality metric is defined as follows:

- Quality ($\Delta_{\equiv}$) measures the difference between the total number of features and the ratio of the sum of the number of times that a feature is selected for each configuration trace and the number of configuration traces.

Formally, given a workflow based on a set of configuration traces (CT) and a set of features ($Features$), the quality can be determined as the following formula:

$$\Delta_{\equiv} = |Features| - \sum_{f \in Workflow} \frac{occurrences(f)}{|CT|} \tag{3}$$

The range of the quality is $[0..|Features|]$, the lower value of quality indicates a better configuration workflow. The number of features and configuration traces are group into the more similar workflow, therefore, it brings about that the quality is near to 0.

For instance, using the example in Figure 3 and the *Cluster 2* in Figure 5, the number included in the rectangle of the feature corresponds to the number of times that the feature is selected regarding the configuration traces. Hence, the quality for the *Cluster 2* can be determined applying the formula as follows:

$$\Delta_{\equiv} = 8 - \left(\frac{3}{3} + \frac{3}{3} + \frac{1}{3} + \frac{1}{3} + \frac{3}{3} + \frac{1}{3} + \frac{2}{3} + \frac{2}{3}\right) \approx 2,67 \tag{4}$$

Comparing the quality results in Table 4, the conclusion is that the original configuration workflow obtains the worst quality and the five clusters obtained using entropy-features have the best quality. Thus, the distribution of configuration traces in the five clusters (cf., Cluster-Features in Figure 8) achieves better results than the original, filtered, even another cluster regarding. In conclusion and according to the defined quality, the Cluster-Features are less complex, more understandable and less spaghetti than the other configuration workflows.

3.4 COLOSSI implementation

COLOSSI is supported by the implementation of a tool which is composed of the next main components:

(1) *Configuration log extractor* is a piece of software module which takes a set of raw configuration log (including timestamps) in a semi-structured format and returns a XES file.
(2) *Configuration log handler* is another piece of software which takes a FM and a XES log as input. First, apply a set of operations over the FM as described in Section 2.3. Then, a data cleaning is carried out over the XES log to get a filtered configuration log. The output of this connector is a new XES log with the filtered configuration log.
(3) *Cluster generator* is a Python/R module which takes a XES log file which is translated into a matrix. This matrix enables the entropy calculation and based on the analysis of certain parameters and the dendrogram, the number of clusters is determined. Using this information, a hierarchical agglomerative clustering algorithm is applied to determine the clusters. A new XES log file is generated for each cluster that composed the final output of this component.
(4) *Discovery connector* is a piece of software which gets the XES file logs of each cluster and automatically feed the ProM to discover the process models by means of the *Inductive Miner*. The output of this component is a process model in Petri-net or BPMN format.

All the resources, thus, configuration logs, the XES files, the workflows discovered, the source code of the COLOSSI tool (i.e., git repository), and a Jupyter notebook that are employed in this work are freely available at [2]http://www.idea.us.es/splc2019/. The notebook is self-explanatory and allows users to work interactively by executing step-by-step instructions to get the clusters.

3.5 Threats to validity

Even though, the experiments presented in this paper provide evidences that the solution proposed is valid, there are some assumptions that we made that may affect their validity. In this section, we discuss the different threats to validity that affect the evaluation.

External validity. The inputs used for the experiments presented in this paper were either realistic or designed to mimic realistic feature models. However, we do not control the development process and it may have errors and not encode all ERP configurations.

The major threats to the external validity are:

[2] DOI:10.5281/zenodo.3236337

- *Population validity*: the ERP feature model that we used may not represent all ERP realistic products. Note that the model was provided after an anonimisation process. Moreover, the timestamps used to derive the traces were relying on the appearance within the input file without an explicit enumeration. To reduce these threats to validity, we chose a single large model that was used in different studies in literature.
- *Ecological validity*: while external validity, in general, is focused on the generalisation of the results to other contexts (e.g., using other models), the ecological validity if focused on possible errors in the experiment materials and tools used. To avoid as much as possible such threats, we relied on previously existing algorithms to perform the process discovery.

Internal validity: concretely, we developed several metrics that reveals different properties of the workflows, however, there might be characteristics of such workflows that are not revealed.

4 RELATED WORK

In this section, we go through the related work of this research.

Configuration workflows. A formal description of configuration workflows is given in [27]. However, a configuration workflow is a bit different from our definition. An activity of the configuration workflow can be mapped to more than just a feature as in our case. However, our approach is complementary because in the handling process we can group different features as well. Furthermore, although formal semantics and automated support for configuration workflows is presented, no automated mechanism is developed to automatically generate configuration workflows from existing configuration logs. In that sense, our approach complements theirs.

Different possible feature orders are defined in [21]. Those orders are used to build web–based configurators hiding the details of the concrete variability model flavours (e.g., OVM, FMs, CVL, etc.). The orders are built from the structure of the variability model. For instance, in the case of FMs, pre–order, pos–order or in–order can be used to determine the feature order in which features are presented to the user. COLOSSI differs from this approach because we use as input configuration logs to automatically derive and cluster configuration workflows. Our approach can be complementary to [21] because different existing workflows could be also measured using process alignment metrics to determine what's the best feature order to be used.

There exist other approaches [67, 68] focused on the field of product configurator design in which configuration workflows has been tackled from the perspective of machine learning.

Application of process mining in different contexts. In order to discover the processes followed by users or systems analysing event logs, process mining has been applied in several scenarios. Depending on the scenario, different are the points of view that could be used to discover a process, such as the activities executed, persons involved, the resources used, the location where the actions occurs, etc. The versatility of process mining techniques has brought about its application to several scenarios [13], being healthcare [38, 49, 52] and IT [39, 47, 55] the most active areas.

The case studies where event logs are produced by human behaviour interactions are specially complex, derived from the free will capacity of the persons that is not always possible to be modelled. This is the context of this paper, where configuration tasks describe the interaction of users with systems. Previous examples in previous scenarios have been developed, such as [2], to analyse how the users interacts with an enterprise resource planning software, or the applicability of software scenarios analysing how the users interact with software to promote improvements about functional specifications or usability aspects [54]. Software development has also provided a complex scenario where process mining can provide mechanism to improve and optimise the known as software process mining [53]. However, configurability issue has not been analysed before with process mining.

High variability in process mining. When there is a high human interaction, as in configuration processes, spaghetti and lasagna processes tend to be obtained. The occurrence of infrequence activities or non-repeated sequence of activities in the analysed log events bring about the necessity to apply frequency-based filtering solutions [12] and other based on the discovery of chaotic set of activities that an be frequent [60].

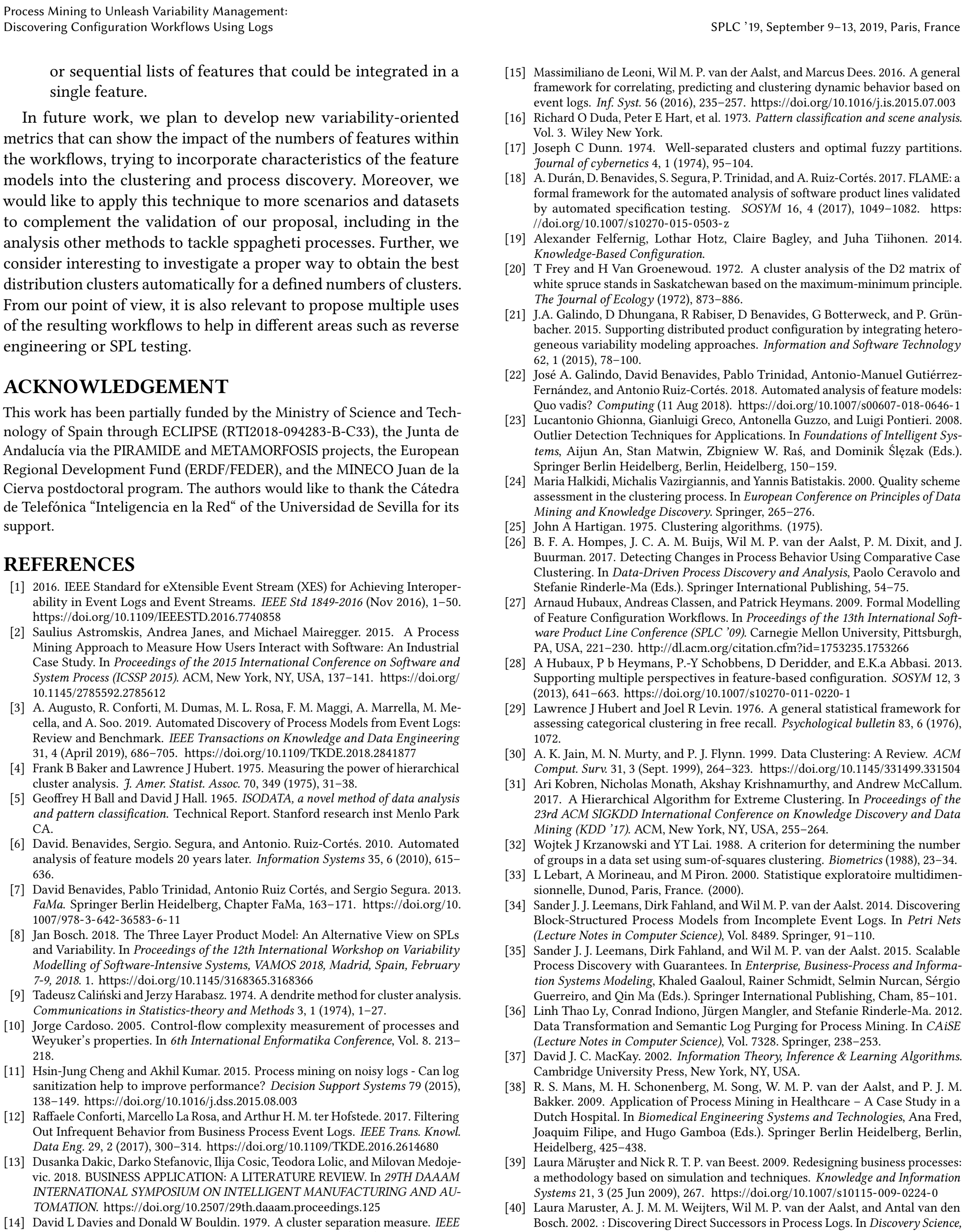
The infrequence patterns in process discovery are frequently treated as noise [36], being removed from the log traces to discover a process that represent the most frequent behaviour [56]. Different types of filtering can be performed: (i) filtering the events that are not belong to the mainstream behaviour [12, 56]; (ii) integrating the filtering as a part of the discovery [34, 41, 65, 70]; (iii) filtering traces, in an unsupervised [23] or supervised way [11], and; (iv) including a previous steps for clustering the problem, facilitating the discrimination of traces according to different points of view or dividing different types of behaviour [15, 59].

In summary, up to our knowledge, this is the first solution for workflow retrieval in SPL-related contexts. It is also relevant the use of process mining techniques in new domains. This paper aims at promoting sinergies between these two areas of study.

5 CONCLUDING REMARKS & LESSONS LEARNED

In this paper, we have coped with the problem of extracting the actual workflows used by SPL configurators analysing configuration logs. To discover configuration workflows, we decided to rely on process mining techniques. Moreover, we propose to apply clustering to improve the resulting configuration workflows reducing the complexity and improving their understandability. From our research on configuration workflows, we learned the following important lessons:

(1) **Reduce the complexity of the configuration workflows.** We have defined a mechanism based on clustering to divide the configuration logs into smaller configurations groups to facilitate the understanding of the configuration workflows inferred from configuration logs.
(2) **Quality measurement.** We have defined a set of metrics adapted from business process literature, to measure the quality of the obtained clusters and configuration workflows.
(3) **Improving decisions about configurators.** The clustering creation and the analysis provide information to expert users about the features that used to be configured together



or sequential lists of features that could be integrated in a single feature.

In future work, we plan to develop new variability-oriented metrics that can show the impact of the numbers of features within the workflows, trying to incorporate characteristics of the feature models into the clustering and process discovery. Moreover, we would like to apply this technique to more scenarios and datasets to complement the validation of our proposal, including in the analysis other methods to tackle sppagheti processes. Further, we consider interesting to investigate a proper way to obtain the best distribution clusters automatically for a defined numbers of clusters. From our point of view, it is also relevant to propose multiple uses of the resulting workflows to help in different areas such as reverse engineering or SPL testing.

ACKNOWLEDGEMENT

This work has been partially funded by the Ministry of Science and Technology of Spain through ECLIPSE (RTI2018-094283-B-C33), the Junta de Andalucía via the PIRAMIDE and METAMORFOSIS projects, the European Regional Development Fund (ERDF/FEDER), and the MINECO Juan de la Cierva postdoctoral program. The authors would like to thank the Cátedra de Telefónica "Inteligencia en la Red" of the Universidad de Sevilla for its support.

REFERENCES

[1] 2016. IEEE Standard for eXtensible Event Stream (XES) for Achieving Interoperability in Event Logs and Event Streams. *IEEE Std 1849-2016* (Nov 2016), 1–50. https://doi.org/10.1109/IEEESTD.2016.7740858

[2] Saulius Astromskis, Andrea Janes, and Michael Mairegger. 2015. A Process Mining Approach to Measure How Users Interact with Software: An Industrial Case Study. In *Proceedings of the 2015 International Conference on Software and System Process (ICSSP 2015)*. ACM, New York, NY, USA, 137–141. https://doi.org/10.1145/2785592.2785612

[3] A. Augusto, R. Conforti, M. Dumas, M. L. Rosa, F. M. Maggi, A. Marrella, M. Mecella, and A. Soo. 2019. Automated Discovery of Process Models from Event Logs: Review and Benchmark. *IEEE Transactions on Knowledge and Data Engineering* 31, 4 (April 2019), 686–705. https://doi.org/10.1109/TKDE.2018.2841877

[4] Frank B Baker and Lawrence J Hubert. 1975. Measuring the power of hierarchical cluster analysis. *J. Amer. Statist. Assoc.* 70, 349 (1975), 31–38.

[5] Geoffrey H Ball and David J Hall. 1965. *ISODATA, a novel method of data analysis and pattern classification.* Technical Report. Stanford research inst Menlo Park CA.

[6] David. Benavides, Sergio. Segura, and Antonio. Ruiz-Cortés. 2010. Automated analysis of feature models 20 years later. *Information Systems* 35, 6 (2010), 615–636.

[7] David Benavides, Pablo Trinidad, Antonio Ruiz Cortés, and Sergio Segura. 2013. *FaMa.* Springer Berlin Heidelberg, Chapter FaMa, 163–171. https://doi.org/10.1007/978-3-642-36583-6-11

[8] Jan Bosch. 2018. The Three Layer Product Model: An Alternative View on SPLs and Variability. In *Proceedings of the 12th International Workshop on Variability Modelling of Software-Intensive Systems, VAMOS 2018, Madrid, Spain, February 7-9, 2018.* 1. https://doi.org/10.1145/3168365.3168366

[9] Tadeusz Caliński and Jerzy Harabasz. 1974. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods* 3, 1 (1974), 1–27.

[10] Jorge Cardoso. 2005. Control-flow complexity measurement of processes and Weyuker's properties. In *6th International Enformatika Conference*, Vol. 8. 213–218.

[11] Hsin-Jung Cheng and Akhil Kumar. 2015. Process mining on noisy logs - Can log sanitization help to improve performance? *Decision Support Systems* 79 (2015), 138–149. https://doi.org/10.1016/j.dss.2015.08.003

[12] Raffaele Conforti, Marcello La Rosa, and Arthur H. M. ter Hofstede. 2017. Filtering Out Infrequent Behavior from Business Process Event Logs. *IEEE Trans. Knowl. Data Eng.* 29, 2 (2017), 300–314. https://doi.org/10.1109/TKDE.2016.2614680

[13] Dusanka Dakic, Darko Stefanovic, Ilija Cosic, Teodora Lolic, and Milovan Medojevic. 2018. BUSINESS APPLICATION: A LITERATURE REVIEW. In *29TH DAAAM INTERNATIONAL SYMPOSIUM ON INTELLIGENT MANUFACTURING AND AUTOMATION.* https://doi.org/10.2507/29th.daaam.proceedings.125

[14] David L Davies and Donald W Bouldin. 1979. A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence* 2 (1979), 224–227.

[15] Massimiliano de Leoni, Wil M. P. van der Aalst, and Marcus Dees. 2016. A general framework for correlating, predicting and clustering dynamic behavior based on event logs. *Inf. Syst.* 56 (2016), 235–257. https://doi.org/10.1016/j.is.2015.07.003

[16] Richard O Duda, Peter E Hart, et al. 1973. *Pattern classification and scene analysis.* Vol. 3. Wiley New York.

[17] Joseph C Dunn. 1974. Well-separated clusters and optimal fuzzy partitions. *Journal of cybernetics* 4, 1 (1974), 95–104.

[18] A. Durán, D. Benavides, S. Segura, P. Trinidad, and A. Ruiz-Cortés. 2017. FLAME: a formal framework for the automated analysis of software product lines validated by automated specification testing. *SOSYM* 16, 4 (2017), 1049–1082. https://doi.org/10.1007/s10270-015-0503-z

[19] Alexander Felfernig, Lothar Hotz, Claire Bagley, and Juha Tiihonen. 2014. *Knowledge-Based Configuration.*

[20] T Frey and H Van Groenewoud. 1972. A cluster analysis of the D2 matrix of white spruce stands in Saskatchewan based on the maximum-minimum principle. *The Journal of Ecology* (1972), 873–886.

[21] J.A. Galindo, D Dhungana, R Rabiser, D Benavides, G Botterweck, and P. Grünbacher. 2015. Supporting distributed product configuration by integrating heterogeneous variability modeling approaches. *Information and Software Technology* 62, 1 (2015), 78–100.

[22] José A. Galindo, David Benavides, Pablo Trinidad, Antonio-Manuel Gutiérrez-Fernández, and Antonio Ruiz-Cortés. 2018. Automated analysis of feature models: Quo vadis? *Computing* (11 Aug 2018). https://doi.org/10.1007/s00607-018-0646-1

[23] Lucantonio Ghionna, Gianluigi Greco, Antonella Guzzo, and Luigi Pontieri. 2008. Outlier Detection Techniques for Applications. In *Foundations of Intelligent Systems*, Aijun An, Stan Matwin, Zbigniew W. Raś, and Dominik Ślęzak (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 150–159.

[24] Maria Halkidi, Michalis Vazirgiannis, and Yannis Batistakis. 2000. Quality scheme assessment in the clustering process. In *European Conference on Principles of Data Mining and Knowledge Discovery.* Springer, 265–276.

[25] John A Hartigan. 1975. Clustering algorithms. (1975).

[26] B. F. A. Hompes, J. C. A. M. Buijs, Wil M. P. van der Aalst, P. M. Dixit, and J. Buurman. 2017. Detecting Changes in Process Behavior Using Comparative Case Clustering. In *Data-Driven Process Discovery and Analysis*, Paolo Ceravolo and Stefanie Rinderle-Ma (Eds.). Springer International Publishing, 54–75.

[27] Arnaud Hubaux, Andreas Classen, and Patrick Heymans. 2009. Formal Modelling of Feature Configuration Workflows. In *Proceedings of the 13th International Software Product Line Conference (SPLC '09).* Carnegie Mellon University, Pittsburgh, PA, USA, 221–230. http://dl.acm.org/citation.cfm?id=1753235.1753266

[28] A Hubaux, P b Heymans, P.-Y Schobbens, D Deridder, and E.K.a Abbasi. 2013. Supporting multiple perspectives in feature-based configuration. *SOSYM* 12, 3 (2013), 641–663. https://doi.org/10.1007/s10270-011-0220-1

[29] Lawrence J Hubert and Joel R Levin. 1976. A general statistical framework for assessing categorical clustering in free recall. *Psychological bulletin* 83, 6 (1976), 1072.

[30] A. K. Jain, M. N. Murty, and P. J. Flynn. 1999. Data Clustering: A Review. *ACM Comput. Surv.* 31, 3 (Sept. 1999), 264–323. https://doi.org/10.1145/331499.331504

[31] Ari Kobren, Nicholas Monath, Akshay Krishnamurthy, and Andrew McCallum. 2017. A Hierarchical Algorithm for Extreme Clustering. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '17).* ACM, New York, NY, USA, 255–264.

[32] Wojtek J Krzanowski and YT Lai. 1988. A criterion for determining the number of groups in a data set using sum-of-squares clustering. *Biometrics* (1988), 23–34.

[33] L Lebart, A Morineau, and M Piron. 2000. Statistique exploratoire multidimensionnelle, Dunod, Paris, France. (2000).

[34] Sander J. J. Leemans, Dirk Fahland, and Wil M. P. van der Aalst. 2014. Discovering Block-Structured Process Models from Incomplete Event Logs. In *Petri Nets (Lecture Notes in Computer Science)*, Vol. 8489. Springer, 91–110.

[35] Sander J. J. Leemans, Dirk Fahland, and Wil M. P. van der Aalst. 2015. Scalable Process Discovery with Guarantees. In *Enterprise, Business-Process and Information Systems Modeling*, Khaled Gaaloul, Rainer Schmidt, Selmin Nurcan, Sérgio Guerreiro, and Qin Ma (Eds.). Springer International Publishing, Cham, 85–101.

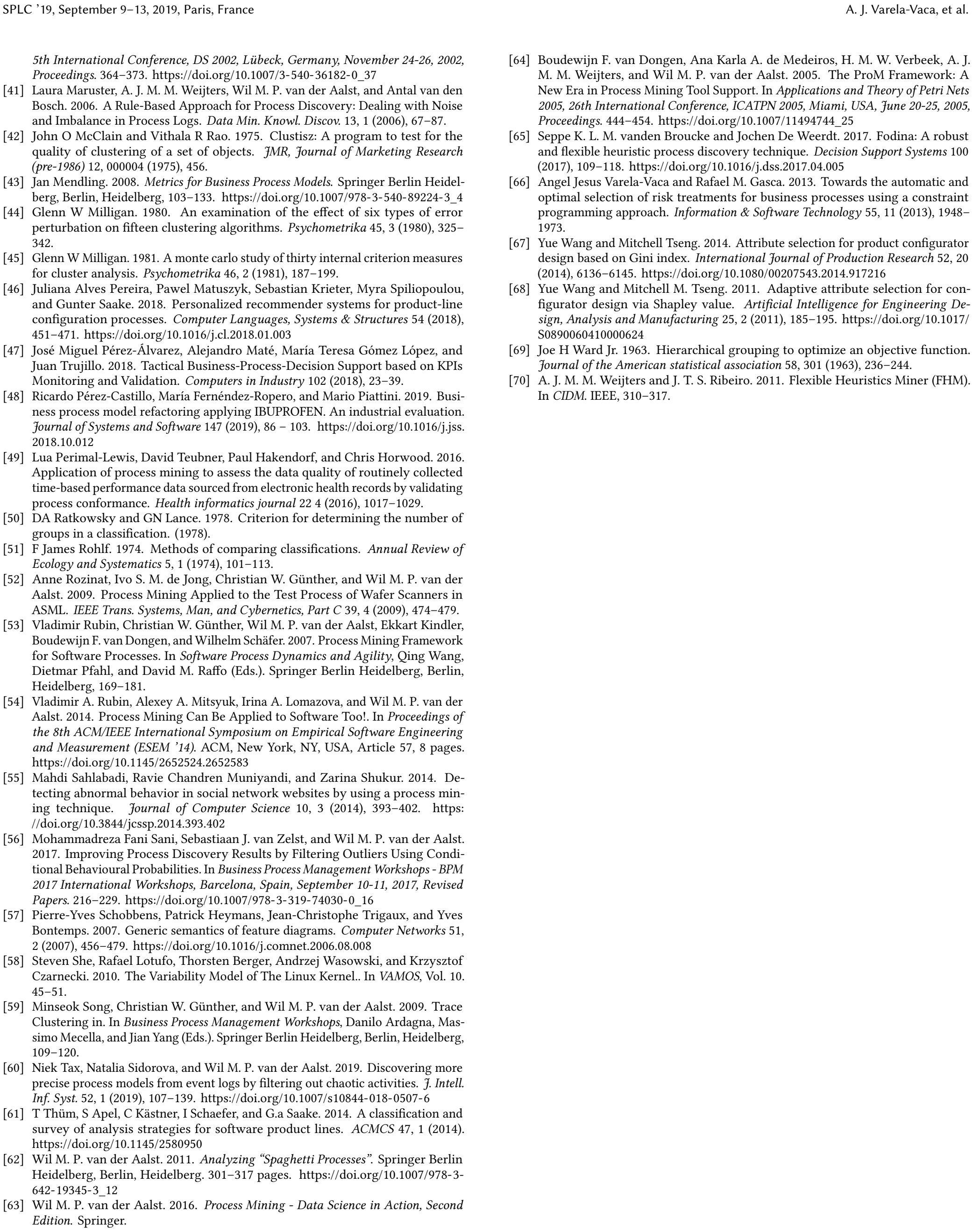
[36] Linh Thao Ly, Conrad Indiono, Jürgen Mangler, and Stefanie Rinderle-Ma. 2012. Data Transformation and Semantic Log Purging for Process Mining. In *CAiSE (Lecture Notes in Computer Science)*, Vol. 7328. Springer, 238–253.

[37] David J. C. MacKay. 2002. *Information Theory, Inference & Learning Algorithms.* Cambridge University Press, New York, NY, USA.

[38] R. S. Mans, M. H. Schonenberg, M. Song, W. M. P. van der Aalst, and P. J. M. Bakker. 2009. Application of Process Mining in Healthcare – A Case Study in a Dutch Hospital. In *Biomedical Engineering Systems and Technologies*, Ana Fred, Joaquim Filipe, and Hugo Gamboa (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 425–438.

[39] Laura Măruşter and Nick R. T. P. van Beest. 2009. Redesigning business processes: a methodology based on simulation and techniques. *Knowledge and Information Systems* 21, 3 (25 Jun 2009), 267. https://doi.org/10.1007/s10115-009-0224-0

[40] Laura Maruster, A. J. M. M. Weijters, Wil M. P. van der Aalst, and Antal van den Bosch. 2002. : Discovering Direct Successors in Process Logs. In *Discovery Science,*



5th International Conference, DS 2002, Lübeck, Germany, November 24-26, 2002, Proceedings. 364–373. https://doi.org/10.1007/3-540-36182-0_37

[41] Laura Maruster, A. J. M. M. Weijters, Wil M. P. van der Aalst, and Antal van den Bosch. 2006. A Rule-Based Approach for Process Discovery: Dealing with Noise and Imbalance in Process Logs. *Data Min. Knowl. Discov.* 13, 1 (2006), 67–87.

[42] John O McClain and Vithala R Rao. 1975. Clustisz: A program to test for the quality of clustering of a set of objects. *JMR, Journal of Marketing Research (pre-1986)* 12, 000004 (1975), 456.

[43] Jan Mendling. 2008. *Metrics for Business Process Models.* Springer Berlin Heidelberg, Berlin, Heidelberg, 103–133. https://doi.org/10.1007/978-3-540-89224-3_4

[44] Glenn W Milligan. 1980. An examination of the effect of six types of error perturbation on fifteen clustering algorithms. *Psychometrika* 45, 3 (1980), 325–342.

[45] Glenn W Milligan. 1981. A monte carlo study of thirty internal criterion measures for cluster analysis. *Psychometrika* 46, 2 (1981), 187–199.

[46] Juliana Alves Pereira, Pawel Matuszyk, Sebastian Krieter, Myra Spiliopoulou, and Gunter Saake. 2018. Personalized recommender systems for product-line configuration processes. *Computer Languages, Systems & Structures* 54 (2018), 451–471. https://doi.org/10.1016/j.cl.2018.01.003

[47] José Miguel Pérez-Álvarez, Alejandro Maté, María Teresa Gómez López, and Juan Trujillo. 2018. Tactical Business-Process-Decision Support based on KPIs Monitoring and Validation. *Computers in Industry* 102 (2018), 23–39.

[48] Ricardo Pérez-Castillo, María Fernéndez-Ropero, and Mario Piattini. 2019. Business process model refactoring applying IBUPROFEN. An industrial evaluation. *Journal of Systems and Software* 147 (2019), 86 – 103. https://doi.org/10.1016/j.jss.2018.10.012

[49] Lua Perimal-Lewis, David Teubner, Paul Hakendorf, and Chris Horwood. 2016. Application of process mining to assess the data quality of routinely collected time-based performance data sourced from electronic health records by validating process conformance. *Health informatics journal* 22 4 (2016), 1017–1029.

[50] DA Ratkowsky and GN Lance. 1978. Criterion for determining the number of groups in a classification. (1978).

[51] F James Rohlf. 1974. Methods of comparing classifications. *Annual Review of Ecology and Systematics* 5, 1 (1974), 101–113.

[52] Anne Rozinat, Ivo S. M. de Jong, Christian W. Günther, and Wil M. P. van der Aalst. 2009. Process Mining Applied to the Test Process of Wafer Scanners in ASML. *IEEE Trans. Systems, Man, and Cybernetics, Part C* 39, 4 (2009), 474–479.

[53] Vladimir Rubin, Christian W. Günther, Wil M. P. van der Aalst, Ekkart Kindler, Boudewijn F. van Dongen, and Wilhelm Schäfer. 2007. Process Mining Framework for Software Processes. In *Software Process Dynamics and Agility*, Qing Wang, Dietmar Pfahl, and David M. Raffo (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 169–181.

[54] Vladimir A. Rubin, Alexey A. Mitsyuk, Irina A. Lomazova, and Wil M. P. van der Aalst. 2014. Process Mining Can Be Applied to Software Too!. In *Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM '14).* ACM, New York, NY, USA, Article 57, 8 pages. https://doi.org/10.1145/2652524.2652583

[55] Mahdi Sahlabadi, Ravie Chandren Muniyandi, and Zarina Shukur. 2014. Detecting abnormal behavior in social network websites by using a process mining technique. *Journal of Computer Science* 10, 3 (2014), 393–402. https://doi.org/10.3844/jcssp.2014.393.402

[56] Mohammadreza Fani Sani, Sebastiaan J. van Zelst, and Wil M. P. van der Aalst. 2017. Improving Process Discovery Results by Filtering Outliers Using Conditional Behavioural Probabilities. In *Business Process Management Workshops - BPM 2017 International Workshops, Barcelona, Spain, September 10-11, 2017, Revised Papers.* 216–229. https://doi.org/10.1007/978-3-319-74030-0_16

[57] Pierre-Yves Schobbens, Patrick Heymans, Jean-Christophe Trigaux, and Yves Bontemps. 2007. Generic semantics of feature diagrams. *Computer Networks* 51, 2 (2007), 456–479. https://doi.org/10.1016/j.comnet.2006.08.008

[58] Steven She, Rafael Lotufo, Thorsten Berger, Andrzej Wasowski, and Krzysztof Czarnecki. 2010. The Variability Model of The Linux Kernel.. In *VAMOS*, Vol. 10. 45–51.

[59] Minseok Song, Christian W. Günther, and Wil M. P. van der Aalst. 2009. Trace Clustering in. In *Business Process Management Workshops*, Danilo Ardagna, Massimo Mecella, and Jian Yang (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 109–120.

[60] Niek Tax, Natalia Sidorova, and Wil M. P. van der Aalst. 2019. Discovering more precise process models from event logs by filtering out chaotic activities. *J. Intell. Inf. Syst.* 52, 1 (2019), 107–139. https://doi.org/10.1007/s10844-018-0507-6

[61] T Thüm, S Apel, C Kästner, I Schaefer, and G.a Saake. 2014. A classification and survey of analysis strategies for software product lines. *ACMCS* 47, 1 (2014). https://doi.org/10.1145/2580950

[62] Wil M. P. van der Aalst. 2011. *Analyzing "Spaghetti Processes".* Springer Berlin Heidelberg, Berlin, Heidelberg. 301–317 pages. https://doi.org/10.1007/978-3-642-19345-3_12

[63] Wil M. P. van der Aalst. 2016. *Process Mining - Data Science in Action, Second Edition.* Springer.

[64] Boudewijn F. van Dongen, Ana Karla A. de Medeiros, H. M. W. Verbeek, A. J. M. M. Weijters, and Wil M. P. van der Aalst. 2005. The ProM Framework: A New Era in Process Mining Tool Support. In *Applications and Theory of Petri Nets 2005, 26th International Conference, ICATPN 2005, Miami, USA, June 20-25, 2005, Proceedings.* 444–454. https://doi.org/10.1007/11494744_25

[65] Seppe K. L. M. vanden Broucke and Jochen De Weerdt. 2017. Fodina: A robust and flexible heuristic process discovery technique. *Decision Support Systems* 100 (2017), 109–118. https://doi.org/10.1016/j.dss.2017.04.005

[66] Angel Jesus Varela-Vaca and Rafael M. Gasca. 2013. Towards the automatic and optimal selection of risk treatments for business processes using a constraint programming approach. *Information & Software Technology* 55, 11 (2013), 1948–1973.

[67] Yue Wang and Mitchell Tseng. 2014. Attribute selection for product configurator design based on Gini index. *International Journal of Production Research* 52, 20 (2014), 6136–6145. https://doi.org/10.1080/00207543.2014.917216

[68] Yue Wang and Mitchell M. Tseng. 2011. Adaptive attribute selection for configurator design via Shapley value. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 25, 2 (2011), 185–195. https://doi.org/10.1017/S0890060410000624

[69] Joe H Ward Jr. 1963. Hierarchical grouping to optimize an objective function. *Journal of the American statistical association* 58, 301 (1963), 236–244.

[70] A. J. M. M. Weijters and J. T. S. Ribeiro. 2011. Flexible Heuristics Miner (FHM). In *CIDM*. IEEE, 310–317.

3.2.5 A NLP-oriented methodology to enhance event log quality

Published in Proceedings of Enterprise, Business-Process and Information Systems Modeling – 22nd International Conference, BPMDS 2021, and 26th International Conference, EMMSAD 2021, held at CAISE 2021, Melbourne, Vic, Australia, June 28-29, 2021, Vol. 421 of LNBIP, pp. 19–35. Springer, 2021.

- ***Authors**: Belén Ramos Gutiérrez, Ángel Jesús Varela Vaca, F. Javier Ortega, María Teresa Gómez López, Moe Thandar Wynn.*
- ***DOI**: `https://doi.org/10.1007/978-3-030-79186-5_2`.*
- ***Rating**: -*

A NLP-Oriented Methodology to Enhance Event Log Quality

Belén Ramos-Gutiérrez[1]( ), Ángel Jesús Varela-Vaca[1],
F. Javier Ortega[1], María Teresa Gómez-López[1],
and Moe Thandar Wynn[2]

[1] IDEA & ITALICA Research Groups, Universidad de Sevilla, Seville, Spain
{brgutierrez,ajvarela,javierortega,maytegomez}@us.es
[2] Queensland University of Technology (QUT), Brisbane, Australia
m.wynn@qut.edu.au
http://www.idea.us.es

Abstract. The quality of event logs is a crucial cornerstone for the feasibility of the application of later process mining techniques. The wide variety of data that can be included in an event log refer to information about the activity, such as what, who or where. In this paper, we focus on event logs that include textual information written in a natural language that contains exhaustive descriptions of activity executions. In this context, a pre-processing step is necessary since textual information is unstructured and it can contain inaccuracies that will provoke the impracticability of process mining techniques. For this reason, we propose a methodology that applies Natural Language Processing (NLP) to raw event log by relabelling activities. The approach let the customised description of the measurement and assessment of the event log quality depending on expert requirements. Additionally, it guides the selection of the most suitable NLP techniques for use depending on the event log. The methodology has been evaluated using a real-life event log that includes detailed textual descriptions to capture the management of incidents in the aircraft assembly process in aerospace manufacturing.

Keywords: Natural Language Processing · Event log quality · Process mining

1 Introduction

Event logs include the footprints generated by an organisation's information systems, being possible to store a wide variety of information [4,6] related to the tracked events, e.g., textual descriptions, timestamps or used resources. In general, event logs need to be adapted for a later (process mining) analysis, for instance, to discover processes. Thereby, the assessment of the quality of an event log [7] is the very first and crucial step for any subsequent analysis. The application of any process mining technique over incorrect or inaccurate event logs, e.g. process discovery, will produce incorrect or inaccurate process models [32].

Several authors have defined criteria to assess the data quality in general [9,21] and event log quality in particular [7,32], such as completeness, correctness, security and

A. Augusto et al. (Eds.): BPMDS 2021/EMMSAD 2021, LNBIP 421, pp. 19–35, 2021.
https://doi.org/10.1007/978-3-030-79186-5_2

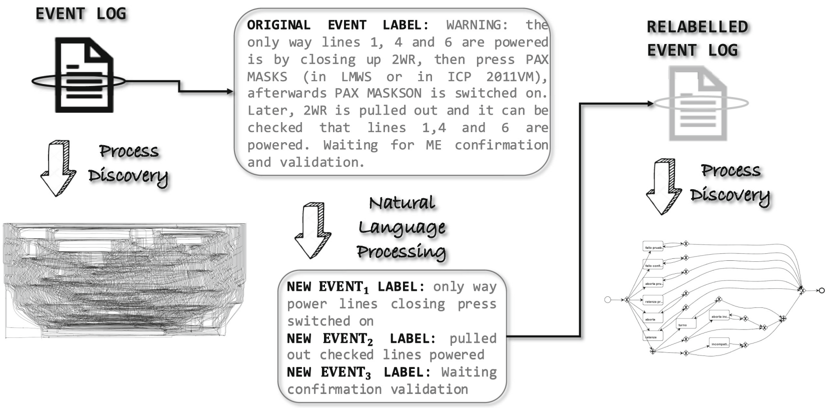

EVENT LOG
ORIGINAL EVENT LABEL: WARNING: the only way lines 1, 4 and 6 are powered is by closing up 2WR, then press PAX MASKS (in LMWS or in ICP 2011VM), afterwards PAX MASKSON is switched on. Later, 2WR is pulled out and it can be checked that lines 1,4 and 6 are powered. Waiting for ME confirmation and validation.
RELABELLED EVENT LOG
Process Discovery
Natural Language Processing
Process Discovery
NEW EVENT$_1$ LABEL: only way power lines closing press switched on
NEW EVENT$_2$ LABEL: pulled out checked lines powered
NEW EVENT$_3$ LABEL: Waiting confirmation validation

Fig. 1. Relabelling application example.

trustworthiness. The Process Mining manifesto [7] introduces the quality of an event log as a quality maturity level.

The imperfections than can produce a low event log quality might be improved analysing the activity labelling, timestamps, case identification, etc. In this paper, we focus on event logs that include some textual descriptions which detail what happened in various moments of process execution. We propose to first identify these activities and their inter-relations from textual constraint descriptions using NLP techniques and then relabelling activities in the event log to extract these details in an easy to handle way.

Figure 1 illustrates how the use of Natural Language Processing (NLP) techniques and the relabelling of activities in an event log, can improve the results of automated process discovery. For this reason, we not only adapt a general methodology to measure and assess the data quality of event logs [27], but also propose a decision-support system to assist in the selection of the most suitable NLP techniques according to the current quality level and the expected assessment. The research question is: *What are the most suitable NLP techniques to use for relabelling an event log in order to improve its quality?* This question does not have a simple answer, since it depends on the current quality of the event log and the dimension or dimensions that must be improved and how.

To answer this research question, it is necessary to define metrics to measure the event log quality and a mechanism to assess how good is this quality level in each context. This is well-known as the fitness-per-purpose [19], where the level of quality must be customised according to the needs, as for example: (1) determining the average length of the label of the activities; (2) the level of noise allowed; and (3) the usual number of activities per trace.

With this goal in mind, we propose a methodology, called LOADING-NLP, which assists in the decision-making for the application of NLP techniques over raw event logs for relabelling activities in accordance with the decision rules about data quality described by experts. A set of fitness-per-purpose metrics and dimensions are proposed to measure and assess the quality of an event log. Both, the metrics and dimensions can be customised, or extended for other examples. In this regard, the measurement and assessment can be adjusted and alternative NLP techniques can be applied. Our methodology also assists the user to select the most suitable NLP techniques. To validate the proposed methodology, we applied it to a real case study based on the management of the incidents produced during the aircraft assembly processes in aerospace manufacturing.

The rest of the paper is organised as follows: Sect. 2 includes the related work in the area. Section 3 introduces the methodology; the measurement and the assessment of the event log quality are discussed in Sect. 4. Section 5 reviews the NLP techniques that can be applied in this context and outlines to what extent they can affect the quality assessment. Section 6 presents the evaluation results and Sect. 7 concludes the paper.

2 Related Work

In order to understand the advantages that this proposal offers, it is necessary to know the level of maturity in the following areas.

Event Log Quality. The necessity to have data with suitable quality is crucial for any process and necessary for later analysis, such as process mining [10]. How to measure and assess the possibility of leveraging their quality is an important topic which has been a focus of study during the last decades. However, event logs appear in new contexts [30] and include features that make it necessary to define new metrics to measure, extend and adapt the dimensions (e.g., completeness, accuracy, simplicity) to the business process context [8,31].

Event Log Improvement. Once the data quality level can be assessed, various are the techniques that can be applied to improve the event log quality [26]. Some solutions are based on timestamp [12,13,15], case identification, and activity relabelling. Regarding the activity relabelling, the solution presented in [25] proposes to detect synonymous and polluted labels in event logs, but no techniques were proposed to improve the quality in accordance to the previous detection, and [24] uses a gamification approach to repair the labels. The types of improvements over event logs depend on the case and the later use of them [18].

Use of NLP in Business Process Management. Previous works have studied the extraction of declarative [2] and imperative business process models [3] from texts. In those works, the NLP techniques have been used in order to facilitate the automation of tasks that require a significant effort detecting patterns of relational order between the activities involved [1]. In addition, the detection of activities and their associated labels is crucial for further analysis and refactoring of the terms to enable an automatic analysis [17]. The text analysis in the context of the business process has also been focused on the detection of inconsistencies between the textual descriptions and the graphical representation [3], as a mechanism of misalignment detection.

Use of NLP to Improve Process Discovery Results. Some works have studied the preprocessing of the event data to improve the discovery task when using real-life logs that are written in natural language. In [23], is done by automatically detecting and classifying eight different semantic roles in event data. In [14], semantic-based techniques are applied to aggregate and normalise event log text information. Other types of analysis have been made to improve the labels of the activities in a process model by detecting erroneous ways of labelling activity that lead to ambiguity and inconsistency [22]. Contrary to our proposal, in all cases, these proposals start from event data in natural language with a very process-oriented construction and a simple and correct syntax.

To sum up, to the best of our knowledge, this is the first work where NLP techniques have been used to improve the event log quality according to a set of proposed metrics, guided by a decision-support system to ascertain the best techniques to apply depending on the event log and its quality level.

3 LOADING-NLP: Methodology for Assessing and Improving Event Log Quality with NLP

When the labels of the events within the log include natural language texts, it is necessary to analyse and treat them to become the log useful. The NLP techniques used to this aim will cause a direct impact on the different number of labels in the log, the number of events per trace, or the similarity between the labels.

Therefore, the best NLP techniques to use depends on the meaning of event log quality in each context, the event log quality before the application of the techniques, and how the experts want it to evolve after the application of the techniques. To support these three aspects, we propose the methodology presented in Fig. 2, described through a BPMN model.

The first step is related to the definition of when the event log is usable (cf., Determine the usability of the event log quality), according to the measurement and the assessment described by a set of decision rules about data quality. If the event log has sufficient quality, it can be used for a process mining analysis. However, if the event log is deemed unsuitable, the expert must determine the dimension or dimensions that must be adjusted and the required assessment (cf., Introduce dimensions and assessment to achieve). Using this information, we propose a decision-support system (cf., Infer NLP techniques to apply) that provides possible NLP techniques to use for improving the event log quality according to the described decision rules about data quality and the requirements of the experts. This process can be repeated until the resulting event log achieves the required level of quality.

4 Determine the Usability of the Event Log Quality

The question of when an event log has sufficient quality does not have a single answer. It will depend on the meaning of event log quality in each context. There are solutions as [27] that provide mechanisms to describe the decision rules related to the measurements and assessments adapted to each context and requirements. At first, we need to define

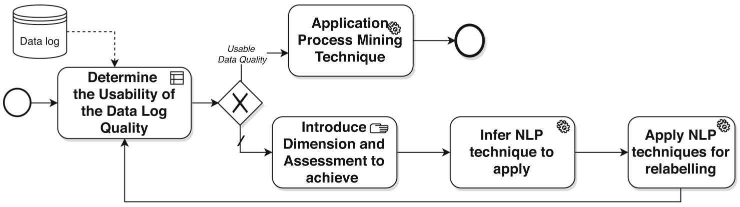

Data log
Determine the Usability of the Data Log Quality
Usable Data Quality
Application Process Mining Technique
Introduce Dimension and Assessment to achieve
Infer NLP technique to apply
Apply NLP techniques for relabelling

Fig. 2. Methodology for the application of NLP techniques.

the decision rules according to data quality using a set of metrics extracted from the event log. Thereby, it is necessary to define a set of metrics to evaluate the dimensions, as detailed in Subsects. 4.1 and 4.2. These measures enable us to perform an assessment according to the meaning of quality defined by the expert, as described in Subsect. 4.3.

4.1 Metrics to Measure the Quality of Event Logs

Based on [11], an ***event log*** is a set of traces that represent different instances of the same process. A ***trace*** is an ordered sequence of events that represents a process instance. Every trace is associated with a unique case identifier. The execution of an activity in a business process is represented as an ***event*** in an event log. Similarly, an event is the representation of the execution of an activity in a business process. Each event is associated with a case identifier, one timestamp and can also have many other contextual attributes. Usually, each event has associated at most one timestamp, which represents the start or the end of the execution of an activity.

To measure the dimensions, some metrics must be extracted from the event log [5, 11], as the mentioned in [8]. In our case, the used metrics are:

- **Number of traces.** Total number of traces in the log, and the trace j is represented by τ_j.
- **Number of events** (ε). Total number of events in the log, and the event i is represented by ε_i. It helps to know the size of the log.
- **Number of different labels.** Number of different labels that occur in every trace.
- **Number of unique labels.** Number of single (unique) labels that appear in the log.

4.2 Quality Dimensions for Event Logs

In general, the data quality dimensions describe the relevant aspects for a data set and typically consist of accuracy, completeness, consistency and uniqueness. However, we cannot guarantee that an event log with a high level of quality in those dimensions will produce valuable business processes. For this reason, other dimensions are included to assess the event log quality in process mining, as was defined in [31]. Those dimensions can be affected by the application of NLP techniques. Based on them, we propose the following dimensions albeit others can also be used together with our methodology:

$$m_{Uniqueness} = \left(\frac{Number\ of\ Unique\ Labels}{Number\ of\ Events}\right) \quad m_{Complexity} = \left(\frac{Avg.\ Number\ of\ Events}{Number\ of\ Traces}\right)$$

$$m_{Relevancy} = \left(\frac{Number\ of\ Different\ Labels}{Number\ of\ Events}\right) \quad m_{Consistency} = \sum_{i=1}^{\varepsilon}\left(\frac{|l(\varepsilon_i) - \overline{l(\varepsilon)}|}{Number\ of\ Events}\right)$$

Uniqueness. If every label in an event log is the same, the discovered process will only include one activity. However, if each label is unique in the traces, the discovered process will have one branch per trace, thus not very useful. The uniqueness dimension, that we propose in a range between $[0..1]$, measures the percentage of single labels regarding the total number of labels. When the values are in the extremes (i.e., 0 or 1), it implies that the process may be too simple (low uniqueness) or too complex (high uniqueness).

Consistency. When the labels of activities are dissimilar, especially for textual formats, they can imply that the descriptions have different granularity. For this reason, the measurement of consistency that we propose is based on the average length[1] of strings in these textual descriptions, and the mean of the distance to the average. Therefore, this dimension is bounded by the length of the longest string.

Relevancy. The relevancy of each label depends on the number of times that it occurs. It is important to analyse the number of different labels according to the total number. It is related to the uniqueness, but it is not exactly the same. The dimension is bound between $[0..1]$.

Complexity. There are several metrics that can represent the complexity of an event log [26], such as the average of events per trace. We propose to measure the complexity by the mean of the number of events per trace. A higher mean implies a higher concentration of events per trace, therefore representing a more complex process.

4.3 Customising the Measurement and Assessment of Event Log Quality

As commented previously, the data quality is an aspect highly related to the later use of the data, hence it must be customised according to the necessities. Following the DMN4DQ proposed in [27,28], DMN (Decision Model and Notation) [20] can be used for facilitating the description of data quality divided into measurement and assessment rules. DMN is a declarative language proposed by OMG to describe decision rules applied to a tuple of input data to obtain a tuple of outputs according to the evaluation of a set of conditions described in FEEL. A DMN table is composed of rows that describe a decision rule as an if-then condition so that, if it is satisfied, the output is returned. Also, DMN permits a hierarchical structure where the output of a DMN table can be the input of another. Using the methodology DMN4DQ, we propose to split up the measurement and assessment in two different levels for each involved dimension. Additionally, the final assessment is obtained by aggregating the assessment of every dimension, as described in Fig. 3.

[1] We use $l()$ function to define the length of a label description.

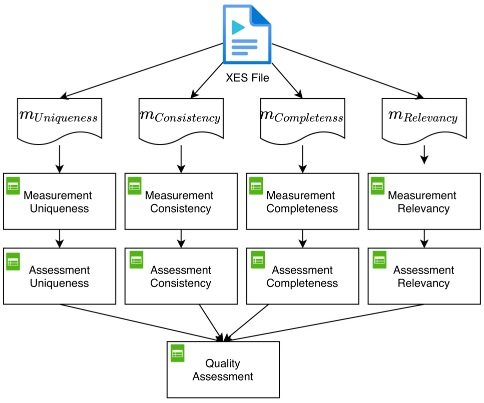


Fig. 3. DMN for describing the Quality Assessment.

DMN tables have various types of columns (`orders`, `inputs`, and `outputs`). The first column establishes the `order` by assigning an index to each row, and includes the hit policy to determine how to act when more than one is satisfied, (cf., *F* to describe that the evaluation of the condition is in order). Each `input` column represents the input variables that are evaluated the condition of the row. An example of DMN tables for the measurement of each dimension is detailed in Tables 1a, 1b, 1c and 1d, that illustrate the four dimensions proposed in this paper. Each dimension is described by a set of domains of the metric values, where the measurement of the metrics is the output value of the table. For example, the consistency dimension, in this case, describes that if the average of the numbers of characters is greater than 30, the measurement of the consistency will be *Very low*. For the measurement of each dimension, only one metric is required as input. In each row of the input column, the conditions are described in FEEL, for instance, the first row in Table 1b establishes that the valid range for the metric $m_{Uniqueness}$ is between 0 and 1. We use the metrics defined in the previous subsection as the inputs. The `outputs` represent the obtained values depending on the condition satisfied, for instance, if the input for $m_{Uniqueness}$ is 0.5 the outputs is *High*. The values and conditions established in Table 1 have been adjusted according to the know-how of experts for the use case at hand.

We should bear in mind that the measurement of a metric does not represent whether the metric is good or not, this is why a later assessment is necessary. Table 2 includes a proposal for the assessment of each dimension, for example, both a *Low* and *Very Low* number of events will imply an *Excellent* assessment in the event log according to the Complexity metric. As previously commented, this assessment has been defined based on the experts' knowledge of the event logs but other assessments can be accommodated. The assessment of each dimension needs to be aggregated to determine a global one. A possible set of decision rules for the aggregation of the assessment of the four dimensions is described in Table 3, albeit another combination can be applied according to the necessity of the organisation.

Table 1. Decision tables for measuring each dimension.

(a) Measurement of Uniqueness Dimension

	Input	$Output_U$
F	$m_{Uniqueness}$	
1	[0, 0.1]	Very Low
2	(0.1, 0.2]	Low
3	(0.2, 0.4]	Medium
4	(0.4, 0.6]	High
5	(0.6, 1]	Very High

(b) Measurement of Consistency Dimension

	Input	$Output_{Cs}$
F	$m_{Consistency}$	
1	[0, 5]	Very High
2	(6, 14]	High
3	(14, 20]	Medium
4	(20, 30]	Low
5	(30, ∞)	Very Low

(c) Measurement of Relevancy Dimension

	Input	$Output_R$
F	$m_{Relevancy}$	
1	[0, 0.1]	Very High
2	(0.1, 0.2]	High
3	(0.2, 0.4]	Medium
4	(0.4, 0.6]	Low
5	(0.6, 1]	Very Low

(d) Measurement of Complexity Dimension

	Input	$Output_{Cx}$
F	$m_{Complexity}$	
1	[0, 4]	Very Low
2	(4, 6]	Low
3	(7, 10]	Medium
4	(11, 15]	High
5	(16, ∞)	Very High

Table 2. Decision tables for the assessment of each dimension.

(a) Assessment of Uniqueness Dimension

	Input	$Assess_U$
F	$Output_U$	
1	Very Low	Fair
2	Low ∨ Medium	Excellent
3	High	Poor
4	Very High	Very Poor

(b) Assessment of Consistency Dimension

	Input	$Assess_{Cs}$
F	$Output_{Cs}$	
1	Very Low	Very Poor
2	Low	Poor
3	Medium	Fair
4	High	Good
5	Very High	Excellent

(c) Assessment of Relevancy Dimension

	Input	$Assess_R$
F	$Output_R$	
1	Very Low ∨ Very High	Very Poor
2	High ∨ Medium	Fair
3	Low	Poor

(d) Assessment of Complexity Dimension

	Input	$Assess_{Cx}$
F	$Output_{Cx}$	
1	Very Low ∨ Low	Excellent
2	Medium	Good
3	High	Poor
4	Very High	Very Poor

Table 3 is designed in such a way that, when at least 3 assessment values for the dimensions is qualified as Excellent, and the remaining one is qualified as Good or Fair, the quality outcome of the log is Excellent. Similarly, when we have three dimensions qualified as Excellent or Good and one as Fair or Poor, the quality outcome of the log will be Good. On the other hand, when we find out two dimensions as Excellent or Good and other two as Fair or Poor, the quality outcome of the log will be Fair, while,

Table 3. Aggregation of the Dimensions for the Quality Assessment

	Inputs				Output
F	$Assess_U$	$Assess_{Cs}$	$Assess_R$	$Asses_{Cx}$	$Quality_{assessment}$
1	Excellent	Excellent	Excellent	Excellent ∨ Good ∨ Fair	Excellent
2	Excellent	Excellent	Good ∨ Fair	Excellent	Excellent
3	Excellent	Good ∨ Fair	Excellent	Excellent	Excellent
4	Good ∨ Fair	Excellent	Excellent	Excellent	Excellent
5	Good ∨ Excellent	Good ∨ Excellent	Good ∨ Excellent	Good ∨ Excellent	Good
6	Poor ∨ Fair	Good ∨ Excellent	Good ∨ Excellent	Good ∨ Excellent	Good
7	Good ∨ Excellent	Poor ∨ Fair	Good ∨ Excellent	Good ∨ Excellent	Good
8	Good ∨ Excellent	Good ∨ Excellent	Poor ∨ Fair	Good ∨ Excellent	Good
9	Good ∨ Excellent	Good ∨ Excellent	Good ∨ Excellent	Poor ∨ Fair	Good
10	Poor ∨ Fair	Poor ∨ Fair	Good ∨ Excellent	Good ∨ Excellent	Fair
11	Good ∨ Excellent	Poor ∨ Fair	Poor ∨ Fair	Good ∨ Excellent	Fair
12	Good ∨ Excellent	Good ∨ Excellent	Poor ∨ Fair	Poor ∨ Fair	Fair
13	Poor ∨ Fair	Good ∨ Excellent	Good ∨ Excellent	Poor ∨ Fair	Fair
14	Very Poor	Very Poor	Very Poor	Very Poor	Very Poor
15	Very Poor	Very Poor	Very Poor ∨ Poor ∨ Fair	Very Poor ∨ Poor ∨ Fair	Very Poor
16	Very Poor ∨ Poor	Very Poor	Very Poor	Very Poor ∨ Poor ∨ Fair	Very Poor
17	Very Poor ∨ Poor ∨ Fair	Very Poor ∨ Poor ∨ Fair	Very Poor	Very Poor	Very Poor
18	-	-	-	-	Poor

when, at least, two dimensions are qualified as Very Poor, the quality outcome of the log will be Very Poor. In any other case, the quality outcome of the log will be Poor.

5 Improving Event Log Quality: NLP Techniques for Relabelling Activities

Our proposal aims to guide selection the NLP techniques for the relabelling of activities to extract the most meaningful and representative words for each process activity, but first, we need to introduce some NLP techniques.

For the sake of clarity, we take the following description of an incident as an example to show the effects of each NLP technique proposed in the paper: *"WARNING: the only way lines 1, 4 and 6 are powered is by closing up 2WR, then press PAX MASKS (in LMWS or in ICP 2011VM), afterwards PAX MASKS ON is switched on. Later, 2WR is pulled out and it can be checked that lines 1, 4 and 6 are powered. Waiting for ME confirmation and validation."*

The NLP techniques that are being proposed to be applied are described below:

Sentence Detection. This technique splits the text into its main components (i.e., sentences) to make it easier for the next steps to extract rich information from them. For our example, the application of this technique provides the next output:

- *Sentence 1.* WARNING: the only way lines 1, 4 and 6 are powered is by closing up 2WR, then press PAX MASKS (in LMWS or in ICP 2011VM), afterwards PAX MASKS ON is switched on
- *Sentence 2.* Later, 2WR is pulled out and it can be checked that lines 1, 4 and 6 are powered.
- *Sentence 3.* Waiting for ME confirmation and validation.

Each sentence is shorter and contains fewer verbs (actions) than the original, so they should be easier to analyse afterwards.

Part-Of-Speech (POS) Tagging. It consists of determining the grammatical function of each word in a text (i.e., noun, verb, adjective, preposition, pronoun, etc.), choosing for each word its corresponding class from a set of predefined tags[2]. The result of a POS-tagging on our example, selecting those words that are tagged as "NOUN", "VERB" or "ADJ" (adjective), therefore excluding the rest:

- *Sentence 1.* Only way power lines closing press switched on
- *Sentence 2.* Pulled out checked lines powered
- *Sentence 3.* Waiting confirmation validation

With this technique, we can keep those words that we consider relevant according to their grammatical category.

Lemmatisation. This technique normalises or substitutes the inflected forms by its lemma. In this way, it is easier to compare texts or even to group together different inflexions of the same lexeme. Lemmatisation can provide a more normalised text which can be better suited for relabelling. The results of the lemmatisation of our example are:

- *Sentence 1.* The only way that power the line 1 , 4 and 6 to be close 2WR, then press PAX MASKS (in LMWS or in ICP 2011VM), afterwards this switch on in PAX MASKS
- *Sentence 2.* Later pull out 2WR and check that the line 1 , 4 and 6 now to be power
- *Sentence 3.* Wait ME confirm valid.

Dependency Parsing. It determines the syntactic relationships between the words in a sentence by obtaining a *dependency tree* which provides information about the root verb of the sentence, its subject, the different objects and complements that it could contain. These are the roots of each sentence in our example detected by a dependency parser: *Sentence 1:* press; *Sentence 2:* pull; *Sentence 3:* Waiting. In this case, the dependency parser is used to extract the main verb of each sentence, so we could identify the action that characterises the corresponding activity.

Acronyms Detection. We have implemented a simple rule-based acronym detection that retrieves those words that are written in upper-cases and their lower-case form do not exist in the target language. Next, we show the acronyms detected by our approach

[2] All the tag sets used in this work come from the community open project called *Universal Dependencies* (https://universaldependencies.org/).

Table 4. Expected impact of NLP techniques on the log quality dimensions.

Dimensions	Sentence detection	POS tagging	Lemmat.	Dependency parsing	Acronym detection
$m_{Uniqueness}$	↓	↓	↓	↓	↑
$m_{Consistency}$	↓	↓	↓	↓	-
$m_{Relevancy}$	↓	↓	↓	↓	↑
$m_{Complexity}$	↑	-	-	-	-

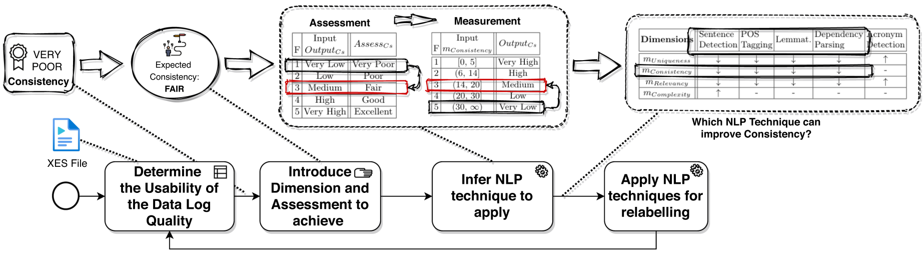
in the example: *2WR, LMWS, 2011VM*. Depending on the context, these acronyms could be useful for detecting relevant entities in the domain.

It is important to bear in mind that the application of some of these techniques to certain texts may lead to the generation of void labels. When this happens, those events with an empty label are not included in the new log. On the other hand, when the NLP technique splits a label into several new ones (e.g., the acronyms 2WR, LMWS, 2011VM), one new event is generated for each new label, but maintaining the other attributes from the original event (e.g., the timestamp attribute).

5.1 Decision-Making for the Application of NLP Techniques

Our proposal for relabelling an event log consists of the application of the aforementioned NLP techniques to the incident descriptions, filtering out or editing them, to produce new simplified texts that are used to replace the original activity labels.

As previously commented in Sect. 5, bear in mind that the application of NLP techniques may generate or delete events according to the new labels generated. We can observe in Table 4 how the detection of sentences produces shorter texts, reducing their diversity but increasing the number of events (we will have one event per sentence, in the same order they appear within the description). Therefore, it can decrease all the proposed dimensions, except the $m_{Complexity}$ which will be increased. Concerning detection of acronyms, it can increase the $m_{Uniqueness}$ and the $m_{Relevancy}$, while the $m_{Complexity}$ and the $m_{Consistency}$ may not be noticeably affected due to the usually short length of the acronyms. The rest of NLP techniques (POS Tagging, lemmatisation, and dependency parsing) are applied to filter out irrelevant elements of the texts, keeping the important ones, and also to unify different inflexions of words into a common meaningful form. Therefore, their effects on the dimensions would be fairly similar. The $m_{Uniqueness}$ and the $m_{Relevancy}$ can be decreased since the normalisation of texts achieved by these techniques should decrease the number of unique events as well as the number of different events. $m_{Consistency}$ can be decreased as well because the length of the texts will be reduced and so will be the difference in length between them. $m_{Complexity}$ is not significantly affected because the number of traces and the number of events stays almost the same.


VERY POOR Consistency
Expected Consistency: FAIR
Assessment
Measurement
XES File
Determine the Usability of the Data Log Quality
Introduce Dimension and Assessment to achieve
Infer NLP technique to apply
Apply NLP techniques for relabelling
Which NLP Technique can improve Consistency?

F	Input $Output_{Cs}$	$Assess_{Cs}$
1	Very Low	Very Poor
2	Low	Poor
3	Medium	Fair
4	High	Good
5	Very High	Excellent

F	Input $m_{Consistency}$	$Output_{Cs}$
1	[0, 5]	Very High
2	(6, 14]	High
3	(14, 20]	Medium
4	(20, 30]	Low
5	(30, ∞)	Very Low

Dimensions	Sentence Detection	POS Tagging	Lemmat.	Dependency Parsing	Acronym Detection
$m_{Uniqueness}$	↓	↓	↓	↓	↑
$m_{Consistency}$	↓	↓	↓	↓	-
$m_{Relevancy}$	↓	↓	↓	↓	↑
$m_{Complexity}$	↑	-	-	-	-

Fig. 4. Inferring the NLP technique to apply.

5.2 Inferring NLP Techniques to Improve Quality Dimensions

The relabelling of event logs through the application of NLP techniques and the definition of the dimensions and metrics discussed in previous sections provide a useful tool that guides us in the selection of the most suitable set of techniques to be applied achieving a certain assessment of quality.

Let's go back to the example proposed in Sect. 4.3. Let assume that there is an event log with a Consistency assessment ($Assess_{Cs}$) equal to $Very\ Poor$ and we want to improve it to $Fair$ as shown in Fig. 4. According to Table 2(d), we will need to take our event log from $Very\ Low$ to, at least, $Medium$ in terms of the measurement of the Consistency ($m_{Consistency}$). Then, looking at Table 1(d), it is necessary to reduce the value of the $m_{Consistency}$. Finally, according to Table 4, we can find out the NLP techniques that decrease the $m_{Consistency}$, and hence can be applied to our event log to achieve our objective. In this case they are *Sentence Detection*, *POS tagging*, *Lemmatisation*, and *Dependency parsing*.

In summary, given an event log with an assessment of quality and the dimensions to be improved, our proposal can help us to choose and apply the proper set of NLP techniques to achieve our objective.

6 Evaluation of the Proposal

For the evaluation, we use an event log (hereinafter Log_{desc}[3]) that represents the description of the evolution of the incidents in the aircraft assembly process which was presented in [29]. For instance, the following text represents a real incident description: *"When reading, the F1 error appears, wiring is verified according to FAQ and there is no continuity in any pinning in sections from 1509VC (pin 16) to 1599VC (pin 12). It is also appreciated that the colour coding concerning the plane (P1) does not correspond. Between FLKC1 and 250VC the wiring is correct"*. It can be easily observed that in this description of an incident, 3 sub-incidents are recorded: *(i) the F1 error appears*, *(ii) there is no continuity in any pinning from 16 to 12*, and; *(iii) the colour coding is incorrect*. For these reasons, textual descriptions can be useful to improve event logs

[3] Characteristics of the event log: 11.342 cases, 114.473 events, number of different labels 78.012, and 10.811 variants.

Table 5. Metrics of the event log used in the example, tagged as Log_{desc}.

Description	Total
Total number of textual descriptions	4,022
Total number of words	72,435
Out-Of-Vocabulary (OOV) words	10,832
Number of descriptions with OOVs	3,468
Grammatical and syntactic errors	1,642
Number of descriptions with errors	1,233

and discovered processes quality, but they must also be carefully processed to obtain relevant results.

The NLP is carried out with the aid of *spaCy* [16], a state-of-the-art Python library with pre-trained language models. Specifically, we have used for this work the largest Spanish pre-trained spaCy model, "es_core_news_lg"[4].

In order to illustrate the complexity of the problem, we show some metrics about Log_{desc} used for our evaluation in Table 5. We can see that 14.95% of the terms in the log are Out-Of-Vocabulary (OOV), which means that those words do not belong to the language at hand (they do not appear in the language model). In this point, domain-dependent enhancements could have been applied to the log, but the evaluation of our technique could have been biased by the nature of the domain, so we decided to keep the log as is. This complicates proper processing of the texts since 86.22% of the descriptions contain such terms. An additional difficulty from the point of view of NLP is the length of the descriptions, since too short texts may be insufficiently informative, and too long texts may be noisy for the task at hand. In this sense, Log_{desc} contains 110 descriptions with 3 or fewer words and 212 descriptions with more than 50 words. Finally, the event log has also been analysed using a grammar and spell checking tool[5], detecting a total of 1,642 errors (apart from the errors provoked by OOV words), which affects the $30, 65\%$ of descriptions.

In order to evaluate the proposed steps, they have been applied to Log_{desc}, performing different sets of NLP techniques. At first, we have applied five techniques, thus, five new event logs have been created: Log_{acro}, the acronym detection is used to keep only these keywords; Log_{dep}, the dependency analysis is applied to keep only the root word of each description; Log_{lemma} contains the lemmatisation of the words; Log_{pos} applying POS tagging and keeping only those words tagged as "NOUN", "VERB" or "ADJ" (nouns, verbs and adjectives); and Log_{sent}, the sentence detection is used to split up each description into its constituent sentences. Second, we propose several pipelines of NLP techniques for the improvement of the event log quality. The Log_{sent} is used as the first step for all the proposed pipelines: Log_{sent_dep}, we simplify sentences only maintaining the root word; $Log_{sent\_dep\_lemma}$, we apply a lemmatisation to the root word previously obtained; $Log_{sent\_dep\_lemma\_acro}$, in addition to the lemmatised

[4] https://github.com/explosion/spacy-models/releases//tag/es_core_news_lg-3.0.0.

[5] Language-Tool: https://github.com/languagetool-org/languagetool.

form of the root words of each description, we also keep the acronyms within them; for Log_{sent_pos}, the POS tagging is applied to keep the nouns, verbs and adjectives of each sentence; $Log_{sent\_pos\_acros}$ the acronyms detected are added to the previous log; with $Log_{sent\_pos\_lemma}$ we keep the lemmatised forms of the words within Log_{sent_pos}; finally, $Log_{sent\_pos\_lemma\_acros}$ adds the acronyms to the previous log.

We have obtained the results of the quality assessment for each log previously described as shown in Table 6. The results show the value for each metric and the final quality reached. The results presented support how the application of the NLP techniques affect the measurements as estimated in Table 4. However, there exist dimensions, such as Relevancy, whose relation among the metric and the assessment is not lineal, e.g., when an NLP is applied to increase the relevancy metric, the assessment can become *Very Poor* instead of *Fair*.

Finally, the implementation of our framework used in this evaluation is available on a website [6].

Table 6. Dimensions values for the event logs applying NLP techniques.

Event log	Complexity	Uniqueness	Relevancy	Consistency	Quality assessment
Log_{desc}	6.734 (Excellent)	0.621 (Very Poor)	0.734 (Very Poor)	58.916 (Very Poor)	Poor
Log_{acro}	2.708 (Excellent)	0.229 (Excellent)	0.399 (Fair)	3.818 (Excellent)	Good
Log_{dep}	6.384 (Excellent)	0.177 (Excellent)	0.291 (Fair)	4.373 (Excellent)	Good
Log_{lemma}	6.707 (Excellent)	0.474 (Poor)	0.620 (Very Poor)	36.091 (Very Poor)	Poor
Log_{pos}	6.697 (Excellent)	0.439 (Poor)	0.586 (Poor)	26.671 (Poor)	Poor
Log_{sent}	8.886 (Good)	0.500 (Poor)	0.643 (Very Poor)	30.104 (Very Poor)	Very Poor
Log_{sent_dep}	8.227 (Good)	0.090 (Fair)	0.183 (Fair)	1.831 (Excellent)	Fair
$Log_{sent\_dep\_lemma}$	8.227 (Good)	0.057 (Fair)	0.126 (Fair)	1.733 (Excellent)	Fair
$Log_{sent\_dep\_lemma\_acro}$	8.325 (Good)	0.104 (Excellent)	0.190 (Fair)	2.560 (Excellent)	Good
Log_{sent_pos}	8.533 (Good)	0.413 (Poor)	0.566 (Poor)	18.924 (Fair)	Poor
$Log_{sent\_pos\_acro}$	8.564 (Good)	0.422 (Poor)	0.576 (Poor)	19.415 (Fair)	Poor
$Log_{sent\_pos\_lemma}$	8.533 (Good)	0.399 (Excellent)	0.552 (Poor)	18.563 (Fair)	Fair
$Log_{sent\_pos\_lemma\_acro}$	8.564 (Good)	0.410 (Poor)	0.562 (Poor)	19.051 (Fair)	Poor

7 Conclusions and Future Work

The preparation of an event log by carefully paying attention to its quality is crucial for the later (process mining) analysis. One of the difficulties is to ascertain when the quality is sufficient for a specific purpose, and which techniques to use to improve the quality. In this paper, we focus on the improvement of the event log quality by using NLP techniques that affect both the measurement and assessment. We propose: (1) a set of metrics to measure the quality of an event log; (2) a mechanism to describe both, data and process rules, for assessing the event log quality; and, (3) a guide for selecting the more proper NLP techniques to apply. The viability of the proposal has been demonstrated by an implementation applied to a real event log from an industrial context. As an extension of the paper, we plan to analyse how the quality of the event

[6] http://www.idea.us.es/loading-nlp.

log can be aligned with the quality of the process discovered. In addition, we will extend the number of metrics and mechanisms to improve the quality level of the event log, not only contextualised to the data textual analysis.

Acknowledgement. Projects (RTI2018-094283-B-C33, RTI2018-098062-A-I00), funded by: FEDER/Ministry of Science and Innovation - State Research, and the Junta de Andalucía via the COPERNICA (P20_01224) project.

References

1. van der Aa, H., Carmona, J., Leopold, H., Mendling, J., Padró, L.: Challenges and opportunities of applying natural language processing in business process management. In: Proceedings of the 27th COLING 2018, Santa Fe, New Mexico, USA, 20-26 August 2018, pp. 2791–2801 (2018)
2. van der Aa, H., Di Ciccio, C., Leopold, H., Reijers, H.A.: Extracting declarative process models from natural language. In: Giorgini, P., Weber, B. (eds.) CAiSE 2019. LNCS, vol. 11483, pp. 365–382. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-21290-2_23
3. van der Aa, H., Leopold, H., Reijers, H.A.: Comparing textual descriptions to process models - the automatic detection of inconsistencies. Inf. Syst. **64**, 447–460 (2017)
4. Van der Aalst, W.: Process Mining Discovery Conformance and Enhancement of Business Processes. Springer-Verlag, Heidelberg (2011). https://doi.org/10.1007/978-3-642-19345-3
5. van der Aalst, W.: Extracting event data from databases to unleash process mining. In: BPM - Driving Innovation in a Digital World, pp. 105–128 (2015)
6. van der Aalst, W.: Process Mining - Data Science in Action, 2nd edn. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49851-4_1
7. van der Aalst, W., et al.: Process mining manifesto. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) BPM 2011. LNBIP, vol. 99, pp. 169–194. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-28108-2_19
8. Andrews, R., van Dun, C.G.J., Wynn, M.T., Kratsch, W., Röglinger, M., ter Hofstede, A.H.M.: Quality-informed semi-automated event log generation for process mining. Decis. Support Syst. **132**, 113265 (2020)
9. Batini, C.: Data quality assessment. In: Liu, L., Özsu, M.T. (eds.) Encyclopedia of Database Systems, 2nd edn. Springer, New York (2018). https://doi.org/10.1007/978-1-4614-8265-9
10. Bose, R.J.C., Mans, R.S., van der Aalst, W.M.: Wanna improve process mining results? In: 2013 IEEE Symposium on Computational Intelligence and Data Mining (CIDM), pp. 127–134. IEEE (2013)
11. Chapela-Campa, D., Mucientes, M., Lama, M.: Discovering infrequent behavioral patterns in process models. In: Carmona, J., Engels, G., Kumar, A. (eds.) BPM 2017. LNCS, vol. 10445, pp. 324–340. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-65000-5_19
12. Conforti, R., La Rosa, M., ter Hofstede, A.: Timestamp repair for business process event logs (2018). http://hdl.handle.net/11343/209011
13. Denisov, V., Fahland, D., van der Aalst, W.M.P.: Repairing event logs with missing events to support performance analysis of systems with shared resources. In: Janicki, R., Sidorova, N., Chatain, T. (eds.) PETRI NETS 2020. LNCS, vol. 12152, pp. 239–259. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-51831-8_12
14. Deokar, A.V., Tao, J.: Semantics-based event log aggregation for process mining and analytics. Inf. Syst. Front. **17**(6), 1209–1226 (2015). https://doi.org/10.1007/s10796-015-9563-4

15. Fischer, D.A., Goel, K., Andrews, R., van Dun, C.G.J., Wynn, M.T., Röglinger, M.: Enhancing event log quality: detecting and quantifying timestamp imperfections. In: Fahland, D., Ghidini, C., Becker, J., Dumas, M. (eds.) BPM 2020. LNCS, vol. 12168, pp. 309–326. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58666-9_18
16. Honnibal, M., Montani, I., Van Landeghem, S., Boyd, A.: spaCy: Industrial-strength Natural Language Processing in Python (2020). https://doi.org/10.5281/zenodo.1212303
17. Leopold, H., Pittke, F., Mendling, J.: Ensuring the canonicity of process models. Data Knowl. Eng. **111**, 22–38 (2017)
18. Martin, N., Martinez-Millana, A., Valdivieso, B., Fernández-Llatas, C.: Interactive data cleaning for process mining: a case study of an outpatient clinic's appointment system. In: Di Francescomarino, C., Dijkman, R., Zdun, U. (eds.) BPM 2019. LNBIP, vol. 362, pp. 532–544. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-37453-2_43
19. Mocnik, F.B., Fan, H., Zipf, A.: Data quality and fitness for purpose (2017). https://doi.org/10.13140/RG.2.2.13387.18726
20. OMG: Decision Model and Notation (DMN), Version 1.2 (2019). https://www.omg.org/spec/DMN
21. Otto, B., Lee, Y.W., Caballero, I.: Information and data quality in networked business. Electron. Mark. **21**(2), 79–81 (2011). https://doi.org/10.1007/s12525-011-0062-2
22. Pittke, F., Leopold, H., Mendling, J.: When language meets language: anti patterns resulting from mixing natural and modeling language. In: Fournier, F., Mendling, J. (eds.) BPM 2014. LNBIP, vol. 202, pp. 118–129. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-15895-2_11
23. Rebmann, A., van der Aalst, H.: Extracting semantic process information from the natural language in event logs. CoRR abs/2103.11761 (2021)
24. Sadeghianasl, S., ter Hofstede, A.H.M., Suriadi, S., Turkay, S.: Collaborative and interactive detection and repair of activity labels in process event logs. In: 2nd ICPM, pp. 41–48 (2020)
25. Sadeghianasl, S., ter Hofstede, A.H.M., Wynn, M.T., Suriadi, S.: A contextual approach to detecting synonymous and polluted activity labels in process event logs. In: Panetto, H., Debruyne, C., Hepp, M., Lewis, D., Ardagna, C.A., Meersman, R. (eds.) OTM 2019. LNCS, vol. 11877, pp. 76–94. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-33246-4_5
26. Suriadi, S., Andrews, R., ter Hofstede, A., Wynn, M.: Event log imperfection patterns for process mining: towards a systematic approach to cleaning event logs. Inf. Syst. **64**, 132–150 (2017)
27. Valencia-Parra, A., Parody, L., Varela-Vaca, A.J., Caballero, I., Gómez-López, M.T.: DMN4DQ: when data quality meets DMN. Decis. Support Syst. **141**, 113450 (2020)
28. Valencia-Parra, Á., Parody, L., Varela-Vaca, Á.J., Caballero, I., Gómez-López, M.T.: DMN for data quality measurement and assessment. In: Di Francescomarino, C., Dijkman, R., Zdun, U. (eds.) BPM 2019. LNBIP, vol. 362, pp. 362–374. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-37453-2_30
29. Valencia-Parra, Á., Ramos-Gutiérrez, B., Varela-Vaca, A.J., Gómez-López, M.T., Bernal, A.G.: Enabling process mining in aircraft manufactures: extracting event logs and discovering processes from complex data. In: Proceedings of the Industry Forum at BPM, Vienna, pp. 166–177 (2019)
30. Vanbrabant, L., Martin, N., Ramaekers, K., Braekers, K.: Quality of input data in emergency department simulations: framework and assessment techniques. Simul. Model. Pract. Theory **91**, 83–101 (2019)

31. Verhulst, R.: Evaluating quality of event data within event logs: an extensible framework. Master's thesis, Rijksuniversiteit Groningen, echnische Universiteit Eindhoven (2016)
32. Wynn, M.T., Sadiq, S.: Responsible process mining - a data quality perspective. In: Hildebrandt, T., van Dongen, B.F., Röglinger, M., Mendling, J. (eds.) BPM 2019. LNCS, vol. 11675, pp. 10–15. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-26619-6_2

Chapter 4

Conclusion and Future Work

This chapter concludes the thesis by summarising the main solutions proposed to achieve the goals identified. Section 4.1 concludes the thesis. After that, Section 4.2 discusses the limitations of the proposal and possible future extensions that might be made.

4.1 Conclusion

In this thesis, some new techniques have been presented to improve process discovery and process optimisation. The contributions are intended to enhance the improvement and automation of the discovery of processes in complex contexts such as logistics and manufacturing, as well as to support decision-making processes. The task of process discovery is not trivial when applied to real cases, but this problem is especially aggravated when the business context is complex, as it involves the choreography of several entities and produces a large number of complex events with different levels of abstraction.

This thesis is focused on the applicability of process mining techniques in today's industry, and therefore, we have worked in collaboration with different organisations to help them solve problems in their processes. Process discovery is an activity that has become increasingly popular in recent years and helps companies understand important aspects of the way they operate on a daily basis. However, in most cases, they do not have the time, resources, or knowledge to make a good discovery with current solutions. In this thesis, we have made some proposals for the areas less explored in the literature with the aim of covering these essential aspects. To this end, we established as our first objective (**OBJ 1**) performing a systematic review of the literature, which helped us identify areas where many challenges remain open. From these challenges, the remaining three objectives of this thesis have emerged.

In the second and third objectives, we have dealt with the problems presented by the input data of the process discovery algorithms. As in any data-driven discipline, the final results are highly dependent on the quality, form, and content of the input data. The second objective (**OBJ 2**) highlights the need to objectively measure the quality of event logs. To achieve this objective, we have proposed a series of metrics to measure and evaluate the different dimensions of event log quality. Similarly, it is necessary to know the impact of the improvements made to the input data on the quality of the resulting process models. That is why we have also proposed a set of metrics to assess the quality, complexity, and understandability of the discovered process model. Together with all these proposals, we have created a methodology that helps decision-making in a flexible way, allowing it to be fully adapted to the needs of experts in any domain. It also manages to automate some tasks that are very tedious and that, according to our analysis of the state-of-the-art, are still performed manually.

The third objective (**OBJ 3**) is related to the extraction and preparation of event logs. To achieve this objective, we have dealt with the difficulties of extracting events suitable for process mining in environments where information is distributed, belongs to different organisations, and may have different semantics. For this purpose, we have introduced two proposals: *(i)* the first is based on improving the event labels that represent the activities executed, using NLP techniques, and *(ii)* the second is based on the clustering of the execution traces present in the event logs used as input for the discovery algorithms. In both cases, the proposals can be customised according to the needs of the business domain and are based on the quality of the input data.

Finally, the fourth objective (**OBJ 4**) is related to process optimisation. To achieve this objective, we have made a proposal based on the design of a flexible business process to optimise costly aircraft manufacturing. Due to our proposal based on graphs and signal reading, we have achieved a significant cost reduction that has been validated by experienced engineers.

4.2 Future Work

The contributions made in this thesis could be improved and expanded in different areas. On the one hand, it is possible to continue working to improve the challenges addressed and, on the other hand, to propose new solutions in other important areas of process management and process mining in industry. The following list shows the next items that we will address in this line of research, listing first the proposals focused on further improving the solutions presented in this thesis and second new proposals framed in other fields of application.

- Automatic extraction of event logs from industry information systems, databases, cyber-physical systems, and sensors remains an unsolved problem. Most current solutions assume the existence of a correct event log to provide the discovery algorithms. However, this task, which is particularly complex, is still mostly performed manually. In future work, we intend to continue working on automated event detection in relational databases and unstructured data from distributed sources. We also intend to work on the automatic construction of event logs to provide the best business insight.

- Current proposals to assess the quality of incoming logs remain insufficient. In most cases, log construction becomes an iterative process that goes through numerous trial-and-error phases until a correct log is formed. In addition, there is a need for experts in the problem domain who also have knowledge of data processing and how process mining techniques work. For this reason, we will continue to work on ways to measure and evaluate the quality and semantics of the event logs.

- As mentioned above, the quality of the input data completely conditions the quality of the processes discovered. However, very little work has been done to identify the strength of this impact objectively, in general, and empirically, in particular. We are working on the largest experiment proposed so far in the literature to be able to analyse these correlations and establish dependency relationships that allow us to improve our methodology guided by the quality of the events and the resulting models.

- As a consequence of the previous point, there is a need to further improve the way of assessing the quality, complexity, and understandability of the process models discovered. We plan to extend our set of metrics and include new metrics that can be customisable according to the domain, allowing them to be generalised and applied in any context.

- Furthermore, as a consequence of this work, there is a need to analyse the correlations between the quality of the input data and the final result. Analysing so many dimensions in a mathematical way can be an extremely complicated task for domain experts. Sometimes there are no strong correlations, but there are trends that can be intuited by a correct visualisation of the data. We will work on visualisations of log quality dimensions and process models to help experts make better decisions about which techniques to apply to logs to improve specific aspects of the log to ensure that these changes have the impact they are looking for in the process models discovered.

- With respect to process optimisation, predictive process mining techniques can be applied to simulate future scenarios and adapt process execution planning on the fly. In this way, bottlenecks, losses, and resource management problems can be avoided.

- Concerning the area of process simulation, we have identified that all current solutions provide full trace simulations. We are working on a hybrid simulator whose results change depending on the data provided by an external planner that analyses the conditions of the physical medium that determine the feasibility of the process execution.

- Regarding another challenge that has been little addressed in the literature, it is necessary to work on new proposals in the area of compliance. We are working on a new methodology for process queries to verify if the actions are correct and if the model is in accordance with the input data.

Bibliography

[1] Wil Van der Aalst. "Analyzing "Spaghetti Processes"". In: *Process Mining*. Springer, 2011, pp. 301–317.

[2] Wil Van der Aalst. "Extracting Event Data from Databases to Unleash Process Mining". In: *BPM - Driving Innovation in a Digital World*. Ed. by Jan vom Brocke and Theresa Schmiedel. Springer, 2015, pp. 105–128. DOI: `10.1007/978-3-319-14430-6\_8`.

[3] Wil Van der Aalst. "Federated Process Mining: Exploiting Event Data Across Organizational Boundaries". In: *2021 IEEE International Conference on Smart Data Services (SMDS)*. IEEE. 2021, pp. 1–7.

[4] Wil Van der Aalst. "Process Discovery: Capturing the Invisible". In: *IEEE Comp. Int. Mag.* 5.1 (2010), pp. 28–41. DOI: `10.1109/MCI.2009.935307`. URL: `https://doi.org/10.1109/MCI.2009.935307`.

[5] Wil Van der Aalst. *Process Mining - Data Science in Action, Second Edition*. Springer, 2016. ISBN: 978-3-662-49850-7. DOI: `10.1007/978-3-662-49851-4`. URL: `https://doi.org/10.1007/978-3-662-49851-4`.

[6] Wil Van der Aalst. "Process mining: discovering and improving Spaghetti and Lasagna processes". In: *2011 IEEE symposium on computational intelligence and data mining (CIDM)*. IEEE. 2011, pp. 1–7.

[7] Wil Van der Aalst. *Process Mining Discovery, Conformance and Enhancement of Business Processes*. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2011. ISBN: 978-3-642-19345-3.

[8] Wil Van der Aalst. "The application of Petri nets to workflow management". In: *Journal of circuits, systems, and computers* 8.01 (1998), pp. 21–66.

[9] Wil Van der Aalst. "Using Process Mining to Bridge the Gap between BI and BPM". In: *IEEE Computer* 44.12 (2011), pp. 77–80. DOI: `10.1109/MC.2011.384`. URL: `https://doi.org/10.1109/MC.2011.384`.

[10] Wil Van der Aalst. "Verification of workflow nets". In: *International Conference on Application and Theory of Petri Nets*. Springer. 1997, pp. 407–426.

[11] Wil Van der Aalst and Christian. W. Gunther. "Finding Structure in Unstructured Processes: The Case for Process Mining". In: *Seventh International Conference on Application of Concurrency to System Design (ACSD 2007)*. 2007, pp. 3–12. DOI: `10.1109/ACSD.2007.50`.

[12] Wil Van der Aalst and Ton Weijters. "Process Mining". In: *Process-Aware Information Systems*. John Wiley & Sons, Ltd, 2005. Chap. 10, pp. 235–255. ISBN: 9780471741442. DOI: `10.1002/0471741442.ch10`.

[13] Wil Van der Aalst, Ton Weijters, and Laura Maruster. "Workflow mining: discovering process models from event logs". In: *IEEE Transactions on Knowledge and Data Engineering* 16.9 (2004), pp. 1128–1142. ISSN: 1041-4347. DOI: `10.1109/TKDE.2004.47`.

[14] Wil Van der Aalst, Ton Weijters, and Laura Maruster. *Workflow mining: Which processes can be rediscovered*. Tech. rep. Eindhoven University of Technology, 2002.

[15] Wil van der Aalst et al. "Process Mining Manifesto". In: *Business Process Management Workshops*. Ed. by Florian Daniel, Kamel Barkaoui, and Schahram Dustdar. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 169–194. ISBN: 978-3-642-28108-2.

[16] Giovanni Acampora et al. "IEEE 1849: The XES Standard: The Second IEEE Standard Sponsored by IEEE Computational Intelligence Society [Society Briefs]". In: *IEEE Comput. Intell. Mag.* 12.2 (2017), pp. 4–8. DOI: `10.1109/MCI.2017.2670420`.

[17] Rakesh Agrawal, Dimitrios Gunopulos, and Frank Leymann. "Mining process models from workflow logs". In: *Advances in Database Technology — EDBT'98*. Ed. by Hans-Jörg Schek et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 467–483. ISBN: 978-3-540-69709-1.

[18] Rainer von Ammon. "Event-Driven Business Process Management". In: *Encyclopedia of Database Systems*. Ed. by LING LIU and M. TAMER ÖZSU. Boston, MA: Springer US, 2009, pp. 1068–1071. ISBN: 978-0-387-39940-9. DOI: `10.1007/978-0-387-39940-9_577`.

[19] Rainer von Ammon et al. "Event-driven business process management and its practical application taking the example of DHL". In: (2008).

[20] Robert Andrews et al. "Quality-informed semi-automated event log generation for process mining". In: *Decis. Support Syst.* 132 (2020), p. 113265.

[21] Adriano Augusto et al. "Automated discovery of process models from event logs: review and benchmark". In: *IEEE transactions on knowledge and data engineering* 31.4 (2018), pp. 686–705.

[22] Adriano Augusto et al. "The connection between process complexity of event sequences and models discovered by process mining". In: *Information Sciences* 598 (2022), pp. 196–215. ISSN: 0020-0255. DOI: `https://doi.org/10.1016/j.ins.2022.03.072`.

[23] Nutchar Senewong Na Ayutaya, Prajin Palungsuntikul, and Wichian Premchaiswadi. "Heuristic mining: Adaptive process simplification in education". In: *2012 Tenth International Conference on ICT and Knowledge Engineering*. 2012, pp. 221–227. DOI: `10.1109/ICTKE.2012.6408559`.

[24] Carlo Batini. "Data Quality Assessment". In: *Encyclopedia of Database Systems, Second Edition*. Ed. by Ling Liu and M. Tamer Özsu. Springer, 2018.

[25] Anatoliy Batyuk and Volodymyr Voityshyn. “Streaming process discovery for lambda architecture-based process monitoring platform”. In: *2018 IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT)*. Vol. 1. IEEE. 2018, pp. 298–301.

[26] Alessandro Berti, Sebastiaan J Van Zelst, and Wil van der Aalst. “Process mining for python (PM4Py): bridging the gap between process-and data science”. In: *arXiv preprint arXiv:1905.06169* (2019).

[27] Alan W Biermann and Jerome A Feldman. “On the synthesis of finite-state machines from samples of their behavior”. In: *IEEE transactions on Computers* 100.6 (1972), pp. 592–597.

[28] Fabian Rojas Blum. *Metrics in process discovery*. Tech. rep. Tech. Rep. TR/DCC. 1–21, 2015.

[29] Alejandro Bogarín et al. “Clustering for Improving Educational Process Mining”. In: *Proceedings of the Fourth International Conference on Learning Analytics And Knowledge*. LAK ’14. New York, NY, USA: ACM, 2014, pp. 11–15. ISBN: 978-1-4503-2664-3. DOI: 10.1145/2567574.2567604.

[30] Jagadeesh Chandra Bose, Ronny S Mans, and Wil Van der Aalst. “Wanna improve process mining results?” In: *2013 IEEE symposium on computational intelligence and data mining (CIDM)*. IEEE. 2013, pp. 127–134.

[31] Seppe K. L. M. vanden Broucke and Jochen De Weerdt. “Fodina: A robust and flexible heuristic process discovery technique”. In: *Decision Support Systems* 100 (2017), pp. 109–118. DOI: 10.1016/j.dss.2017.04.005.

[32] Berkay Buharali. “Data science in design process”. PhD thesis. Master’s thesis, TU/e-Eindhoven University of Technology, 2015.

[33] Joos Buijs, Boudewijn F van Dongen, and Wil Van der Aalst. “Quality dimensions in process discovery: The importance of fitness, precision, generalization and simplicity”. In: *International Journal of Cooperative Information Systems* 23.01 (2014), p. 1440001.

[34] Jorge Cardoso. “Control-flow complexity measurement of processes and Weyuker’s properties”. In: *6th International Enformatika Conference*. Vol. 8. 2005, pp. 213–218.

[35] Jorge Cardoso. “Evaluating the process control-flow complexity measure”. In: *IEEE International Conference on Web Services (ICWS’05)*. IEEE. 2005.

[36] Filip Caron et al. “A process mining-based investigation of adverse events in care processes”. In: *Health Information Management Journal* 43.1 (2014), pp. 16–25.

[37] David Chapela, Manuel Mucientes, and Manuel Lama. “Discovering Infrequent Behavioral Patterns in Process Models”. In: *15th International Conference, BPM 2017, Proceedings*. 2017, pp. 324–340.

[38] Hsin-Jung Cheng and Akhil Kumar. “Process mining on noisy logs - Can log sanitization help to improve performance?” In: *Decision Support Systems* 79 (2015), pp. 138–149. DOI: 10.1016/j.dss.2015.08.003.

[39] Raffaele Conforti, M. La Rosa, and A. ter Hofstede. "Timestamp repair for business process event logs." In: *http://hdl.handle.net/11343/209011* (2018).

[40] Raffaele Conforti, Marcello La Rosa, and Arthur H. M. ter Hofstede. "Filtering Out Infrequent Behavior from Business Process Event Logs". In: *IEEE Trans. Knowl. Data Eng.* 29.2 (2017), pp. 300–314. DOI: 10.1109/TKDE.2016.2614680.

[41] Jonathan E. Cook and Alexander L. Wolf. "Discovering Models of Software Processes from Event-based Data". In: *ACM Trans. Softw. Eng. Methodol.* 7.3 (July 1998), pp. 215–249. ISSN: 1049-331X. DOI: 10.1145/287000.287001.

[42] Jonathan E. Cook et al. "Discovering models of behavior for concurrent workflows". In: *Computers in industry* 53.3 (2004), pp. 297–319.

[43] Angelo Corallo, Mariangela Lazoi, and Fabrizio Striani. "Process mining and industrial applications: A systematic literature review". In: *Knowledge and Process Management* 27.3 (2020), pp. 225–233.

[44] Sreerupa Das and Michael C Mozer. "A unified gradient-descent/clustering architecture for finite state machine induction". In: *Advances in neural information processing systems*. 1994, pp. 19–26.

[45] Vadim Denisov, Dirk Fahland, and Wil Van der Aalst. "Repairing Event Logs with Missing Events to Support Performance Analysis of Systems with Shared Resources". In: *Application and Theory of Petri Nets and Concurrency*. Springer, 2020, pp. 239–259. ISBN: 978-3-030-51831-8.

[46] Amit V Deokar and Jie Tao. "Semantics-based event log aggregation for process mining and analytics". In: *Information Systems Frontiers* 17.6 (2015), pp. 1209–1226.

[47] Prabhakar M. Dixit, Joos Buijs, and Wil Van der Aalst. "ProDiGy : Human-in-the-loop process discovery". In: *12th International Conference on Research Challenges in Information Science, RCIS 2018, Nantes, France, May 29-31, 2018*. IEEE, 2018, pp. 1–12. DOI: 10.1109/RCIS.2018.8406657.

[48] Marlon Dumas et al. *Fundamentals of business process management*. Vol. 1. Springer, 2013.

[49] Christoph Emmersberger, Florian Springer, and Christian Wolff. "Location Based Logistics Services and Event Driven Business Process Management". In: *IMC*. Vol. 53. Communications in Computer and Information Science. 2009, pp. 167–177. DOI: 10.1007/978-3-642-10263-9_15.

[50] Can Eren. "Providing running case predictions based on contextual information". PhD thesis. PhD thesis, EINDHOVEN UNIVERSITY OF TECHNOLOGY, 2012.

[51] David M. Eyers et al. "Integrating Process-Oriented and Event-Based Systems (Dagstuhl Seminar 16341)". In: *Dagstuhl Reports* 6.8 (2016), pp. 21–64. DOI: 10.4230/DagRep.6.8.21.

[52] Diogo R. Ferreira and Cláudia Alves. "Discovering User Communities in Large Event Logs". In: *Business Process Management Workshops - BPM 2011 International Workshops, Clermont-Ferrand, France, August 29, 2011, Revised Selected Papers, Part I*. Ed. by Florian Daniel, Kamel Barkaoui, and Schahram Dustdar. Vol. 99. Lecture Notes in Business Information Processing. Springer, 2011, pp. 123–134. DOI: `10.1007/978-3-642-28108-2\_11`.

[53] Dominik Andreas Fischer et al. "Enhancing Event Log Quality: Detecting and Quantifying Timestamp Imperfections". In: *BPM*. Springer International Publishing, 2020, pp. 309–326. ISBN: 978-3-030-58666-9.

[54] Lee Geishecker and Frank Buytendijk. *Introducing the CPM Suites Magic Quadrant*. URL: `www.gartner.com`.

[55] Lucantonio Ghionna et al. "Outlier Detection Techniques for Process Mining Applications". In: *Foundations of Intelligent Systems*. Ed. by Aijun An et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 150–159. ISBN: 978-3-540-68123-6.

[56] María Teresa Gómez López, Rafael M. Gasca Gasca, and José Miguel Pérez-Álvarez. "Compliance validation and diagnosis of business data constraints in business processes at runtime". In: *Information Systems* 48 (2015), pp. 26–43.

[57] Gianluigi Greco et al. "Discovering expressive process models by clustering log traces". In: *IEEE Transactions on Knowledge and Data Engineering* 18.8 (2006), pp. 1010–1027.

[58] Volker Gruhn and Ralf Laue. "Approaches for Business Process Model Complexity Metrics". In: *Technologies for Business Information Systems*. Ed. by Witold Abramowicz and Heinrich C. Mayr. Dordrecht: Springer Netherlands, 2007, pp. 13–24. ISBN: 978-1-4020-5634-5. DOI: `10.1007/1-4020-5634-6_2`.

[59] Christian W Günther. "Process mining in flexible environments". In: (2009).

[60] Christian W. Günther. "XES Extensible Event Stream standard definition". In: *Sl: sn* (2009).

[61] Christian W Günther and Wil Van der Aalst. "Fuzzy mining–adaptive process simplification based on multi-perspective metrics". In: *International conference on business process management*. Springer. 2007, pp. 328–343.

[62] Michal Halaška and Roman Šperka. "Process Mining – the Enhancement of Elements Industry 4.0". In: *2018 4th International Conference on Computer and Information Sciences (ICCOINS)*. 2018, pp. 1–6. DOI: `10.1109/ICCOINS.2018.8510578`.

[63] Kees M. van Hee, Natalia Sidorova, and Jan Martijn E. M. van der Werf. "Business Process Modeling Using Petri Nets". In: *Trans. Petri Nets Other Model. Concurr.* 7 (2013), pp. 116–161. DOI: `10.1007/978-3-642-38143-0_4`.

[64] B. F. A. Hompes, H. M. W. Verbeek, and Wil. Van der Aalst. “Finding Suitable Activity Clusters for Decomposed Process Discovery”. In: *Data-Driven Process Discovery and Analysis*. Ed. by Paolo Ceravolo, Barbara Russo, and Rafael Accorsi. Cham: Springer International Publishing, 2015, pp. 32–57.

[65] Barbara Kitchenham, David Budgen, and Pearl Brereton. *Evidence-Based Software Engineering and Systematic Reviews*. 1st ed. An optional note. CRC Press, 2015. ISBN: 9781482228656.

[66] Barbara Kitchenham and Stuart Charters. *Guidelines for performing Systematic Literature Reviews in Software Engineering*. Tech. rep. EBSE-2007-01. School of Computer Science and Mathematics, Keele University., 2007.

[67] Marco Kuhrmann, Daniel Méndez Fernández, and Maya Daneva. “On the pragmatic design of literature studies in software engineering: an experience-based guideline”. In: *Empirical Software Engineering* 22.6 (2017), pp. 2852–2891. DOI: `10.1007/s10664-016-9492-y`.

[68] Antti M Latva-Koivisto. “Finding a complexity measure for business process models”. In: *Helsinki University of Technology, Systems Analysis Laboratory* (2001).

[69] Sander J. J. Leemans, Dirk Fahland, and Wil Van der Aalst. “Discovering Block-Structured Process Models from Event Logs Containing Infrequent Behaviour”. In: *Business Process Management Workshops - BPM 2013 International Workshops, Beijing, China, August 26, 2013, Revised Papers*. Ed. by Niels Lohmann, Minseok Song, and Petia Wohed. Vol. 171. Lecture Notes in Business Information Processing. Springer, 2013, pp. 66–78. DOI: `10.1007/978-3-319-06257-0\_6`.

[70] Sander J. J. Leemans, Dirk Fahland, and Wil Van der Aalst. “Discovering Block-Structured Process Models from Incomplete Event Logs”. In: *Petri Nets*. Vol. 8489. Lecture Notes in Computer Science. Springer, 2014, pp. 91–110.

[71] Sander J. J. Leemans, Dirk Fahland, and Wil Van der Aalst. “Scalable Process Discovery with Guarantees”. In: *Enterprise, Business-Process and Information Systems Modeling - 16th International Conference, BPMDS 2015, 20th International Conference, EMMSAD 2015, Held at CAiSE 2015, Stockholm, Sweden, June 8-9, 2015, Proceedings*. Ed. by Khaled Gaaloul et al. Vol. 214. Lecture Notes in Business Information Processing. Springer, 2015, pp. 85–101. DOI: `10.1007/978-3-319-19237-6\_6`.

[72] Massimiliano de Leoni, Wil Van der Aalst, and Marcus Dees. “A general framework for correlating, predicting and clustering dynamic behavior based on event logs”. In: *Inf. Syst.* 56 (2016), pp. 235–257. DOI: `10.1016/j.is.2015.07.003`.

[73] Ricardo Lira et al. “Tailored Process Feedback Through Process Mining for Surgical Procedures in Medical Training: The Central Venous Catheter Case”. In: *Business Process Management Workshops*. Ed. by Florian Daniel, Quan Z. Sheng, and Hamid Motahari. Cham: Springer International Publishing, 2019, pp. 163–174. ISBN: 978-3-030-11641-5.

[74] Angel R Martínez Lorente, Frank Dewhurst, and Barrie G Dale. “TQM and business innovation”. In: *European Journal of Innovation Management* (1999).

[75] David C. Luckham. *The power of events - an introduction to complex event processing in distributed enterprise systems*. ACM, 2005. ISBN: 978-0-201-72789-0.

[76] Linh Thao Ly et al. “Data Transformation and Semantic Log Purging for Process Mining”. In: *CAiSE*. Vol. 7328. Lecture Notes in Computer Science. Springer, 2012, pp. 238–253.

[77] David J. C. MacKay. *Information Theory, Inference & Learning Algorithms*. New York, NY, USA: Cambridge University Press, 2002. ISBN: 0521642981.

[78] Adetokunbo Makanju, A. Nur Zincir-Heywood, and Evangelos E. Milios. “Clustering event logs using iterative partitioning”. In: *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, June 28 - July 1, 2009*. Ed. by John F. Elder IV et al. ACM, 2009, pp. 1255–1264. DOI: 10.1145/1557019.1557154.

[79] Adetokunbo Makanju et al. “LogView: Visualizing Event Log Clusters”. In: *Sixth Annual Conference on Privacy, Security and Trust, PST 2008, October 1-3, 2008, Fredericton, New Brunswick, Canada*. Ed. by Larry Korba, Stephen Marsh, and Reihaneh Safavi-Naini. IEEE Computer Society, 2008, pp. 99–108. DOI: 10.1109/PST.2008.17.

[80] Niels Martin. “Using Indoor Location System Data to Enhance the Quality of Healthcare Event Logs: Opportunities and Challenges”. In: *Business Process Management Workshops*. Ed. by Florian Daniel, Quan Z. Sheng, and Hamid Motahari. Cham: Springer International Publishing, 2019, pp. 226–238. ISBN: 978-3-030-11641-5.

[81] Niels Martin et al. “Opportunities and challenges for process mining in organizations: results of a Delphi study”. In: *Business & Information Systems Engineering* 63.5 (2021), pp. 511–527.

[82] Laura Maruster et al. “A Rule-Based Approach for Process Discovery: Dealing with Noise and Imbalance in Process Logs”. In: *Data Min. Knowl. Discov.* 13.1 (2006), pp. 67–87.

[83] Thomas J McCabe. “A complexity measure”. In: *IEEE Transactions on software Engineering* 4 (1976), pp. 308–320.

[84] Jan Mendling. *Metrics for process models: empirical foundations of verification, error prediction, and guidelines for correctness*. Vol. 6. Springer Science & Business Media, 2008.

[85] Jan Mendling. “Testing density as a complexity metric for EPCs”. In: *German EPC workshop on density of process models*. Vol. 19. 2006.

[86] Jan Mendling, Gustaf Neumann, and Wil van der Aalst. “Understanding the Occurrence of Errors in Process Models Based on Metrics”. In: *On the Move to Meaningful Internet Systems 2007: CoopIS, DOA, ODBASE, GADA, and IS*. Ed. by Robert Meersman and Zahir Tari. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 113–130. ISBN: 978-3-540-76848-7.

[87] Franz-Benjamin Mocnik, Hongchao Fan, and Alexander Zipf. "Data Quality and Fitness for Purpose". In: May 2017. DOI: 10.13140/RG.2.2.13387.18726.

[88] Hoang Thi Cam Nguyen et al. "Autoencoders for improving quality of process event logs". In: *Expert Systems with Applications* 131 (2019), pp. 132–147.

[89] OMG. *Business Process Model and Notation (BPMN), Version 2.0.2*. Tech. rep. Object Management Group, Dec. 2013.

[90] OMG. *Decision Model and Notation (DMN), Version 1.2*. Object Management Group, 2019.

[91] Boris Otto, Yang W. Lee, and Ismael Caballero. "Information and data quality in networked business". In: *Electronic Markets* 21.2 (2011), pp. 79–81.

[92] Timothy Lawrence Pascoe. "Allocation of resources CPM". In: *REVUE FRANCAISE DE RECHERCHE OPERATIONNELE* 10.38 (1966), pp. 31–31.

[93] Ken Peffers et al. "A Design Science Research Methodology for Information Systems Research". In: *Journal of Management Information Systems* 24.3 (2014), pp. 45–77. ISSN: 07421222. DOI: 10.2753/MIS0742-1222240302.

[94] Marcos Rivas Peña and Sussy Bayona-Oré. "Process Mining and Automatic Process Discovery". In: *2018 7th International Conference On Software Process Improvement (CIMPS)*. IEEE. 2018, pp. 41–46.

[95] Fabian Pittke, Henrik Leopold, and Jan Mendling. "When language meets language: Anti patterns resulting from mixing natural and modeling language". In: *International Conference on BPM*. Springer. 2014, pp. 118–129.

[96] Wouter Poncin, Alexander Serebrenik, and Mark van den Brand. "Process Mining Software Repositories". In: *15th European Conference on Software Maintenance and Reengineering, CSMR 2011, 1-4 March 2011, Oldenburg, Germany*. Ed. by Tom Mens, Yiannis Kanellopoulos, and Andreas Winter. IEEE Computer Society, 2011, pp. 5–14. DOI: 10.1109/CSMR.2011.5.

[97] Belén Ramos-Gutiérrez et al. "A NLP-Oriented Methodology to Enhance Event Log Quality". In: *Enterprise, Business-Process and Information Systems Modeling - 22nd International Conference, BPMDS 2021, and 26th International Conference, EMMSAD 2021, Held at CAiSE 2021, Melbourne, VIC, Australia, June 28-29, 2021, Proceedings*. Ed. by Adriano Augusto et al. Vol. 421. Lecture Notes in Business Information Processing. Springer, 2021, pp. 19–35. DOI: 10.1007/978-3-030-79186-5_2.

[98] Belén Ramos-Gutiérrez et al. "Discovering configuration workflows from existing logs using process mining". In: *Empir. Softw. Eng.* 26.1 (2021), p. 11. DOI: 10.1007/s10664-020-09911-x.

[99] Belén Ramos-Gutiérrez et al. "Self-Adaptative Troubleshooting for to Guide Resolution of Malfunctions in Aircraft Manufacturing". In: *IEEE Access* 9 (2021), pp. 42707–42723. DOI: 10.1109/ACCESS.2021.3066253. URL: https://doi.org/10.1109/ACCESS.2021.3066253.

[100] Belén Ramos-Gutiérrez et al. "When business processes meet complex events in logistics: A systematic mapping study". In: *Comput. Ind.* 144 (2023), p. 103788. DOI: `10.1016/j.compind.2022.103788`.

[101] Hind R'bigui and Chiwoon Cho. "The state-of-the-art of business process mining challenges". In: *International Journal of Business Process Integration and Management* 8.4 (2017), pp. 285–303.

[102] Adrian Rebmann and Han van der Aa. "Extracting Semantic Process Information from the Natural Language in Event Logs". In: *CoRR* abs/2103.11761 (2021). arXiv: `2103.11761`.

[103] Vasja Roblek, Maja Meško, and Alojz Krapež. "A complex view of industry 4.0". In: *Sage open* 6.2 (2016), p. 2158244016653987.

[104] George D Robson. *Continuous process improvement*. Simon and Schuster, 2010.

[105] Anne Rozinat and Wil Van der Aalst. "Conformance checking of processes based on monitoring real behavior". In: *Inf. Syst.* 33.1 (2008), pp. 64–95. DOI: `10.1016/j.is.2007.07.001`.

[106] Sareh Sadeghianasl et al. "A Contextual Approach to Detecting Synonymous and Polluted Activity Labels in Process Event Logs". In: *OTM 2019 Conferences*. Springer, 2019, pp. 76–94.

[107] Sareh Sadeghianasl et al. "Collaborative and Interactive Detection and Repair of Activity Labels in Process Event Logs". In: *2nd International Conference on Process Mining, ICPM 2020, Padua, Italy, October 4-9, 2020*. Ed. by Boudewijn F. van Dongen, Marco Montali, and Moe Thandar Wynn. IEEE, 2020, pp. 41–48. DOI: `10.1109/ICPM49681.2020.00017`. URL: `https://doi.org/10.1109/ICPM49681.2020.00017`.

[108] Mohammadreza Fani Sani, Sebastiaan J. van Zelst, and Wil Van der Aalst. "Improving Process Discovery Results by Filtering Outliers Using Conditional Behavioural Probabilities". In: *Business Process Management Workshops - BPM 2017 International Workshops, Barcelona, Spain, September 10-11, 2017, Revised Papers*. Ed. by Ernest Teniente and Matthias Weidlich. Vol. 308. Lecture Notes in Business Information Processing. Springer, 2017, pp. 216–229. DOI: `10.1007/978-3-319-74030-0\_16`.

[109] Cleiton dos Santos Garcia et al. "Process mining techniques and applications–A systematic mapping study". In: *Expert Systems with Applications* 133 (2019), pp. 260–295.

[110] August-Wilhelm Scheer, Oliver Thomas, and Otmar Adam. "Process Modeling Using Event-Driven Process Chains." In: *Process-aware information systems* 119 (2005).

[111] Josef Schiefer et al. "Simulating Business Process Scenarios for Event-Based Systems". In: *ECIS*. University of St. Gallen, 2007, pp. 1729–1740.

[112] Roger G. Schroeder et al. "Six Sigma: Definition and underlying theory". In: *Journal of Operations Management* 26.4 (2008). Special Issue: Research in Supply Chain Quality, pp. 536–554. ISSN: 0272-6963. DOI: `https://doi.org/10.1016/j.jom.2007.06.007`.

[113] Feri Setiawan, Bernardo Nugroho Yahya, and Seok-Lyong Lee. "A VERIFICATION APPROACH FOR HUMAN BEHAVIOR MODELING." In: *International Journal of Industrial Engineering* 25.1 (2018).

[114] Minseok Song, Christian W. Günther, and Wil Van der Aalst. "Trace Clustering in Process Mining". In: *Business Process Management Workshops, BPM 2008 International Workshops, Milano, Italy, September 1-4, 2008. Revised Papers*. Ed. by Danilo Ardagna, Massimo Mecella, and Jian Yang. Vol. 17. Lecture Notes in Business Information Processing. Springer, 2008, pp. 109–120. DOI: 10.1007/978-3-642-00328-8_11.

[115] Suriadi Suriadi et al. "Event log imperfection patterns for process mining: Towards a systematic approach to cleaning event logs". In: *Inf. Syst.* 64 (2017), pp. 132–150. DOI: 10.1016/j.is.2016.07.011. URL: https://doi.org/10.1016/j.is.2016.07.011.

[116] Niek Tax, Natalia Sidorova, and Wil Van der Aalst. "Discovering more precise process models from event logs by filtering out chaotic activities". In: *J. Intell. Inf. Syst.* 52.1 (2019), pp. 107–139. DOI: 10.1007/s10844-018-0507-6.

[117] Álvaro Valencia-Parra et al. "DMN4DQ: When data quality meets DMN". In: *Decision Support Systems* 141 (2021), p. 113450. ISSN: 0167-9236. DOI: https://doi.org/10.1016/j.dss.2020.113450.

[118] Wil Van Der Aalst. "Process mining: Overview and opportunities". In: *ACM Transactions on Management Information Systems (TMIS)* 3.2 (2012), pp. 1–17.

[119] Lien Vanbrabant et al. "Quality of input data in emergency department simulations: Framework and assessment techniques". In: *Simul. Model. Pract. Theory* 91 (2019), pp. 83–101.

[120] Irene Vanderfeesten et al. "Quality metrics for business process models". In: *BPM and Workflow handbook* 144 (2007), pp. 179–190.

[121] Ángel Jesús Varela-Vaca et al. "Process mining to unleash variability management: discovering configuration workflows using logs". In: *Proceedings of the 23rd International Systems and Software Product Line Conference, SPLC 2019, Volume A, Paris, France, September 9-13, 2019*. Ed. by Thorsten Berger et al. ACM, 2019, 37:1–37:12. DOI: 10.1145/3336294.3336303.

[122] H. M. W. Verbeek et al. "XES, XESame, and ProM 6". In: *Information Systems Evolution - CAiSE Forum 2010, Hammamet, Tunisia, June 7-9, 2010, Selected Extended Papers*. Ed. by Pnina Soffer and Erik Proper. Vol. 72. Lecture Notes in Business Information Processing. Springer, 2010, pp. 60–75. DOI: 10.1007/978-3-642-17722-4_5. URL: https://doi.org/10.1007/978-3-642-17722-4_5.

[123] Rick Verhulst. "Evaluating quality of event data within event logs: an extensible framework". MA thesis. echnische Universiteit Eindhoven: Rijksuniversiteit Groningen, 2016.

[124] Jochen De Weerdt et al. "Active Trace Clustering for Improved Process Discovery". In: *IEEE Transactions on Knowledge and Data Engineering* 25.12 (2013), pp. 2708–2720.

[125] Ton Weijters and Wil Van der Aalst. "Rediscovering workflow models from event-based data". In: *Proceedings of the 11th Dutch-Belgian Conference on Machine Learning (Benelearn 2001)*. 2001, pp. 93–100.

[126] Ton Weijters and Wil Van der Aalst. "Workflow Mining Discovering Workflow Models from Event-Based Data". In: *Knowledge Discovery from Temporal and Spatial Data (W12)* (2002), p. 6.

[127] Ton Weijters, Wil Van der Aalst, and AK Alves De Medeiros. "Process mining with the heuristics miner-algorithm". In: *Technische Universiteit Eindhoven, Tech. Rep. WP* 166 (2006), pp. 1–34.

[128] Ton Weijters and J. T. S. Ribeiro. "Flexible Heuristics Miner (FHM)". In: *CIDM*. IEEE, 2011, pp. 310–317.

[129] Ton Weijters and JTS Ribeiro. "Flexible heuristics miner (FHM)". In: *2011 IEEE symposium on computational intelligence and data mining (CIDM)*. IEEE. 2011, pp. 310–317.

[130] Mathias Weske. "Process Management Concepts, Languages, Architectures". In: (2012).

[131] Claes Wohlin. "Guidelines for snowballing in systematic literature studies and a replication in software engineering". In: *EASE*. ACM, 2014, 38:1–38:10. DOI: 10.1145/2601248.2601268.

[132] Moe Thandar Wynn and Shazia Sadiq. "Responsible Process Mining - A Data Quality Perspective". In: *Business Process Management*. Springer, 2019, pp. 10–15. ISBN: 978-3-030-26619-6.

[133] Anton Yeshchenko et al. "Context-Aware Predictive Process Monitoring: The Impact of News Sentiment". In: *On the Move to Meaningful Internet Systems. OTM 2018 Conferences*. Ed. by Hervé Panetto et al. Cham: Springer International Publishing, 2018, pp. 586–603. ISBN: 978-3-030-02610-3.

[134] Fadwa Zaoui and Nissrine Souissi. "Roadmap for digital transformation: A literature review". In: *Procedia Computer Science* 175 (2020). The 17th International Conference on Mobile Systems and Pervasive Computing (MobiSPC),The 15th International Conference on Future Networks and Communications (FNC),The 10th International Conference on Sustainable Energy Information Technology, pp. 621–628. ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2020.07.090.