Volume 11 • Issue 2 • 2019

sciendo

International Society for Manufacturing, Service and Management Engineering

received: 2 January 2019
accepted: 15 June 2019

pages: 106-118

Performance of an automated process model discovery – the logistics process of a manufacturing company

Michal Halaška, Roman Šperka

ABSTRACT

The simulation and modelling paradigms have significantly shifted in recent years under the influence of the Industry 4.0 concept. There is a requirement for a much higher level of detail and a lower level of abstraction within the simulation of a modelled system that continuously develops. Consequently, higher demands are placed on the construction of automated process models. Such a possibility is provided by automated process discovery techniques. Thus, the paper aims to investigate the performance of automated process discovery techniques within the controlled environment. The presented paper aims to benchmark the automated discovery techniques regarding realistic simulation models within the controlled environment and, more specifically, the logistics process of a manufacturing company. The study is based on a hybrid simulation of logistics in a manufacturing company that implemented the AnyLogic framework. The hybrid simulation is modelled using the BPMN notation using BIMP, the business process modelling software, to acquire data in the form of event logs. Next, five chosen automated process discovery techniques are applied to the event logs, and the results are evaluated. Based on the evaluation of benchmark results received using the chosen discovery algorithms, it is evident that the discovery algorithms have a better overall performance using more extensive event logs both in terms of fitness and precision. Nevertheless, the discovery techniques perform better in the case of smaller data sets, with less complex process models. Typically, automated discovery techniques have to address scalability issues due to the high amount of data present in the logs. However, as demonstrated, the process discovery techniques can also encounter issues of opposite nature. While discovery techniques typically have to address scalability issues due to large datasets, in the case of companies with long delivery cycles, long processing times and parallel production, which is common for the industrial sector, they have to address issues with incompleteness and lack of information in datasets. The management of business companies is becoming essential for companies to stay competitive through efficiency. The issues encountered within the simulation model will be amplified through both vertical and horizontal integration of the supply chain within the Industry 4.0. The impact of vertical integration in the BPMN model and the chosen case identifier is demonstrated. Without the assumption of smart manufacturing, it would be impossible to use a single case identifier throughout the entire simulation. The entire process would have to be divided into several subprocesses.

KEY WORDS

process mining, automated process discovery, simulation, agent-based simulation, ABS

DOI: 10.2478/emj-2019-0014

Corresponding author:

Michal Halaška

Silesian University in Opava,
Czech Republic
e-mail: halaska@opf.slu.cz

Roman Šperka

Silesian University in Opava,
Czech Republic
e-mail: sperka@opf.slu.cz

Introduction

In today's world, various industries and economic sectors are changing as a result of the digital transformation, which is part of fourth industrial revolution called "Industry 4.0" (Slusarczyk, 2018; Qin et al., 2016). The shift from simple digitisation of the previous industrial revolution is going to force companies across the supply chain to re-examine the way they do business. The concept of Industry 4.0 was introduced by the German government in 2011 in the context of its hi-tech strategy aimed at the industrial sector and was quickly adopted all around the world. Industry 4.0 is closely linked to the Internet of Things (IoT) and represents the ability of industrial components to communicate with each other (Roblek et al., 2016; Piccarozzi et al., 2018). However, Industry 4.0 does not relate only to the digitalisation of the industrial sector but considers the entire value-added chain, including sub-systems like, for example, research and development, retailers, suppliers, customers etc. The main idea of the Industry 4.0 concept is aimed at preserving the competitiveness in the light of increasingly more demanding customers. Several concepts and technologies are available for the fulfilment of the main objective of Industry 4.0. Firstly, the literature specifies three types of integration: horizontal, vertical and end-to-end integration. Vertical integration considers processes within an organisation, while horizontal integration emphasises cross-organisational processes within a value chain (Sony, 2018). End-to-end integration assumes the involvement of the product itself within both the horizontal and vertical integrations (Wang et al., 2016). Secondly, the particular integrations are joined by concepts of smart factories, smart product, new business models and new customer services (Qin et al., 2016). Thirdly, there are several leading technological solutions with a major impact on production and services: Cyber-Physical Systems (CPS), big data analytics, cloud computing, autonomous machines, simulations, augmented reality, IoT etc. (Pan et al., 2015; Kolberg & Zühlke, 2015), where the use of all of such technologies leads towards further digitisation and computerisation of production, service and market processes. Thus, in the future of manufacturing, the particular sub-systems of a value-added chain will be connected into an intelligent network with the use of CPS to relate physical and virtual spaces. This paradigm shift means that an information system manages an intelligent network while considering physical factors to allow independent process management, which represents a fundamentally new aspect of the production process leading towards reshaping of production, consumption, transportation and delivery systems (Rodič, 2017). As it is obvious from the literature regarding the Industry 4.0, there will be significant requirements for the management of business processes across the entire value-added chain.

Thus, the subject of this study is big data analytics and simulations, or, to be more precise, process mining and agent-based modelling and simulation (ABS) because of their potential to enhance process management of the network of sub-systems within the Industry 4.0 concept. The further digitisation and computerisation of business processes within the Industry 4.0 mean that less common approaches within the business practice, like ABS, are becoming trendy, and it is expected that in the near future, they will become common in many areas of business. One of the conditions of successful ABS employment is the ability to automatically produce appropriate process models that can be exploited by ABS. The idea is supported by other requirements for Industry 4.0, such as self-organisation, self-adaptation, reconfigurability, self-awareness etc. (Pisching et al., 2018; Dinardo et al., 2018; Wan et al., 2017). They will be part of the process management of an intelligent network of a value chain. And the automated discovery of process models will be necessary not only with regards to the use of ABS within Industry 4.0 but also within the Industry 4.0 concept in general. Thus, the objective of this research paper is to benchmark the automated process discovery techniques on the realistic simulation models of supply-chain elements. The design of supply chain and operations play a significant role in the success or failure of a company (Kozma, 2017). As it is crucial for a simulation model to work with the most precise model possible to ensure that the following analysis brings the best outputs in terms of enhancement, the prediction and understanding of the investigated system were based on the data produced by the system. If the business processes are poorly designed or contain errors, then customer needs are not fully satisfied due to the insufficient performance of the process. Similarly, if, using the simulation modelling at both operational and strategic level, the decision-making process is based on unprecise process models, the impacts will be equivalently bad.

Simulation modelling is used for the representation of real or imagined systems or processes for the purpose of its analysis and understanding. Today, the use of simulation modelling is well established in science, engineering etc. (Abar et al., 2017). It is used for prediction, performance analysis, process discovery, etc. In business practice, modelling is used mainly as a tool for operational and strategic management and decision making in many of its areas like marketing, management, logistics, scheduling, etc. Simulation modelling is powerful because it allows investigating the influence of random variables on a dynamic system using both quantitative and qualitative views

(Doomun & Jungun, 2008; Hlupić & Vukšić, 2004). There are many approaches to simulation modelling, such as analytical modelling, based mostly on mathematical theories (Gries et al., 2016), system dynamics (Macal, 2010; Borshchev & Filippov, 2004), discrete event simulation (Siebers et al., 2010; Chan, Son & Macal, 2010). However, ABS is becoming increasingly more popular for several reasons. Firstly, it offers a broad scope of analysis in terms of levels of the used abstraction of complex modelled systems, thus allowing the analysis of much greater detail than is possible using other paradigms. Active elements of the system are represented by software agents with defined behavioural patterns replicating the complexity of the system (Kelly et al., 2013; North & Macal, 2008). Secondly, evidence is available showing that ABS work well with the most crucial technologies and concepts of Industry 4.0, be it IoT or smart products (Savaglio et al., 2017), smart manufacturing (Bannat et al., 2011), vertical integration (Hsieh, 2015), CPS (Leitao et al., 2016), autonomy and related self-organisation, self-awareness, machine-human and machine-machine interaction etc., (Boes & Migeon, 2017; Pomarlan & Bateman, 2018; Claes et al., 2017).

Several research papers attempt to evaluate the performance of automated process discovery techniques, for example, Augusto et al. (2018) and Weerdt et al. (2012). This paper is organised as follow: the following section presents a literature review of process mining techniques with a focus on the automated process discovery. The third section details the methodology of the research. The fourth section comments on the results of the benchmarking of the chosen process discovery techniques. Finally, the results are summarised and discussed.

1. Process mining

Process mining is a group of techniques combining the data-based point of view of data science with the process-oriented one. Process mining is related to the general domain of knowledge discovery in databases (KDD) as it has a similar approach to the analysis of large repositories of data and learning from them. Similarly to KDD, within the process mining domain, researchers developed numerous quantitative techniques and approaches to allow examining the execution of traces of business activities from the process-oriented perspective. In that sense, the focus of process mining is on processes and makes the distinction between process mining and KDD or business intelligence (BI) tools, eminent as the BI tools, focus primarily on key performance indicators (KPIs) and, thus, lack the ability to provide insight into the root causes of process inefficiency and erroneousness (Weerdt et al., 2012). Process mining can be defined as a group of techniques that search for hidden information and patterns in the data allowing for the performance analysis of the actual processes based on data produced by processes itself (Aalst, 2016; Aalst et al., 2011). This data is stored by information systems supporting such processes and recording execution events of processes, such as the start of the case, the execution of the task within a case, and others. There are various properties of an event that can be tracked and recorded, such as timestamps, costs, prices etc. The sequence of all events related to a particular case is called a trace, and the collection of such records is referred to as an event log. Thus, an event log has to carry certain minimal information to be applicable for a process mining analysis. Firstly, it has to distinguish between particular process instance or cases. Secondly, events within cases have to be ordered and, lastly, there has to be a function that assigns actions to events within the log (Aalst, 2015). As there are information and communication technologies in the background of the main driving forces of Industry 4.0, there will be a considerable amount of event logs produced by information systems supporting the processes of Industry 4.0, such as CPS, enterprise information systems, enterprise resource planning systems etc.

As of now, there are five significant areas of research within the process mining domain. The automated process discovery focuses on building process models from real data using various algorithms and approaches (Aalst, 2016). Next, conformance is checked using the evaluation and comparison of process models and event logs based on different criteria to identify commonalities and discrepancies between behaviour of process models, process model and event log or event logs (Buijs, Dongen & Aalst, 2014; Buijs, Dongen & Aalst, 2012; Aalst, 2005). The enhancement of the process means the extension or improvement of an existing process model using the information about an actual process in some event log (Aalst, 2016). Further, operational support focuses on particular processes online and in real time. This means that operational support not only uses post-mortem data but also pre-mortem data from unfinished process instances (Aalst et al., 2011). Lastly, there is deviance mining, which is a group of techniques used to analyse deviances of different variants of processes (Nguyen et al., 2016). The process mining techniques are briefly introduced for the sake of completeness, and the focus of the rest of this section is on automated process discovery techniques.

1.1. Automated process discovery

Throughout the process mining literature, out of the previously mentioned areas, the automated process discovery is the most widely researched. As input, automated process discovery techniques take an event log containing the information about the behaviour of the analysed process, and then produce a process model representing control-flow containing relations between tasks observed or implied in the event log (Aalst, Weijters & Maruster, 2004). However, for discovered process models to be useful, they have to find an appropriate balance between several properties (Aalst, 2016; Buijs, Dongen & Aalst, 2014), such as fitness, precision, generalisation and simplicity. The fitness quality dimension describes the fraction of the behaviour in the event log that can be replayed by the process model, essentially meaning that the discovery method generates traces that are present in the log or are similar to a trace in the log. On the other hand, the precision quality dimension estimates the behaviour unseen in the event log but allowed by the process model, essentially meaning that the discovered model should not generate traces that are too different from the behaviour seen in the log. The generalisation quality dimension is indicated if the event log is not overfitting the behaviour present in the event log as the event log itself may contain only partial behaviour of the analysed system, essentially meaning that the discovery method generates traces not seen in the model which have similar behaviour to the traces seen in the event log. Finally, the simplicity quality dimension states that the discovered process model should be as simple as possible. As criteria go against each other, it is necessary to find the appropriate balance between them; however, this is not an easy task, especially considering real-life event logs. Thus, according to Augusto et al. (2018), there are two major problems which occur during the application of automated process discovery methods on real-life event logs: 1) the discovery method produces large spaghetti-like models (Fig. 1), which are incomprehensible, unstructured and very hard to analyse and work with (Aalst, 2016; Aalst, 2011); and 2) they produce models with unsatisfactory quality dimensions, be it poor fit of the log or ever-generalised model.

According to Tiwari, Turner & Majeed (2008), pioneering work in the area of the automated process discovery and process mining discipline, in general, was done by Agrawal, Gunopulos & Leymann (1998) and Cook & Wolf (1998) and their foundational approaches. Agrawal, Gunpulos & Leymann (1998) focused on mining models from workflow systems with the main focus on the appropriate ordering of

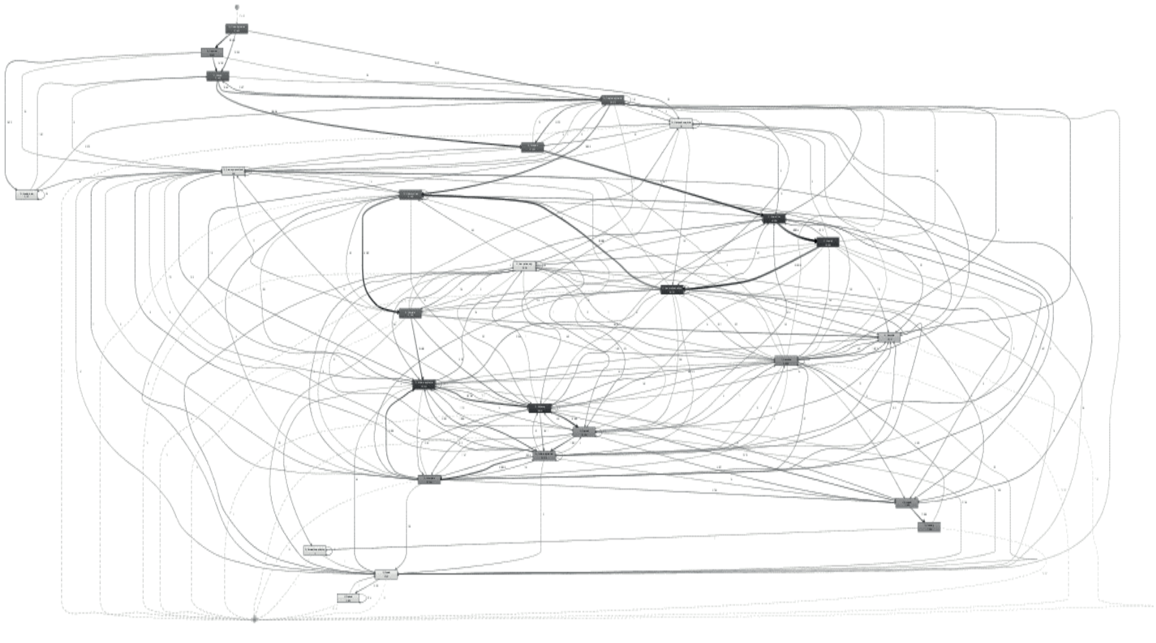

Fig. 1. Example of a spaghetti-like process model

activities and the successful termination of the process. Cook & Wolf (1998) described the application of Markov method within the process mining domain in addition to RNet and Ktail methods and evaluated the three proposed approaches to the automated process discovery: algorithmic, statistical and probabilistic. As was predicted by Cook & Wolf (1998), the most popular approach to the automated process discovery will be the algorithmic approach. This prediction turns out to be true and algorithmic approach to the automated process discovery is by far the most popular approach among researchers of the field (Tiwari, Turner & Majeed, 2008; Augusto et al., 2018).

One of the most influential techniques of the automated process discovery was introduced by Aalst, Weijters & Maruster (2003) and called α-algorithm. In their work, Aalst, Weijters & Maruster proved that α-algorithm is capable of discovering structured workflow-nets, which are an important class of Petri nets in the area of business processes, from complete event logs, assuming that they do not contain any noise. However, the original α-algorithm had several shortcomings in the form of short loops, invisible, duplicate or implicit tasks and non-free-choice constructs (Medeiros, Aalst & Weijters, 2003). Thus, the α-algorithm was extended several times. Firstly, Medeiros et al. (2005) introduced so-called α+-algorithm, so it was able to deal with short loops using the pre-processing of patterns specific to short loops. Next, Wen et al. (2007) and Wen, Wang & Sun (2006) introduced α++-algorithm that was able to detect non-free-choice constructs by considering a new relation called the implicit dependency. Wen et al. (2010) introduced α#-algorithm, capable of mining invisible tasks by considering the relation called the mendacious dependency. The latest version of an α-algorithm, so-called α$-algorithm was introduced by Guo et al. (2015). The algorithm uses improved mendacious and implicit dependency relations, and besides invisible and non-free-choice constructs, it is also able to mine invisible tasks in non-free-choice constructs.

HeuristicsMiner is another influential approach to the automated process discovery, which was introduced by Weijters, Aalst & Medeiros (2006). HeuristicsMiner was introduced to deal with noise and incompletion of event logs, where noise means events recorded in the log that are not supposed to be there and that do not represent the behaviour of the analysed process. On the other hand, an incomplete event log means missing data. It is an extension of α-algorithm in a sense that it considers frequencies, by which activity relationships occur (Aalst, Weijters & Medeiros, 2003) in the event log. In addition to robustness of an event log, HeuristicsMiner is also capable of dealing with short loops and non-local dependencies. Broucke & Weerdt (2017) introduced the discovery technique Fodina that is based on HeuristicsMiner and which handles the noise in the log and discover duplicate activities. Flexible Heuristics Miner (Weijters & Ribeiro, 2011) is yet another discovery technique based on HeristicsMiner. Similarly to previous techniques, Flexible Heuristics Miner can also deal well with noise in event logs.

In a series of papers, Leemans et al. (2013a, b; 2014) introduced the so-called inductive mining. Later versions focused on infrequency and incompleteness. Inductive mining produces process models in the form of process trees. The advantage of inductive mining is that it does provide guarantees in terms of soundness and re-discoverability of discovered process models. Leemans, Fahland & Aalst (2015; 2016) introduced the framework based on inductive mining that adds the advantage of scalability, while still guaranteeing the soundness and re-discoverability. Evolutionary Tree Miner introduced by Buijs, Dongen and Aalst (2012; 2014) belongs to the group of genetic algorithms and extracts process models from event logs in the form of a process tree.

Split Miner proposed by Augusto et al. (2017) is a technique with consistently high and balanced fitness, precision and generalisation that guarantees the deadlock-freedom for cyclic process models and the soundness for the acyclic. It merges an innovative approach to filter the directly-follows graph induced by an event log, with an approach to identify combinations of split gateways that accurately capture the concurrency, conflict and causal relations between neighbours in the directly-follows graph.

Günther & Aalst (2007) introduced Fuzzy Miner to tackle with unstructured processes. Fuzzy Miner is an adaptive simplification and visualisation technique based on significance and correlation measures to visualise the behaviour in event logs at various levels of abstraction (Weerdt et al., 2012). Previously mentioned algorithms use Petri nets as a representation of discovered process models. However, the discovered fuzzy model cannot be translated to the Petri net, which is a severe disadvantage to the Fuzzy Miner approach as it limits the comparability of Fuzzy Miner to other techniques. The same problem is characteristic of many more techniques.

Applying the genetic algorithm to process discovery, Medeiros, Weijters & Aalst (2007) introduced the so-called genetic process mining. An effort of the genetic process mining was to overcome problems related to non-free-choice constructs, and furthermore, invisible and duplicate task. Previously mentioned discovery techniques are limited by a local search, which is causing problems in discovering non-free-choice constructs or invisible and duplicate task. Thus, the global approach of genetic process mining comes into play, enabling the discovery of non-local behaviour (Weerdt et al., 2012). The advantage of the genetic process mining is that while most of other process discovery techniques focus only on one or two quality dimensions at the same time (Buijs, Dongen & Aalst, 2012), the genetic process mining can address all four quality dimensions.

Furthermore, several authors, for example, Werf et al. (2009), Aalst et al. (2010), based their automated discovery techniques on the principles of the theory of regions and integer linear programming (ILP). One of the main goals of automated discovery algorithms based on the theory of regions and ILP was to address the issue related to the assumption of completeness of the event log and the related problem of overfitting or underfitting the discovered process model by solving a series of ILPs. ILP was also used by Zelst et al. (2018) in their approach to automated process discovery. HybridILPMiner by Zelst et al. (2018)'s is based on the theory of regions and discovers relaxed sound workflow nets built on hybrid variable-based regions. Dongen & Aalst (2004) introduced the multi-phase process mining to mine instances of processes that can be later translated into other models, such as Petri nets of Event-driven Process Chains (EPCs). Correspondingly to techniques based on the theory of regions, the multi-phase process mining addresses the assumption of completeness of the log. The divide and conquer framework (Verbeek, Aalst & Munoz-Gama; 2017, Verbeek & Aalst, 2015) decomposes the process model discovery into smaller parts working with discovery techniques, and in this study, with ILP.

2. Methodology

The methodology section is divided into the following subsections: the first subsection describes the procedure of the acquisition of event logs from hybrid simulation models in AnyLogic framework. The second subsection describes business processes captured in a simulation model. The third subsection describes automated process discovery techniques used to benchmark and the use of metrics.

2.1. Procedure of the acquisition of event logs

To evaluate different automated process discovery techniques and assess their performance with ABS, a hybrid simulation was chosen from the AnyLogic framework (2019), based on which synthetic event logs were generated by replaying the process model. Hybrid simulation means that the simulation model consists of two or more approaches, meaning that the simulation model combines characteristics of two approaches, for example, ABS and discrete-event simulation, which was also used for the purpose of this study. The AnyLogic framework does not directly produce event logs needed for the process mining analysis. Thus, first, it is necessary to acquire such event logs. For this purpose, the BPMN 2.0 notation and the business process simulator (BIMP, 2019) were used. First, based on flowcharts and statecharts of business processes contained in the chosen hybrid simulation, models were transformed into BPMN process models. Then, the BPMN process models were simulated using the BIMP software that can produce an event log in the form of an MXML file. The BPMN notation is expressive enough to reproduce the control flow of a hybrid simulation model without any sacrifices (Fig. 2).

Furthermore, as stated in the Introduction section, the main advantages of ABS are the autonomy of agents, the complexity of the models etc. The autonomy of agents allows them to make decisions and, thus, determine the control-flow in particular process instances through such decisions. However, automated process discovery techniques are mainly focused on relations between occurring events and their sequence, and not necessarily on the reasons why the behaviour occurred. Thus, by expressing the behaviour of the modelled system using the BPMN notation, all the information relevant to automated process discovery techniques is preserved. While simulating the BPMN model in the business process simulator BIMP to acquire an event log, each event has to have a timestamp, so it can be ordered within the trace and, thus, processing times and arrival distributions of process tasks have to be defined for the purpose of generating an event log in the BIMP simulator. Where possible, parameters were used from the hybrid simulation model; otherwise, they were made artificially. However, it is necessary to keep in mind

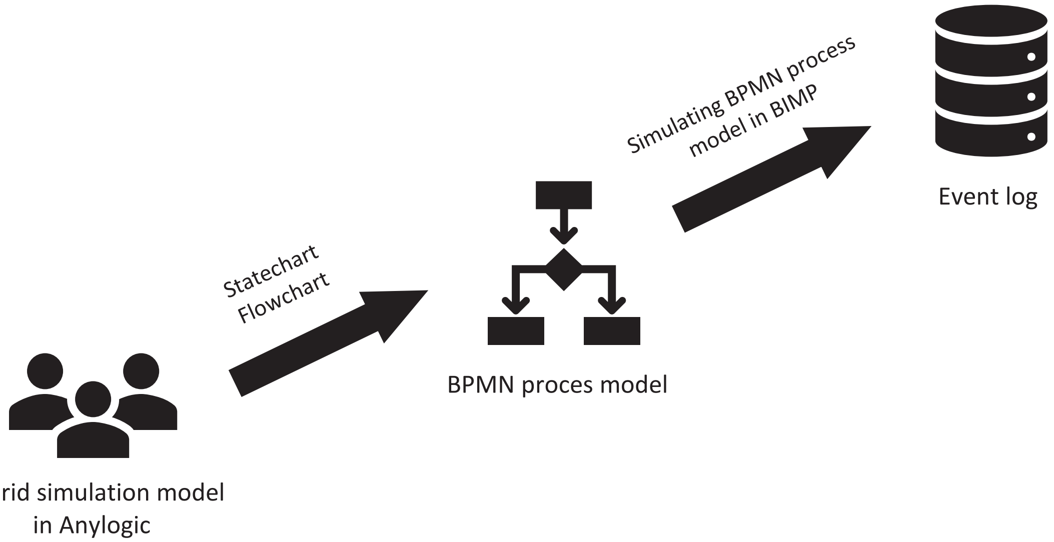

Fig. 2. Procedure of the acquisition of event logs from AnyLogic

that this does not pose a problem to the validity of the event log because the interest is not in the performance of the process itself and, thus, particular timestamps, but rather in the control flow of the process.

2.2. Description of a simulation model

A simulation model simulates the logistic process in a small job shop. Specifically, it is a logistics process describing the import of raw material, its storage, transformation into a product and its export. The overall process of the job shop simulation is as follows: the raw material is delivered to the receiving dock, where it is placed into storage until the processing occurs at the machine. Finished products are palletised and then moved to storage at s shipping dock until the completed pallets can be loaded on a truck.

The BPMN process model of the hybrid simulation is provided in Fig. 3 and is as follows: the start event in the business model is represented by the arrival of a truck with raw material. When the truck arrives at the docks, the system checks if the forklift is available. In the case that a forklift is available, it is assigned, pallets are unloaded from the truck and simultaneously assigned. In another case, the system automatically checks for an available forklift again until the forklift is assigned. After the pallets are unloaded, they are transferred into docks and stored. When the time comes, the pallets are assigned to particular machines for processing and transported to the assigned machines. This job used the second group of forklifts. Once the pallets are transferred to the machines, the raw material is processed. After the processing, the finished products are collected and put into storage. This processing part of the logistics process lies inside the big XOR gate. When the time comes, the system schedules a truck and finished products are prepared for export. When the truck arrives, the finished products are loaded, and the process ends when the loaded truck leaves. Fig. 4 basically represents the same model, but with 4 added

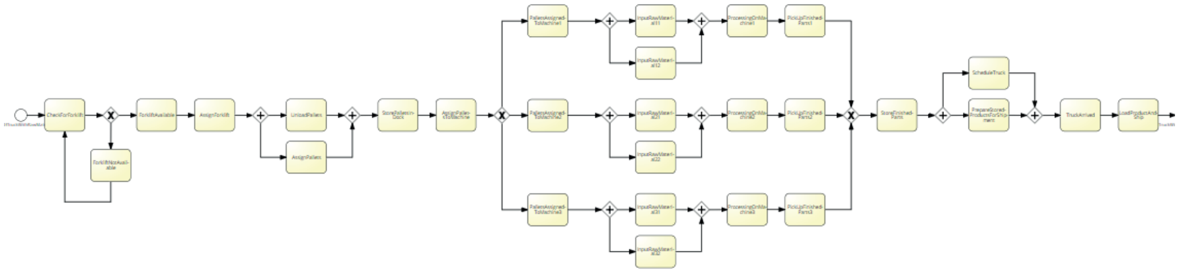

Fig. 3. BPMN process model of the hybrid simulation model

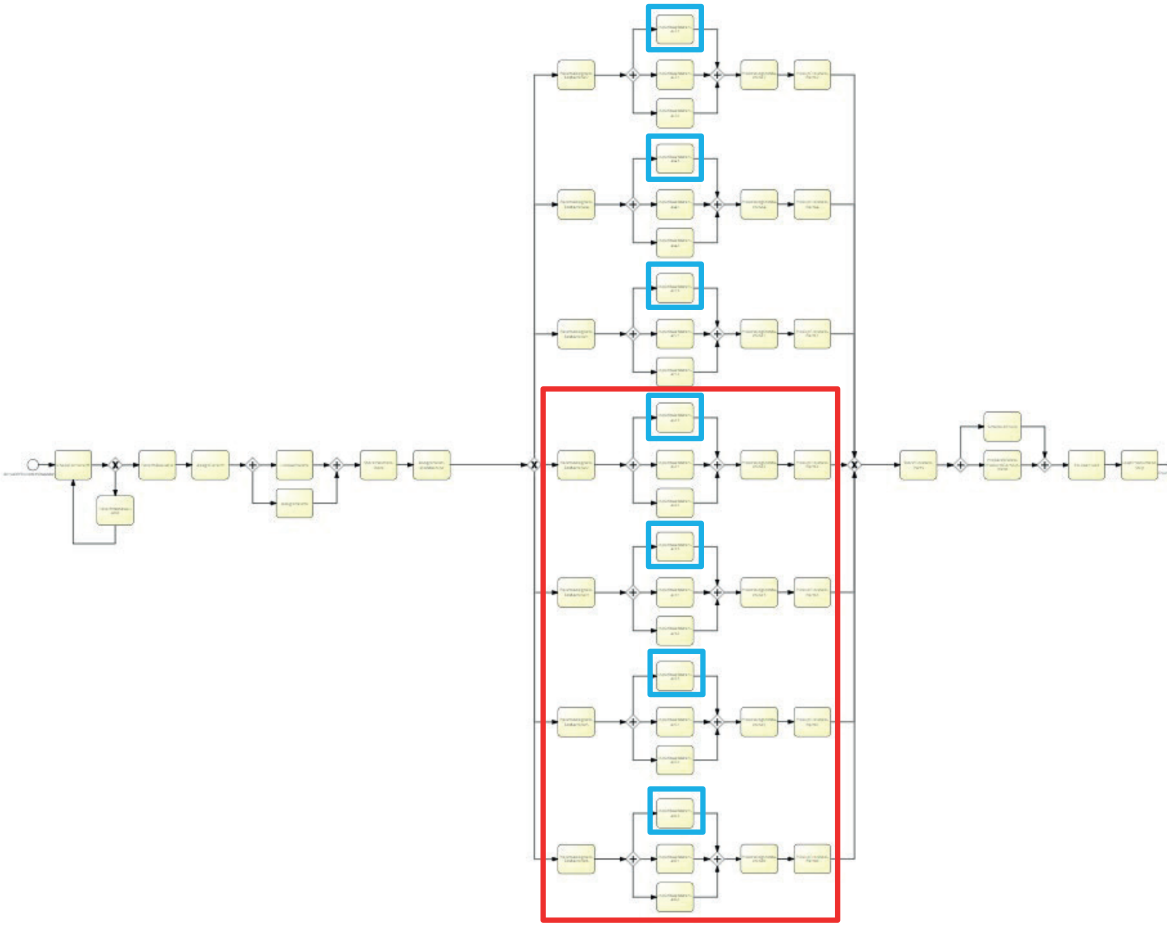

Fig.4. Extended BPMN process model of the hybrid simulation model

machines (red rectangle in Fig. 4) and one additional input of raw material (blue rectangles in Fig. 4) per each work line as parallel work lines are common in manufacturing. Finally, particular products made in the job shop were chosen as the case identifier for the simulation. The idea behind the case identifier being equal to a particular manufactured product is based on one of the pillars of Industry 4.0, where the product and even material are equipped with chips and thus, trackable in the cyber-physical environment.

Tab. 1. Selected automated process discovery techniques

Automated process discovery technique	Related studies
Structure HeuristicsMiner (sHM6)	Augusto et al. (2018)
Split Miner (SM)	Augusto et al. (2017)
Inductive Miner (IM)	Leemans et al. (2014)
Fodina (FO)	Broucke and Weerdt (2017)
α$	Guo et al. (2015)

2.3. Automated process discovery techniques and metrics

The focus regarding the evaluation of automated process discovery techniques is mainly on two previously mentioned quality dimensions: fitness and precision. Simply put, fitness measures the ability of the model to reproduce behaviour contained in the log. The range of the fitness function is the interval [0,1], where the value of fitness equal to 1 means that the process model can replay every trace in the event log. Precision, on the other hand, measures the ability of a model to generate the behaviour present in the event log. Similarly to fitness, the range of the precision function is the interval [0,1], where the value of precision equal to 1 means that any trace produced by the process model is found in the event log. Both quality dimensions can be combined into one index called the F-score, which is the harmonic mean of the two measures. For the purpose of this study, Markovian fitness and precision are used (Augusto et al., 2019).

It is necessary that the used modelling language has executable semantics so the quality dimension of

Tab. 2. Fitness and precision values for the process model depicted in Fig. 3 – 100 cases

Algorithm	Fitness	Precision	F-score	Soundness	Struct.
sHM6	1.0000	0.0763	0.1418	Sound	1.0000
SM	1.0000	0.0763	0.1418	Sound	1.0000
IM	1.0000	0.0763	0.1418	Sound	1.0000
FO	0.5918	0.0810	0.1425	Sound	1.0000
A$	1.0000	0.2269	0.3698	Sound	1.0000

Tab. 3. Fitness and precision values for the process model depicted in Fig. 3 – 8000 cases

Algorithm	Fitness	Precision	F-score	Soundness	Struct.
sHM6	1.0000	0.1002	0.1821	Sound	1.0000
SM	1.0000	0.1002	0.1821	Sound	1.0000
IM	1.0000	0.1002	0.1821	Sound	1.0000
FO	1.0000	0.1002	0.1821	Sound	1.0000
A$	1.0000	0.1002	0.1821	Sound	1.0000

Tab. 4. Fitness and precision values for the process model depicted in Fig. 4 – 100 cases

Algorithm	Fitness	Precision	F-score	Soundness	Struct.
sHM6	0.9488	0.0385	0.0739	Sound	1.0000
SM	-	-	-	-	-
IM	1.0000	0.0134	0.1775	Sound	1.0000
FO	0.7628	0.0130	0.0264	Sound	1.0000
A$	-	-	-	-	-

Tab. 5. Fitness and precision values for the process model depicted at Fig. 4 – 8000 cases

Algorithm	Fitness	Precision	F-score	Soundness	Struct.
sHM6	1.0000	0.1105	0.1990	Sound	1.0000
SM	-	-	-	-	-
IM	1.0000	0.1105	0.1990	Sound	1.0000
FO	1.0000	0.1105	0.1990	Sound	1.0000
A$	-	-	-	-	-

fitness and precision are computable. Petri nets are popular in many different areas of system modelling, while simultaneously having executable semantics. Furthermore, Petri nets are used by a relatively large number of automated process discovery techniques for representation of discovered process models. Thus, it is required that discovery techniques selected for the benchmark use Petri nets for the representation of the discovered process model. Also, those techniques were included that produce models which are convertible into Petri nets (Process Trees, BPMN models). Secondary criteria for the selection of the automated process discovery technique was the accessibility of the technique itself. The selected techniques can be found in Tab. 1.

3. Results

Tables 2–5 show benchmark results of BPMN process models depicted in Figs. 3–4. The evaluations were performed using the predefined parameters for particular process discovery techniques recommended by the developers of software packages. No same evaluations with optimised setting parameters of process discovery algorithms were done due to high demands on computing performance. Across all scenarios, discovered process models were sound and structured. According to Tables 2 and 3, all the process discovery algorithms perform well regarding the fitness quality measure with respect to the process model in Fig. 3. The exception is the Fodina discovery technique (FO), which performs relatively poorly in the simulation when the event log contains only 100 cases. However, it performs in the same way as the rest of the discovery algorithms, when there are 8000 cases in the event log. In the case of precision, all the discovery algorithms performed very poorly and thus, all the discovery algorithms also had a poor F-score.

Tables 4 and 5 represent the fitness quality of the process model from Fig. 4 and 8000 cases. sHM6, IM and FO performed well, again achieving the highest possible score. However, when the event log contained only 100 cases, the algorithms had a significantly lower performance regarding the fitness quality in two cases (sHM6 and FO). In the case of precision, all the discovery algorithms performed very poorly again and, thus, all the discovery algorithms also had a poor F-score. The comparison of the performance of particular discovery techniques listed in Tables 2 and 4 and then Tables 3 and 5, respectively, demonstrates that in the case of a simpler process model seen in Fig. 3, discovery algorithms performed better when using a smaller log, and in the case of a more complex model seen in Fig. 4, discovery algorithms performed better when using a bigger log. However, the comparison of the performance of particular discovery techniques listed in Tables 2 and 3 and then Tables 4 and 5, shows that process models with 100 cases have worse overall performance than process models with 8000 cases.

Conclusion and discussion

Based on the evaluation of benchmark results of chosen discovery algorithms, the discovery algorithms perform better overall with more extensive event logs (Tables 2 and 4, 3 and 5, respectively), which makes sense because the more information is contained in the event log, the better process models are produced by discovery techniques in general. However, on the other hand, the discovery algorithms that use less extensive event logs perform better, discovering less complex process models (Tables 2 and 3, 4 and 5, respectively). This also makes sense, because if the discovery algorithm has only limited information available in the log, the less complex models are more reliable to discover respecting the quality dimensions. At some point, Table 4 also shows that the decreasing values of precision have a negative influence on achieved values of fitness. The results above have a practical impact on the management of business processes, as, under the circumstances of Industry 4.0, it makes much more sense to consider adjusting the design of business processes to the available imperfect analytical tools.

It should be considered that one of the essential current problems of automated process discovery techniques is scalability due to a large amount of data that is generated and recorded by information systems and that has to be processed. However, as demonstrated, the process discovery techniques can also have problems of an opposite nature. This is especially true for companies with long delivery cycles, long processing times and parallel production, which are also common within industrial and related sectors. This is also amplified through the vertical and, later, on an even larger scale through horizontal integration of the supply chain within Industry 4.0. The impact of vertical integration in the BPMN model and the chosen case identifier is apparent, as otherwise without the assumption of smart manufacturing, we would not be able to use the single case identifier throughout the entire simulation. The entire process would need to be divided into several subprocesses. The management of business processes is nowadays essential for many companies to be competitive. However, with further progress of the Industry 4.0 concept, the analysis of business processes should be considered as a result of imperfect analytical methods and the emphasis of customers on effectiveness.

Acknowledgement

The work was supported by Project SGS/8/2018 project “Advanced methods and procedures of business process management” implemented by the Silesian University in Opava, Czechia.

Literature

van der Aalst, W. M. P. (2005). Business Alignment: Using Process Mining As a Tool for Delta Analysis and Conformance Testing. *Requirements Engineering*, *10*(3), 198-211. doi: 10.1007/s00766-005-0001-x van der Aalst, W. M. P. (2011). Process Mining: Discovery, Conformance and Enhancement of Business Processes. Berlin Heidelberg, Germany: Springer-Verlag.

van der Aalst, W. M. P. (2015). Extracting Event Data from Databases to Unleash Process Mining. In J. vom Brocke & T. Schmiedel (Eds.), *BPM - Driving Innovation in a Digital World* (pp. 105-128). Switzerland: Springer Cham (Management for Professionals).

van der Aalst, W. M. P. (2016). *Process Mining: Data Science in Action*. 2nd edn. Berlin Heidelberg, Germany: Springer-Verlag.

van der et al. Aalst, W. M. P. (2011). Process Mining Manifesto. In F. Daniel, K. Barkaoui, & S. Dustdar (Eds.), *Business Process Management Workshops. International Conference on Business Process Management* (pp. 169-194). Berlin, Heidelberg, Germany: Springer

van der Aalst, W. M. P., Rubin, V., Verbeek, H. M. V., van Dongen, B. F., Kindler, E., & Günther, C. W. (2010). Process mining: a two-step approach to balance between underfitting and overfitting. *Software & Systems Modeling*, *9*(1), 87-111. doi: 10.1007/s10270-008-0106-z

van der Aalst, W. M. P., Weijters, T., & Maruster, L. (2004). Workflow mining: discovering process models from event logs. *IEEE Transactions on Knowledge and Data Engineering*, *16*(9), 1128-1142. doi: 10.1109/TKDE.2004.47

van der Aalst, W. M. P., Weijters, T., & Maruster, L. (2004). Workflow mining: discovering process models from event logs. *IEEE Transactions on Knowledge and Data Engineering*, *16*(9), 1128-1142. doi: 10.1109/TKDE.2004.47

Abar, S., Theodoropoulos, G. K., Lemarinier P., & O'Hare, G. M. P. (2017). Agent Based Modelling and Simulation tools: A review of the state-of-art software. *Computer Science Review*, 24, 13-33. doi: 10.1016/j.cosrev.2017.03.001

Agrawal, R., Gunopulos, D., & Leymann, F. (1998). Mining Process Models from Workflow Logs. In H. Schek, F. Saltor, I. Ramos, & G. Alonso (Eds.), *Proceedings of the 6th International Conference on Extending Database Technology (EDBT'98), Lecture Notes in Computer Science, vol. 1377* (pp. 469-483), Berlin, Germany: Springer.

AnyLogic (2019). Simulation modelling software tool. Retrieved from https://www.anylogic.com

Augusto, A. Conforti, R., Dumas, M., La Rosa, M., Maggi, F. M., Marrella, A., Mecella, M., & Soo, A. (2017). *Automated Discovery of Process Models from Event Logs: Review and Benchmark*. Retrieved from http://arxiv.org/abs/1705.02288

Augusto, A., Conforti, R., Dumas, M., & La Rosa, M. (2017). Split Miner: Discovering Accurate and Simple Business Process Models from Event Logs. In *2017 IEEE International Conference on Data Mining (ICDM)* (pp. 1-10), New Orleans, United States: IEEE.

Augusto, A., Conforti, R., Dumas, M., La Rosa, M., & Bruno, G. (2018). Automated discovery of structured process models from event logs: The discover-and-structure approach. *Data & Knowledge Engineering*, 117, 373-392. doi: 10.1016/j.datak.2018.04.007

Augusto, A., et al. (2019). *Measuring Fitness and Precision of Automatically Discovered Process Models: A Principled and Scalable Approach*. Retrieved from https://minerva-access.unimelb.edu.au/bitstream/handle/11343/219723/main.pdf

Bannat, A., et al. (2011). Artificial Cognition in Production Systems. *IEEE Transactions on Automation Science and Engineering*, *8*(1), 148-174. doi: 10.1109/TASE.2010.2053534

BIMP (2019). Business Process Simulator. Retrieved from http://bimp.cs.ut.ee

Boes, J., & Migeon, F. (2017). Self-organizing multi-agent systems for the control of complex systems. *Journal of Systems and Software*, 134, 12-28. doi: 10.1016/j.jss.2017.08.038

Borshchev, A., & Filippov, A. (2004). *From System Dynamics and Discrete Event to Practical Agent Based Modeling: Reasons, Techniques, Tools*. Retrieved from https://www.researchgate.net/publication/233820565_From_System_Dynamics_and_Discrete_Event_to_Practical_Agent_Based_Modeling_Reasons_Techniques_Tools

van den Broucke, S. K. L. M., & De Weerdt, J. (2017). Fodina: A robust and flexible heuristic process discovery technique. *Decision Support Systems*, 100, 109-118. doi: 10.1016/j.dss.2017.04.005

Buijs, J. C. A. M., van Dongen, B. F., & van der Aalst, W. M. P. (2012). On the Role of Fitness, Precision, Generalization and Simplicity in Process Discovery. In Meersman, et al. (Eds.), *On the Move to Meaningful Internet Systems: OTM 2012. OTM Confederated International Conferences* (pp. 305-322). Berlin, Heidelberg, Germany: Springer. doi: 10.1007/978-3-642-33606-5_19

Buijs, J. C. A. M., Dongen, B. F. van, & Aalst, W. M. P. van der (2012). A genetic algorithm for discovering process trees. In *2012 IEEE Congress on Evolutionary Computation* (pp. 1-8). Brisbane, Australia: IEEE. doi: 10.1109/CEC.2012.6256458

Buijs, J. C. a. M., van Dongen, B. F., & van der Aalst, W. M. P. (2014) Quality Dimensions in Process Discovery: The Importance of Fitness, Precision, Generalization and Simplicity. *International Journal of Cooperative Information Systems*, *23*(01), 1440001. doi: 10.1142/S0218843014400012

Buijs, J. C. A. M., van Dongen, B. F., & van der Aalst, W. M. P. (2014). Quality Dimensions in Process Discovery: The Importance of Fitness, Precision, Generalization and Simplicity. *International Journal of Cooperative Information Systems*, *23*(1), p. 1440001. doi: 10.1142/S0218843014400012

Chan, W. K. V., Son, Y. J., & Macal, C. M. (2010). Agent-based simulation tutorial - simulation of emergent behavior and differences between agent-based simulation and discrete-event simulation. In *Proceedings of the 2010 Winter Simulation Conference* (pp. 135-150). Baltimore, United States: IEEE. doi: 10.1109/WSC.2010.5679168

Claes, D., Oliehoek, F., Baier, H., & Tuyls, K. (2017). Decentralised Online Planning for Multi-Robot Warehouse Commissioning. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems.*

Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems (AAMAS '17) (pp. 492-500). Richalnd, United States: SC.

Cook, J. E., & Wolf, E. L. (1998). Discovering Models of Software Processes from Event-Based Data. *ACM Transactions on Software Engineering and Methodology*, *7*, 215-249.

De Weerdt, J., De Backer, M., Vanthienen, J., & Baesens, B. (2012). A multi-dimensional quality assessment of state-of-the-art process discovery algorithms using real-life event logs. *Information Systems*, *37*(7), 654-676. doi: 10.1016/j.is.2012.02.004

Dinardo, G., Fabbiano, L., & Vacca, G. (2018). A smart and intuitive machine condition monitoring in the Industry 4.0 scenario. *Measurement*, 126, 1-12. doi: 10.1016/j.measurement.2018.05.041

van Dongen, B. F., & van der Aalst, W. M. P. (2004). Multi-phase Process Mining: Building Instance Graphs. In P. Atzeni, et al. (Eds.). *Conceptual Modeling – ER 2004. International Conference on Conceptual Modeling* (pp. 362-376). Berlin Heidelberg, Germany: Springer. doi: 10.1007/978-3-540-30464-7_29

Doomun, R., & Vunka Jungum, N. (2008). Business process modelling, simulation and reengineering: call centres. *Business Process Management Journal*, *14*(6), 838-848.

Goedertier, S., Martens, D., Vanthiene, J., Baesens, B. (2009). Robust Process Discovery with Artificial Negative Events. *Journal of Machine Learning Research*, *10*, 1305-1340.

Gries, M., Kulkarni, C., Sauer, C., & Keutzer, K. (2003). Comparing analytical modeling with simulation for network processors: a case study. In *Automation and Test in Europe Conference and Exhibition 2003 Design* (pp. 256-261). Munich, Germany: IEEE. doi: 10.1109/DATE.2003.1253838

Günther, C. W., & van der Aalst, W. M. P. (2007). Fuzzy Mining – Adaptive Process Simplification Based on Multi-perspective Metrics. In G. Alonso, P. Dadam, & M. Rosemann, (Eds.), *Business Process Management* (pp. 328-343). Berlin Heidelberg, Germany: Springer.

Guo, Q., Wen, L., Wang, Z., & Yu, P. S. (2015). Mining Invisible Tasks in Non-free-choice Constructs. In H. R. Motahari-Nezhad, J. Recker, & M. Weidlich, (Eds.). *Business Process Management* (pp. 109-125). Cham, Germany: Springer International Publishing.

Hlupić, V., & Vukšić, V. B. (2004). Business Process Modelling Using SIMUL8. Retrieved from https://www.researchgate.net/publication/254419366_BUSINESS_PROCESS_MODELLING_USING_SIMUL8

Hsieh, F.-S. (2015). Scheduling Sustainable Supply Chains based on Multi-agent Systems and Workflow Models. In *2015 10th International Conference on Intelligent Systems and Knowledge Engineering* (pp. 252-259). New York, United States: IEEE.

Kelly, R. A., et al. (2013). Selecting among five common modelling approaches for integrated environmental assessment and management. *Environmental Modelling & Software*, *47*, 159-181. doi: 10.1016/j.envsoft.2013.05.005

Kolberg, D., & Zühlke, D. (2015). Lean Automation enabled by Industry 4.0 Technologies. *IFAC-PapersOnLine*, *48*(3), 1870-1875. doi: 10.1016/j.ifacol.2015.06.359

Kozma, T. (2017). Cooperation in the supply chain network. *Forum Scientiae Oeconomia*, *5*(3), 45-58.

Leemans, S. J. J., Fahland, D., & van der Aalst, W. M. P. (2013a). Discovering Block-Structured Process Models from Event Logs - A Constructive Approach. In J.-M. Colom, & J. Desel, (Eds.), *Application and Theory of Petri Nets and Concurrency. International Conference on Applications and Theory of Petri Nets and Concurrency* (pp. 311-329). Berlin Heidelberg, Germany: Springer. doi: 10.1007/978-3-642-38697-8_17

Leemans, S. J. J., Fahland, D., & van der Aalst, W. M. P. (2013b). Discovering Block-Structured Process Models from Event Logs Containing Infrequent Behaviour. In N. Lohmann, M. Song, & P., Wohed (Eds.), *Business Process Management Workshops. International Conference on Business Process Management* (pp. 66-78). Berlin, Germany: Springer International Publishing. doi: 10.1007/978-3-319-06257-0_6

Leemans, S. J. J., Fahland, D., & van der Aalst, W. M. P. (2014). Discovering Block-Structured Process Models from Incomplete Event Logs. In G. Ciardo, & E. Kindler (Eds.), *Application and Theory of Petri Nets and Concurrency. International Conference on Applications and Theory of Petri Nets and Concurrency* (pp. 91-110). Berlin, Germany: Springer International Publishing. doi: 10.1007/978-3-319-07734-5_6

Leemans, S. J. J., Fahland, D., & van der Aalst, W. M. P. (2015). Scalable Process Discovery with Guarantees. In K. Gaaloul, et al. (Eds.), *Enterprise, Business-Process and Information Systems Modeling. International Conference on Enterprise* (pp. 85-101). Berlin, Germany: Springer International Publishing. doi: 10.1007/978-3-319-19237-6_6

Leemans, S. J. J., Fahland, D., & van der Aalst, W. M. P. (2018). Scalable process discovery and conformance checking. *Software & Systems Modeling*, *17*(2), 599-631. doi: 10.1007/s10270-016-0545-x

Leitão, P., et al. (2016). Smart Agents in Industrial Cyber–Physical Systems. *Proceedings of the IEEE*, *104*(5), 1086-1101. doi: 10.1109/JPROC.2016.2521931

Macal, C. M. (2010). To Agent-based Simulation from System Dynamics. In *Proceedings of the Winter Simulation Conference* (pp. 371-382). Baltimore, Maryland: WSC.

Macal, C. M., & North, M. J. (2008). Agent-based Modeling and Simulation: ABMS Examples. In *Proceedings of the 40th Conference on Winter Simulation* (pp. 101-112). Miami, Florida, United States: WSC.

de Medeiros, A. K. A., van Dongen, B. F., van der Aalst, W. M. P., & Weijters, A. J. M. M. (2005). *Process Mining: Extending the α-algorithm to Mine Short Loops*. Retrieved from https://pdfs.semanticscholar.org/dd4b/bc6f1550fc6601b21bd83f5c5ff3b13a309d.pdf

de Medeiros, A. K. A., Weijters, A. J. M. M., & van der Aalst, W. M. P. (2007). Genetic process mining: an experimental evaluation. *Data Mining and Knowledge Discovery*, *14*(2), 245-304. doi: 10.1007/s10618-006-0061-7

Nguyen, H., et al. (2016). *Business Process Deviance Mining: Review and Evaluation*. Retrieved from http://arxiv.org/abs/1608.08252

Pan, M., et al. (2015). Applying Industry 4.0 to the Jurong Island Eco-industrial Park. *Energy Procedia*, *75*, 1536-1541. doi: 10.1016/j.egypro.2015.07.313

Piccarozzi, M., Aquilani, B., & Gatti, C. (2018). Industry 4.0 in Management Studies: A Systematic Literature Review. *Sustainability*, *10*(10), 3821. doi: 10.3390/su10103821

Pisching, M. A., et al. (2018). An architecture based on RAMI 4.0 to discover equipment to process operations required by products. *Computers & Industrial Engineering*, *125*, 574-591. doi: 10.1016/j.cie.2017.12.029

Pomarlan, M., & Bateman, J. (2018). Robot Program Construction via Grounded Natural Language Semantics & Simulation. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems* (pp. 857-864). Richland, United States: International Foundation for Autonomous Agents and Multiagent Systems (AAMAS '18).

Qin, J., Liu, Y., & Grosvenor, R. (2016). A Categorical Framework of Manufacturing for Industry 4.0 and Beyond. *Procedia CIRP*, *52*, 173-178. doi: 10.1016/j.procir.2016.08.005

Roblek, V., Meško, M., & Krapež, A. (2016). A Complex View of Industry 4.0. *SAGE Open*, *6*(2), 1-11. doi: 10.1177/2158244016653987

Rodič, B. (2017). Industry 4.0 and the New Simulation Modelling Paradigm. *Organizacija*, *50*(3), 193-207. doi: 10.1515/orga-2017-0017

Savaglio, C., et al. (2018). Agent-Based Computing in the Internet of Things: A Survey. In M. Ivanović, et al. (Eds.), *Intelligent Distributed Computing XI* (pp. 307-320), Cham, Germany: Springer International Publishing. doi: 10.1007/978-3-319-66379-1_27

Siebers, P. O., et al. (2010). Discrete-event simulation is dead, long live agent-based simulation!. *Journal of Simulation*, *4*(3), 204-210. doi: 10.1057/jos.2010.14

Ślusarczyk, B. (2018). Industry 4.0 : are we ready?, *Polish Journal of Management Studies*, *17*(1), 232-248. doi: 10.17512/pjms.2018.17.1.19

Sony, M. (2018). Industry 4.0 and lean management: a proposed integration model and research propositions. *Production & Manufacturing Research*, *6*(1), 416-432. doi: 10.1080/21693277.2018.1540949

Tiwari, A., Turner, C. J., & Majeed, B. (2008). A review of business process mining: State-of-the-art and future trends. *ResearchGate*, 14(1), 5-22. doi: 10.1108/14637150810849373

Verbeek, H. M. W., & van der Aalst, W. M. P. (2015). Decomposed Process Mining: The ILP Case. In F. Fournier, & J. Mendling, (Eds.), *Business Process Management Workshops* (pp. 264-276). Berlin: Springer International Publishing.

Verbeek, H. M. W., van der Aalst, W. M. P., & Munoz-Gama, J. (2017). Divide and Conquer: A Tool Framework for Supporting Decomposed Discovery in Process Mining. *The Computer Journal*, *60*(11), 1649-1674. doi: 10.1093/comjnl/bxx040

Wan, J., et al. (2018). Toward Dynamic Resources Management for IoT-Based Manufacturing. *IEEE Communications Magazine*, *56*(2), 52-59. doi: 10.1109/MCOM.2018.1700629

Wang, S., et al. (2016). Towards smart factory for industry 4.0: a self-organized multi-agent system with big data based feedback and coordination. *Computer Networks*, *101*, 158-168. doi: 10.1016/j.comnet.2015.12.017

Weijters, A. J. M. M., & Ribeiro, J. T. S. (2011). Flexible Heuristics Miner (FHM). In *2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)* (pp. 310-317). Paris, France: IEEE. doi: 10.1109/CIDM.2011.5949453

Weijters, A. J. M. M., van der Aalst, W.M.P., & Medeiros, A. K. A. D. (2006). Process Mining with the Heuristics-Miner Algorithm. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.118.8288&rep=rep1&type=pdf

Wen, L., et al. (2007). Mining process models with non-free-choice constructs. *Data Mining and Knowledge Discovery*, *15*(2), 145-180. doi: 10.1007/s10618-007-0065-y

Wen, L., Wang, J., & Sun, J. (2006). Detecting Implicit Dependencies Between Tasks from Event Logs. In *Frontiers of WWW Research and Development - APWeb 2006. Asia-Pacific Web Conference* (pp. 591-603), Berlin, Heidelberg, Germany: Springer. doi: 10.1007/11610113_52

van der Werf, J. M. E. M., van Dongen, B. F., Hurkens, C.J., & Serebrenik, A. (2009). Process Discovery using Integer Linear Programming. Fundamenta Informaticae, *94*(3-4), 387-412. doi: 10.3233/FI-2009-136

van Zelst, S. J., et al. (2018). Discovering workflow nets using integer linear programming. *Computing*, *100*(5), 529–556. doi: 10.1007/s00607-017-0582-5