# Using process mining and event log analysis for better business strategy decision-making

Yuri van Midden University of Twente P.O. Box 217, 7500AE Enschede The Netherlands j.h.vanmidden@student.utwente.nl

# ABSTRACT

Process mining is often used for conformance checking and performance checking in businesses. However, after establishing the problems, seeking solutions and deploying them, that is often the end of the process, whilst event logs analysis can also be used to yield business performance. This paper proposes methods to utilise process mining and event logs analysis to aid business decision makers in narrowing down the considered business options, and yield business performance. After defining how event logs analysis can be used to execute performance checking and thus narrowing down business options, a case study is conducted wherein the methods are tested. Three experiments are conducted in which at the start of the event logs all possible scenarios are considered, and after a given interval the worst performing scenarios are eliminated. The experiments show that the proposed methods of narrowing down business scenarios all converge to similar best-performing scenarios, and that event logs analysis has great potential to aid business decision makers.

#### Keywords

process mining, event log analysis, business strategy, logistics, simulations

# 1. INTRODUCTION

For business decision makers, it can be hard to make strategical decisions that improve the business and often there are time and money constraints that restrict testing and evaluating all possible business scenarios. As resources, time and money are limited, there is a call to make decisions that promise most business performance.

Process mining is an emerging discipline relating to data science and BPM discovery, aiming to support in the field of business process analysis and management[5]. Often process mining is used to do performance and conformance checking in a business, to answer questions like: "Why does it take so long to get a product from A to B?" or "What part of a process takes a lot of time?"

Even though process mining provides answers to a lot of these questions, process mining is often used a as goal in itself, instead of as a means to improve business decision-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

35<sup>th</sup> Twente Student Conference on IT July. 2<sup>nd</sup>, 2021, Enschede, The Netherlands.

Copyright 2021, University of Twente, Faculty of Electrical Engineering, Mathematics and Computer Science. making and strategy. Furthermore, process mining and event logs analysis are scarcely used as tools for business decision-makers to evaluate different scenarios whilst the business is operating. Usually event logs are considered as a whole, instead of narrowed down and evaluated to smaller time spans.

Hence the aim of this research is to propose a method to help business decision-makers in determining a strategy for effectively narrowing down the number of considered business scenarios. The narrowing down of business scenarios should be time efficient and substantiated by statistical arguments. We propose an approach to steer in selecting scenarios that are expected to yield the most promising business performance. This is achieved by evaluating event logs of different scenarios over given time spans, after which - at given intervals - a scenario-selection is made to continue for evaluation.

To achieve this goal, the following research questions are answered:

- 1. What are suitable performance indicators in a business in the logistical domain?
- 2. How can event logs be used to derive quality measures?
- 3. How can event log analysis be used to narrow down possible business scenarios to comply with the requirements of the performance indicators?

The answers to the research questions will be applied to a case study as a means of validation.

The remainder of this document is structured as follows: first some background information on event logs and process mining will be provided. Thereafter the research questions will be answered, followed up by the case study that integrates the previously accumulated findings. Lastly, the results will be presented and a conclusion will be drawn.

## 2. BACKGROUND

#### 2.1 Event logs

Events in the context of *event logs* and *process mining* are actions that are recorded in a log. They typically consist of data like the start time, completion time, activity description, allocated resources, cost and a case ID [6]. For accurate process mining it is important that events always consist of a case ID and a timestamp, in order to be able to do performance and conformance checking. A brief example of an event log can be seen in table 1. An ingredient, milk, comes into a factory and is being moved around a forklift vehicle. The sequence of such actions combine into an event log. Lastly, event logs do not need

ingroutonos, when quanty measures					
timestamp	ingredientID and quality	event	resource		
9:00:00	milk1:100%	enter factory	Forklift A		
9:01:25	milk1:98%	picked up	Forklift B		
9:03:02	milk1:93%	dropped off	Forklift B		
9:03:03	milk1:93%	start processing	Machine A		

Table 1. Example event log for a factory processing fresh ingredients, with quality measures



Figure 1. An example Petri net, adopted from [7] (page 27)

to origin from a single source, they can be a combination of multiple prepared data sets.

## 2.2 Process Mining

With the event logs at hand, process mining can be used for process discovery, conformance checking and performance analysis [6]. With process mining, it is possible to identify *event traces* from event logs, and create a model from the discovered traces. An example of such a mined process model (i.e. Petri net) can be found in figure 1.

Process mining is intended to answer several questions in the context of a given business. Typically these questions are divided into two categories: *performance questions* and *conformance questions* [7]. Performance questions may be "Why are these products always late?" or "What resources are overloaded?" whilst conformance questions can be "Which part in the process is often skipped?" or "What business rules are often violated?" Using the event logs and the discovered Petri net model, an event log replay on the Petri net can be executed to monitor the performance of the events on the given traces. This also enables for evaluating what strategies result in business rules being violated.

We determine the following perspectives: conformance checking perspective, time & case perspective, organizational perspective and the data perspective.

#### 2.2.1 Conformance checking

Conformance checking is divided info four core principles, according to Van der Aalst: *fitness, simplicity, precision* and *generalization* [5].

We can determine the *fitness* of an event trace. A process model has a perfect fitness if all traces in the log can be replayed by the model from beginning to end [5]. We can define fitness by looking at the case (for example, *milk1* from the example in table 1), and determine whether all events in the case exist in the process model from the beginning to the end. Another way of determining fitness is by checking if an event is represented in the process model at all. If all these analysis turn out to be true, then the fitness is 1. If none of the cases are represented by the model, the fitness is 0. Second the *simplicity* of a model can be checked. Van der Aalst [7] describes the method of *Occam's Razor*, which comes down to that the simplest representation of the behaviour of the event logs is probably the best one.

Third, *precision* is related to fitness. The model is precise when it does not allow for more types of behaviour then can be imposed by the (potential) events.

Last, *generalization* is opposite to precision. When a model is too specific, it may fit for some specific event logs, but not be able to fit other event logs. Therefore a model should be generalized enough.

For businesses in the logistics domain, we can determine several threshold values that trigger conformance monitoring. For instance, we often hear "a 99% success rate", or "99.9% success rate", which can be translated into conformance checking threshold values. In this example the fitness of the event traces should always be above 99 or 99.9, and if the threshold is exceeded (i.e. value drops below 99), the business scenario may be altered to yield better performance again.

#### 2.2.2 Case & time perspective

Second the case-and-time perspective is related to performance checking in process mining. Executing performance analysis in tools like ProM [8], aids in determining the time for cases to get from the beginning to the end in the model, as well as some basic statistics like the mean, median, standard deviation, minimum and maximal time for some or all cases.

For businesses in the logistics domain, performance indicators such as the *mean* of the clearing times of all cases or the *mean* + *standard deviation* of the clearing times of all cases.

## 2.2.3 Organizational perspective

Thirdly, key performance indicators (KPIs) are considered, as they represent the means of the company in order to achieve their goals. For example, if a product at the end of the production line must have a "quality value" of at least 90%, it can be argued that this value should be a threshold for elimination business scenario options.

#### 2.2.4 Data perspective

Lastly, the data perspective is considered, which is most important for quality measures in this case. The event logs often contain a lot of hidden data that can be used for performance analysis, one being the case & time indicator. An other factor one can imagine is whether customers need to go through a lot of repetitive processes, making the amount of repetitions a performance indicator. Lastly, it is possible to infer data from the event logs, and use this data for performance measures. For example, if a cost attribute is included in the event logs, one can classify the costs as *high*, *medium* or *low*, which can be evaluated in the process discovery phase as separate processes. This classifier method can aid in enriching the existing data sets.

# 3. RESEARCH QUESTIONS

This research aims to describe methods to narrow down the number of considered business scenario options, using process mining and event log analysis. These methods will thereafter be applied to a case study. First, the research questions mentioned in the introduction will be answered in this section:

1. What are suitable performance indicators in a busi-

ness in the logistical domain?

- 2. How can event logs be used to derive quality measures?
- 3. How can event log analysis be used to narrow down possible business scenarios to comply with the requirements of the performance indicators?

## **3.1** Performance indicators

This research will mainly focus on processing event logs in a logistical context, therefore we need to consider possible actors and their interests in the business. Furthermore we will analyse what Key Performance Indicators are essential for the operations. These will form a basis upon which we can carry out our case study.

There are several performance indicators that can be considered in a logistical context. For example, a CEO wants to keep the waiting times for a delivery as low as possible, while maintaining low costs, and by using as little personnel as possible. For answering this research question, we want to determine what the indicators are that make us switch to a different strategy if needed.

Let us consider a factory that processes food. The factory uses several fresh ingredients as an input and produces a final product that is shipped to grocery stores. The input ingredients are likely to show a quality decay over time and therefore we define a few performance indicators, such as ingredient quality at the start, processing time, amount of personnel and vehicles needed to transport the ingredients through the plant. Furthermore the quality of the end product should be high enough and the transport to the grocery store should go as smooth as possible. The key performance indicators in this example can be formulated as follows:

- The time from raw ingredients to finished product does not exceed 3 hours.
- The temperature of the ingredient throughout the process should not exceed 20 degrees centigrade.
- The waiting time from finished product to transport to the grocery store should not exceed 2 days.
- $\bullet\,$  The quality decline of the ingredients should not exceed  $20\%\,$

If we take a look at the event logs a factory described above may generate, we can imagine a few data points that are included in the event logs, like product quality, the timestamp, the unique identifier of the ingredient, the resource that is handling the ingredient. An example can be found in table 1.

It is not hard to see that some of these data points in this table are suitable for quality and performance checking. For instance, the time it took for a forklift vehicle to transport the goods can be a quality measure, or the quality decay of the ingredients themselves.

Krauth et al.[3] have created a framework that indicate suitable performance indicators for logistical service providers. They divided their framework in internal and external KPIs, where internal means that the KPIs are relevant for the company's management and employees, and external means the customer and society as a whole. Internal KPIs for managerial use are for example *number of deliveries, trips per period, average fuel use per km, % of failed orders* and *human resource costs.* 

#### 3.1.1 Process mining and event log analysis

Using process mining and a continuous analysis of the event logs, the aforementioned KPI's success rate can be monitored and evaluated. Even more, when a business scenario does not yield much performance and the KPI cannot be met, event logs analysis can help uncover this and eliminate such scenarios from the set of considered options.

#### **3.2** Using event logs for quality measures

Event logs are capable of holding other data fields than just the case ID, timestamp and action, such as a temperature measure, or a quality attribute. As Mannhardt et al. [4] state, these attributes are often not used in process discovery, and therefore result in unreliable quality diagnostics for the discovered models. Therefore, methods are sought to perform analysis based on these quality measures.

One method described before, is the time & case performance indicator where individual event traces are classified based on the case times. A method to classify these case times is by using the mean case time and the standard deviation to determine the human-readable score of the cases. An example can be found in table 2. These performance classes can be combined with the events in the event logs, which enables for performance checking using process mining (i.e. how often does a trace go from 'good' to 'insufficient'?).

Table 2. Basic classifier for performance values

value	performance class
$<\mu-2\sigma$	very bad
$< \mu - \sigma$	bad
$<\mu$	insufficient
$< \mu + \sigma$	sufficient
$<\mu+2\sigma$	good
$> \mu + 2\sigma$	$very \ good$

## 3.3 Narrowing down business scenarios

In this section methods are proposed to narrow down business scenarios for decision-makers, using event logs analysis. Possible scenarios one can think of are combinations of resource allocation (personnel and vehicles), investments and costs, KPIs, machinery, and so forth. First, the lifecycle of a typical data mining project will be discussed using the CRISP-DM model [9].

#### 3.3.1 CRISP-DM

Let us consider the CRISP data mining model, in which one considers the life cycle of a data mining project in a given business. The model is portrayed in figure 2. Summarised, the model tells us how a data mining project should be assessed: we typically start with Business Understanding, where we aim to figure out what business intelligence we need in order to solve a problem. From there we proceed to Data Understanding, where we collect data and get familiar with the available data. Next, we process the data in such a way that we can get a final and usable data set from the raw data, the Data Preparation phase. Next we start the Modelling phase, which essentially means we tweak the data sets and calibrate it to optimal values, in order to ensure we can tackle the data mining project. In the Evaluation phase we ensure that the obtained model is exactly what is need to tackle the data mining problem, and we come to a final decision in whether we deploy any solution in the business. As can



Figure 2. "Phases of the CRISP-DM Process Model for Data Mining." Derived from [9]

be seen in the figure (2), some phases can be executed in any desired order.

#### 3.3.2 Methods for narrowing down business scenarios

In order to determine the best performing business scenarios, several methods can be applied when analysing event logs. The methods effectively touch upon every aspect of the CRISP-DM model. In order to define certain goals, business understanding and data understanding are needed. In order to define our strategy for narrowing down business scenarios, accurate data preparation and modelling are needed.

The aim is to define methods that narrow down business scenarios, so that the most promising business scenarios remain. For example, let us consider a set of event logs (n=100) of a factory of different periods in time with altering business scenarios. One is able to play through these event logs simultaneously and evaluate the conformance and performance after every day, or every hour, or even at any given moment in time. The conformance and performance can be determined using process mining and basic statistics, and thus more promising business scenarios can be derived.

For this research, various event log replay strategies have been defined. For example: "Play through the event logs simultaneously and after one hour, ..."

- "...discard half of all the scenarios that performed worst. Then repeat with the remaining logs."
- "...discard the single worst performing scenario. Then repeat with the remaining logs."
- "...calculate the mean *score* of all scenarios and discard every scenario that performed worst than the mean value. Then repeat with the remaining logs."
- "...discard any scenario that has a lower score than the *mean - standarddeviation*. Then repeat with the remaining logs."

• ...

These findings will be put into practise in the case study below.

# 4. CASE STUDY

The case study is intended to test the findings of the research questions. The case study is building upon previous research by Bemthuis, R. [1] where a conceptual agentbased simulation framework is proposed, to analyse and learn from emergent behaviour in complex business situations. The simulations that were run involved a factory that processes certain products, which were moved around by various types of vehicles: human-driven forklifts (HDF), automated guided vehicles (AGV) and unmanned aerial vehicles (UAV), visualised in figure 3. An overview of the factory is depicted in figure 4. We can see the factory is partitioned into region 1, region 2 and region 3. Typically, during the simulations the products were transported from region 1 to region 2 to region 3.

Certain dispatching rules were imposed on the simulations, distinguished by so-called vehicle-initiated and productinitiated rules. The vehicle-initiated rules were: random, pick lowest product quality decay, pick highest product quality decay. The product-initiated rules were: random, call lowest utilisation, call shortest travel distance. Furthermore simulations were run with altering compositions of the vehicles.

All permutations of the different scenarios run in the simulations, resulted in data sets of 27 different scenarios on which we conduct our experiments. The scenarios can be viewed in Table 3. The scenarios are divided into three resource allocations:

- Scenarios 1-9: 3 UAVs, 1 HDF, 1 AGV
- Scenarios 10-18: 3 UAVs, 2 HDFs, 2 AGVs
- Scenarios 19-27: 2 HDFs, 2 AGVs

Table 3. Scenarios simulated in the data sets				
	random	lowest	highest	
		quality	quality	
random	$\{1,10,19\}$	$\{8,17,26\}$	$\{9,18,27\}$	
lowestUtilization	$\{2,11,20\}$	$\{4,13,22\}$	$\{6,15,24\}$	
shortestTravelDist.	$\{3,12,21\}$	$\{5,14,23\}$	$\{7, 16, 25\}$	

The final goal of the case study is to narrow down the possible business scenarios whilst analysing the event logs, and yield the supposedly most promising scenario for a business decision maker. As a performance indicator the quality of the products when dropped of in *region* 3 is used.

# 4.1 Methodology

The data sets, obtained from [2] include event logs of all different scenarios in table 3. Using the knowledge accounted for in section 3.3, various experiments are run to narrow down the possible business scenarios to yield the best performing scenario. By executing the various methods, the experiment converges to the best strategy after a certain period.

The raw data sets are of the format portrayed in table 4. One can see that a product arrives at the factory in *region* 1, is the picked up and transported to *region* 2, and finally dropped off in *region* 3, which is considered the end point.

The experiments will focus mainly on the *currentDecayLevel* as a performance measure. This value is the representation of the quality of the product throughout the process, between 100 and 0, where 100 is high quality and 0 is low quality.



Figure 3. Visualisation of various vehicles in the simulations, adopted from [1]

Table 4. Sample taken nom event logs used in the experiments, in this case. Scenario 1 (Expl.txt)					
uniqueID	productIDstr	event	timeStamp	vehicle	currentDecayLevel
969	.Models.MUs.product3:15	arrivalAtSource	00:06:54.7458	-	100.0%
970	.Models.MUs.product3:15	productCallsTransportRegion1	00:06:54.7458	-	100.0%
971	.Models.MUs.product3:15	assignedToVehicleRegion1	00:06:54.7458	UAV:2	100.0%
972	.Models.MUs.product3:15	pickedUpRegion1	00:06:54.7458	UAV:2	100.0%
					•••
984	.Models.MUs.product3:15	droppedOffRegion2	00:07:00.9958	UAV:2	98.7%
985	.Models.MUs.product3:15	startProcessingRegion2	00:07:00.9958	UAV:2	98.7%
1658	.Models.MUs.product3:15	droppedOffRegion3	00:10:57.4733	UAV:4	97.1%
					••••

Table 4. Sam	ole taken from ev	ent logs used in the	experiments, in this	case: scenario 1	(Exp1.txt
--------------	-------------------	----------------------	----------------------	------------------	-----------



Figure 4. The case study 'factory plant', adopted from [1]

## 4.1.1 Process discovery

To get an overview of the event traces in the event logs, process discovery is applied. The discovered process model is portrayed in figure 5. In this instance the process mining tool  $\text{Disco}^1$  was used for the process discovery, with the default settings and filtered on complete event traces (so that incomplete event traces are discarded). This process model applies for all scenarios provided in the data sets.

#### 4.2 Experiments

For this research, several experiments are defined to narrow down the business scenarios. Every experiment uses a short algorithm that returns a ranking of the scenarios over a given time span.

The steps to determine the performance of scenarios is summarised as follows:

- 1. Traverse through event logs from timestamp  ${\cal A}$  to timestamp  ${\cal B}$
- 2. Filter on all events called droppedOffRegion3
- 3. Of the remaining events, calculate the mean of all values of *currentDecayLevel* and save it

<sup>1</sup>https://fluxicon.com/disco/

From now on this algorithm will be referred to as the *scenarios performance algorithm*. The resulting value is considered the performance measure in these experiments. A high result value means that the scenario yielded better performance than a low result. This is algorithm is executed for every scenario, after which a ranking can be made of every scenario over the time span from A to B. Pseudocode for this algorithm can be found in Appendix A, algorithm 1.

When running the experiments, the algorithm starts at the start of the event logs (*e.g. timestamp* 0) and traverses from time span AB, to time span BC, to time span CD, and so forth.

The different experiments that were executed for this research were summarised as follows:

- 1. After every iteration, use the top half of the scenarios according to the performance value and iterate once again until one scenario remains.
- 2. After every iteration, use the scenarios that scored higher than the mean of the performance values of all scenarios and iterate once again.
- 3. After every iteration, discard the scenario that yielded least performance and re-iterate with the other scenarios.

All experiments were run using Python and the Pandas<sup>2</sup> data analysis library for Python. The data sets were loaded and then processed using the described steps. The experiments themselves are explained in more detail below.

#### 4.2.1 Experiment 1

In the first experiment, after every iteration of the scenarios performance algorithm, the resulting performances are sorted from best performance to worst performance. Then the lower half of the scenarios are discarded, and the scenarios performance algorithm is executed on the

<sup>&</sup>lt;sup>2</sup>https://pandas.pydata.org/



Figure 5. The mined process model from the provided data sets. This is scenario 1 in particular. Accompanying time values are the mean durations between the actions.

top half of the scenarios. This method is re-iterated until one scenario remains.

The time interval chosen in this experiment is 60 minutes, which means that after 60 minutes, all events droppenOffRegion3 are evaluated over the time span of the past 60 minutes. Following this time interval, the first iteration will run from timestamp 0:00:00.0 to 0:59:59.9, the second iteration from 1:00:00.0 to 1:59:59.9, and so forth. Pseudocode for this experiment can be found in appendix A, algorithm 2.

#### 4.2.2 Experiment 2

This second experiment is similar to the first experiment, but incorporates a key difference. The results of the scenarios performance algorithm are now combined together to calculate the mean performance of all scenarios. Then the scenarios performance algorithm is performed on all scenarios that had a better performance score than the mean performance. This repeats until one scenario remains.

The time interval remained 60 minutes. Pseudocode for this experiment can be found in appendix A.

#### 4.2.3 Experiment 3

In the third experiment, the results of the scenarios performance algorithm are sorted, and the scenario that yielded the least performance is discarded. The rest of the scenarios are evaluated over the next time span and this is reiterated until few scenarios remain.

The chosen time interval varies: a bigger sample size of the finished products is desired (in more time, more products can reach the finished state), however this takes more time. The chosen time intervals are 5, 10, 20 and 30 minutes. It is worthy to note that the experiments ran for 11 hours (660 minutes), which means that an interval of 30 minutes and a set of 27 scenarios would require 810 minutes to determine the best scenario. Even though, the event logs were 'too short' to be evaluated until the end, the experiment with a 30 minute interval resulted in a top 6.

For this experiment, the pseudocode is stated in appendix A.

# 5. RESULTS

In this section, the results of the experiments will be discussed. It is interesting to look at the results as a business decision maker. They run several scenarios, in this case in a factory, and then eliminate scenarios that yield the least business performance.

## 5.1 Experiment 1

The first experiment ran for four hours, and after every hour the bottom half of the scenarios were eliminated. This lead to visualisation of the run in figure 6, below.



Figure 6. Visualisation of experiment 1. After every hour, half of the scenarios are discarded. Numbers in the data points represent the current ranking.

This result shows that scenarios 12, 16 and 15 formed the top three of all the scenarios after 3 hours. As can be seen

in table 3, scenarios 12 and 16 both utilised the productinitiated rule: *call shortestTravelDistance*, where scenario 16 also utilised the vehicle-initiated rule: *call highest quality.* According to this experiment, these scenarios yield the most business performance.

## 5.2 Experiment 2

The second experiment ran for five hours. After every hour, the mean product quality of all scenarios was calculated, and every scenario that scored under this value was eliminated. This lead to the visualisation of the run shown in figure 7.



Figure 7. Visualisation of experiment 2. After ever hour, scenarios performing worse than average are discarded. Numbers in the data points represent the current ranking.

It is very notable that the top three scoring scenarios are different than from the first experiment, while the conditions were quite similar. Only the elimination conditions were changed. In this experiment scenarios 12, 14 and 16 came out as the highest scoring scenarios. Scenario 15, which in experiment 1 ended in the top 3 scenarios, was eliminated after the fourth hour into the experiment, as the performance indicator over the past hour scored lower than the average of all the scenarios.

In experiment 1, scenario 14 was eliminated as part of the bisecting strategy, which does not necessarily mean that its score was 'bad'. In fact, when looking at the raw results, the difference between the scores of scenario 14 (89.5%) and scenario 15 (89.7%), that was not eliminated, was only 0.2% which is small when compared to other values.

## 5.3 Experiment 3

The third experiment was executed with relatively short intervals, namely 5, 10, 20 and 30 minutes. The scenario that yielded the least performance was eliminated and experiment continued in the next time span. The visualisation of one entire run can be found in appendix 10. One of the interesting results came when the experiment was executed with an time slot interval of 20 minutes, which is considered to be a suitable interval as:

- The sample size of finished products is large enough during most time spans
- It was possible to finish the algorithm before the event logs ran out, i.e. it was possible to eliminate every scenario but one.

The visualisation of the last intervals in the experiment (minutes 340-540), filtered on the six top scoring scenarios, is depicted in figure 8 below.



Figure 8. The rank progression of the 6 best performing scenarios, in the last three hours of the experiment.

Two key events stand out in this graph:

- The scenario that yielded most performance after 540 minutes (12) was ranked seventh when polled after 360 minutes.
- The scenario that yielded the second most performance after 520 minutes (15), was ranked fifth after 340 minutes into the experiment.

One question that arises is whether these scenarios were performing so much worse than others at that point in time. Therefore, the actual mean values of the product quality after every interval was plotted and depicted in the graph in figure 9.



Figure 9. We can see that the quality value is actually very close in every scenario, even though they are ranked after each interval.

This graph shows that even though scenario 12 was ranked seventh after the 360 minutes interval, the actual value

of the product qualities were not too low. Furthermore, some scenarios that were ranked higher later dropped their values to a lower value than scenario 12 at that point.

Another finding is that the value of scenario 15 at the 340 minutes interval (ranked fifth at that point) was higher (90.4) than the values of most scenarios during intervals later in the run.

## 5.4 Summary

The three conducted experiments show similar results. Most of the experiments return scenarios 12, 14, 15 and 16 as the best scenarios. When the course of the run-time of the experiments is observed, these scenarios often score pretty high. However, and important observation to be made is that the scores of the scenarios fluctuate after every interval, as illustrated clearly in experiment 3. The top ranked scenarios tend to vary in the ranking, however they stay at the top scoring best.

## 6. DISCUSSION

In this section, a few observations are lead out for improvement on this particular work. The experiments show that event logs analysis and scenario elimination based on the results can help business decision makers in narrowing down the available options.

However, a few ideas are proposed to make the experiments and thus the results more reliable:

- The sample size in the time intervals needs to be considered. When business decisions are made based upon small sample sizes, the decision may turn out to demote business performance.
- Statistical analysis need to be taken into account when processing the results. For example, a combination of the mean and standard deviation can be used as performance indicators in stead of just the mean values in these experiments.

Future research can mainly expand on the analysis of other possible performance indicators in event logs, and on classifying the data found in event logs to perform statistical analysis on eliminating scenarios that do not yield the intended performance.

# 7. CONCLUSION

In many business domains, many variables have influence on the performance of a business. For business decision makers it can be hard to make choices that yield the most performance, as time and money constraints limit the ability to test all variations of the aforementioned variables. Using event logs analysis and performance analysis, this research aims to provide methods for business decision makers to narrow down the number of considered business options.

To achieve this goal, three questions need answering. Firstly, the performance indicators of a business in the logistical sector are defined. Secondly, an analysis is made on how event logs can aid in providing performance information and thirdly, definitions are made for how event logs analysis can be used to narrow down business scenario options and comply with the given performance indicators.

These findings are used in a case study, where three experiments are conducted in a simulated factory. First, the performance indicator in the event logs are defined, after which the experiments can be conducted. All possible scenarios (i.e. the variations of the business variables) are evaluated for a given time period, after which the worst performing scenarios are dropped. The experiments then continue until they converge to the best performing scenarios, according to the performance score. The methods of evaluating the event logs and dropping strategies is altered across the three experiments.

The experiments returned similar results in that the top performing scenarios were similar across the experiments. Furthermore, these results show that event logs analysis, combined with performance indicators can aid business decision makers in narrowing down the considered options in a time-efficient manner.

## 8. ACKNOWLEDGMENTS

Special thanks to Rob Bemthuis for providing the data sets for the event logs analysis, a lot of knowledge of these data sets and for his inspiring ideas for this research.

# 9. REFERENCES

- R. Bemthuis, M. Mes, M. E. Iacob, and P. Havinga. Using agent-based simulation for emergent behavior detection in cyber-physical systems. In 2020 Winter Simulation Conference (WSC), pages 230–241.
- [2] R. Bemthuis, M. M.R.K., I. M.E., and H. P.J.M. Data underlying the paper: Using agent-based simulation for emergent behavior detection in cyber-physical systems. 2021.
- [3] E. Krauth, H. Moonen, V. Popova, and M. C. Schut. Performance measurement and control in logistics service providing. In *ICEIS* (2), pages 239–247.
- [4] F. Mannhardt, M. De Leoni, and H. A. Reijers. Heuristic mining revamped: An interactive, data-aware, and conformance-aware miner. In *CEUR Workshop Proceedings*, volume 1920.
- [5] W. Van der Aalst. Process mining: Data science in action. Process Mining: Data Science in Action. 2016.
- [6] W. Van Der Aalst et al. Process mining manifesto. In Lecture Notes in Business Information Processing, volume 99 LNBIP, pages 169–194. 2012.
- [7] W. M. P. van der Aalst. Process Mining; Data Science in Action. Springer, Berlin, Heidelberg, 2nd edition, 2016.
- [8] H. M. W. Verbeek, J. C. A. M. Buijs, B. F. Van Dongen, and W. M. P. Van Der Aalst. Prom 6: The process mining toolkit. In *CEUR Workshop Proceedings*, volume 615, pages 34–39.
- [9] R. Wirth and J. Hipp. Crisp-dm: Towards a standard process model for data mining. In Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining, volume 1. Springer-Verlag London, UK.

# APPENDIX

# A. ALGORITHM PSEUDOCODES

See algorithms 1 and 2 for the pseudocode that enabled the experiments.

## **B. RAW RESULT OF EXPERIMENT 3**

See figure 10 for the complete course of the run of experiment 3, with an interval of 20 minutes.

```
Algorithm 1 Function for calculating the mean of the decay values of 'finished' products, in an array of scenarios (event logs)
```

```
1: function DETERMINEMEANDECAYOVERTIME(array scenariosData, timeStamp start, int delta)
       calulatedDecayMeans : [scenarioID, calculatedValue]
2:
3:
       for all scenario \in scenariosData do
 4:
          decayLevels: [value]
          for all event \in scenario.events do
5:
6:
              if start < event.timeStamp < (start + delta) AND event = droppedOffRegion3 then
 7:
                 decayLevels \frown [event.currentDecayValue]
              end if
8:
9:
          end for
10:
          calculatedDecayMeans \sim [scenario.id, mean(decayLevels.values)]
11:
       end for
       return calculatedDecayMeans
                                                ▷ Array of tuples of { scenarioID, mean decay level of finished product }
12:
13: end function
```

Algorithm 2 Functions for the three experiments

**Require:** int deltaMin = 60  $\triangleright$  Since this is the same value throughout algorithm, this variable is global

1: function ELIMINATEBOTTOMHALFANDCONTINUE(array scenariosData, timeStamp previousTimeStamp)

- 2:  $sortedScenariosData \leftarrow sortOn(scenariosData.meanDecayLevels)$
- 3:  $meanQualityDecay \leftarrow mean(scenariosData.meanDecayLevels)$

Experiment 1

- 4:  $topScenariosData \leftarrow sortedScenariosData[0 \cdots 0.5 \times length]$
- Experiment 2
- 5:  $topScenariosData \leftarrow sortedScenariosData[0 \cdots \{sortedScenariosData.value > meanQualityDecay\}]$ Experiment 3
- 6:  $topScenariosData \leftarrow sortedScenariosData[0 \cdots length 1]$

**Require:**  $lastResults \leftarrow [scenarioData]$  **Require:**  $lastStartMinute \leftarrow 60$ **Ensure:** int *iterations* = 5

▷ This value is picked by hand

9: while iterations > 0 do

- 10: results  $\frown$  ELIMINATEBOTTOMHALFANDCONTINUE(*lastResults*, *lastStartMinute*)
- 11:  $lastStartMinute \leftarrow lastStartMinute + deltaMin$
- 12:  $iterations \leftarrow iterations 1$

 $13:\ \mathbf{end}\ \mathbf{while}$ 



Figure 10. After every 20 minutes, discard the single worst performing scenario over the past time slot