Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Detection of Temporal Dependencies in Alarm Time Series of Industrial Plants

Jens Folmer, Falk Schuricht and Birgit Vogel-Heuser

Institute for Automation and Information Systems
Technische Universität München, Garching near Munich
(e-mail: folmer,schuricht, vogel-heuser @ ais.mw.tum.de)

Abstract: Concerning industrial plants, operators face the problem that more alarms are generated than can be physically perceived and addressed by a single operator. Such a situation is called alarm flood. The main reason for alarm floods are causally related disturbances, which either way raise an alarm, based on a single causal disturbance. These dependencies are difficult to recognize during the engineering of an AMS (Alarm Management System). However, the alarms are logged and stored as time series (historical data). Information about the alarm types and the time stamps of their occurrence can be used to analyze the time series data and thus finding dependencies between different alarms. This contribution presents an approach to find temporal dependencies between alarm events in an alarm time series. Therefore an algorithm was designed, implemented, and evaluated to detect temporal dependencies in alarm time series.

Keywords: Timing Series Analysis, Manufacturing Processes, Process Automation, Data Processing, Alarm Systems, Delay Analysis

1. INTRODUCTION

Process Control Systems (PCS) are one of the core elements in process control industry such as oil refineries or pharmaceutical and chemical plants. The PCS presents the most important information about the process plant to the operator. In addition to the observation of process data, the operator has to recognize abnormal plant situations. An abnormal plant situation is indicated by process visualizations and alarm lists. Messages about abnormal plant situations can be divided into two types, defined by the EEMUA (2013) as alarm and alert.

An information message is indicated by using textual or graphical elements, e.g. based on a P&ID (IEC 62424 (2013)). The alarm management system (AMS) is an integral component of the PCS. The AMS presents detailed information of raised alarms in a chronological or priority order visualized by a table. By intervening, the operator is able to clear abnormal situations.

Due to causal dependencies of failures (e.g. based on process/material flow), too many alarms may frequently be raised. Each alarm is displayed to the operator, who must handle this vast quantity of alarms. This effect is called "alarm flood". Operators only acknowledge alarms, while handling the abnormal situation based on their process knowledge and experience. In such situations operators do not have enough time to analyze every alarm properly without exceeding the recommended operator response time. Thus, the prediction and reduction of alarm floods has a high priority when improving alarm systems design. The EEMUA 191 recommends an alarm rate less than one alarm per 10 minutes in steady operation. To reduce the quantity of alarms, on the one hand improved visualizations can be applied to reduce the overload of information as proposed in Pantforder and Vogel-Heuser (2009); Folmer et al. (2011). On the other hand, one should identify causal dependent alarms based on time series analysis to predict causal dependent malfunctions to inform operators, which then are able to intervene earlier. Those causal dependent alarms become more apparent when alarms are generated during plants' run-time. The different generated alarms are stored in an alarm log as historical records of all alarms. These alarm logs can be automatically analyzed in order to find temporal and causal dependencies between alarms, by using statistical analysis techniques. The results of the analysis can be reviewed by experts to re-design the AMS or/and to design a forecast system.

The reminder of the contribution is as follows. Related work on time series analysis is presented in section 2. In section 3, a new approach to find causal dependent alarms is proposed. The results of the analysis based on industrial data logs are presented in section 4. Section 5 concludes this contribution and introduces further research.

2. RELATED WORK ON ALARM DATA CAUSALITY ANALYSIS

Data Mining methods are used to discover important information in large data sets. Regarding alarm data sets and causality analysis, the important information is represented by a sequence of alarm events, which occur at certain time stamps. In this section, related works to find causal dependent alarms by consideration of time are presented.

2.1 Sequential Pattern Mining

Sequential Pattern Mining, a sub discipline of Data Mining, is focusing on methods to find pattern in sequential data sets. Zhao and Bhowmick (2003) as well as Boghey and Singh (2013) give an overview of several Sequential Pattern Mining methods. Sequential data can be represented by a single sequence or a set of sequences containing single elements. A sequence is an amount of elements, which can be ordered (e.g. by time) or in random order. Often additional information, like geometrical or temporal data, is stored along with each element within a sequence. Sequences can for instance be represented as a string or a number of logged events whereat each event is generated during the plants run-time, e.g. a failure event. The objective of Sequential Pattern Mining is to find interesting patterns in the data set. Patterns can be an amount of frequently occurring elements. They can also be sub-sequences, which are correlated with interesting events. In addition, dependencies between sub-sequences can be found by examining their frequency of occurrence or by ordering in the data set. Important information in alarm data sets are sub-sequences, which occur frequently. The detection and analysis of the dependencies of these sub-sequences lead to structural information about the alarm sequence.

The authors themselves (Folmer and Vogel-Heuser (2012)) presented the Automatic Alarm Data Analyzer (AADA) Algorithm. It clusters frequent occurring sub-sequences in alarm logs, by searching alarm sequences using finite automaton. Each recognized alarm (in the alarm log) is represented by a state and transitions are used to represent state transitions of the alarm that is raised. This leads to an automaton, which encapsulates the overall structure of the sequential data. The authors extracted time dependencies by ordering the alarms in the alarm log, but did not use time dependencies during the analysis itself for recognizing causal dependencies. However, they pointed out how important pre-processing of the alarm log is by taking important alarm types into account and not the overall alarm log.

Ahmed et al. (2013) introduce several methods of analyzing alarm flood data. Alarm floods consist of a set of ordered alarm events. They are determined by clustering alarm floods and comparing them to each other. Similarity between the flood is measured by finding a mapping between alarm flood with dynamic time warping. However, these methods do not consider temporal information to find useful patterns or dependencies.

All proposed approaches focus on determining relationships between events in general. Only Ahmed et al. (2013) focus on alarms in industrial automation or industrial plants. However, previously mentioned approaches do not consider time intervals between single events. Additionally, they are only dealing with pattern sequences.

2.2 Time-Series Data Mining

Time-Series Data Mining deals with huge sequences of elements having a temporal order. Esling and Agon (2012) and Fu (2011) give an overview over several techniques of Time-Series Data Mining. Some methods concentrate on changing the representation of time-series to reduce the dimensionality or on enhancing the performance of other mining techniques. Main task is finding sequences or sub-sequences in the time-series, which occur frequently or have some abnormal features. For that purpose, sequences or sub-sequences are often compared to each other using a similarity measure. The sub-sequences in the time-series are then clustered to find some overall structure. Temporal aspects are taken into account regarding the found clusters. The temporal behavior of the time-series in terms of time intervals between sequence elements or the detection of cyclic occurring patterns is then analyzed. Of particular interest is to find causal relationships in time-series. This task is called Temporal Association Rule Mining. Even if the approaches mentioned above are taking time intervals into consideration, no research is dealing with uncertainty of time intervals between data points. Furthermore, they do not focus on alarms of industrial automation systems.

2.3 Association Rule Mining

Schluter and Conrad (2011) describe several approaches to find association rules in time-series. A temporal relationship is discovered by analyzing the ordering and interval of sub-sequences in a time-series. If a sub-sequence B follows a sub-sequence A within a certain time frame, there is a temporal dependency. Association rules can be found by regarding the temporal order, also between different time-series.

Associated events can occur periodically in a time-series. Some approaches like Thuan et al. (2012) and Li et al. (2001) focus on finding cyclic association rules, i.e. rules that are always true in certain time intervals. Höppner and Klawonn (2002) and Bouandas and Osmani (2007) developed methods to find temporal association rules in sequences that consist of intervals. A sequence is regarded as a set of states, which have a defined start and end time. These state intervals are related to each other. Rules are found by determining frequent occurrences of interval relationships.

In Mannila et al. (1995) an algorithm is developed, which finds frequent episodes in sequences of events. An episode is a set of events, which occur frequently in a certain order. In this approach, time is taken into account by defining a time frame, in which episodes can be found in the event time-series.

The presented approaches incorporate time in different ways. Often (pre-defined) time frames are regarded in which certain sub-sequences must follow each other, to be taken into account for a temporal dependency. Ordering and relationships of temporal state intervals are also used to find association rules.

In the presented contribution, association rules are found by regarding time intervals of explicit length between elements of a sequence. The approach assumes that the emergence of a sequence element causes the occurrence of another element after a certain time. Those causal dependencies, which incorporate hard real time constraints, can detect unknown interrelations in data sets, which can lead to better understanding of the underlying processes.

3. DEVELOPED CONCEPT

A time series can be obtained from an alarm log with N alarms. Each time an alarm is raised, it is logged with the time stamp of its occurrence. The corresponding alarm log A_{log} can then be formulated as a set of ordered elements as follows:

$$A_{log} = \{a_1, a_2, ..., a_{N-1}, a_N\} \tag{1}$$

Whereat a_N occurs after a_{N-1}. Each alarm a_N is a tuple

$$a_i = (t_i, c_i), \quad i \in \mathbb{N}, \quad 1 \le i \le N = |A_{log}| \tag{2}$$

where c_i is a unique alarm identifier (ID – usually an identifying number), which occurs at a certain time t_i. In the following, two alarms with IDs "ID_c" (cause alarm) and "ID_e" (effected alarm) are considered. Alarms with "ID_e" are an effect of a caused alarms with "ID_c" after a certain time interval. Depending on the considered process, this time interval may deviate. In this contribution, this time interval which indicates a causal dependency between cause and effect is called a temporal relationship. The existence of such a relationship is verified by statistical methods.

3.1 Statistical description of temporal relationships

Inspecting the two related alarms of the corresponding alarm log (log), "ID_e" is assumed to be caused by "ID_c". In case of the temporal relationships regarded in this paper, an occurrence of "ID_c" always causes "ID_e" after a certain time with some deviation. The time interval, in which these alarms follow each other, is used as a random variable. This random variable can be described by a probability density function. Parameters of the density function are estimated by analyzing all existing time intervals between two related alarms in the log. Resulting in a probability distribution, a confidence interval is used to verify that the temporal interval has a high probability within the defined boundary and, hence, a high significance.

However, this approach assumes that every raised alarm potentially causes another alarm. There is also the possibility, that an alarm is sometimes caused by another event, e.g. if the operator intervenes early to bring under control the abnormal situation as soon as the effected alarm occurs. In that case, time intervals would lead to wrong probability density functions. To tackle this challenge, in the next section the possible mappings between alarms are described.

3.2 Mapping cause and effects of alarms

In case of the temporal relationships, an occurrence of "ID_c" causes "ID_e" after a certain time with some deviation. However, it is not a priori obvious, which root cause alarm must be mapped to the corresponding affected alarm, because the proposed approach does not include additional information, e.g. plant structure. Fig. 1 shows two examples of possible mappings between alarms. In Fig. 1 (a), the caused alarm is always the next alarm in the log with alarm ID "ID_e", whereby some independent alarms occur between the dependent alarms. To discover a temporal relationship, the first occurrence of an alarm with "ID_e" after emergence of an alarm with "ID_c" can be used to calculate the time interval between both alarms.

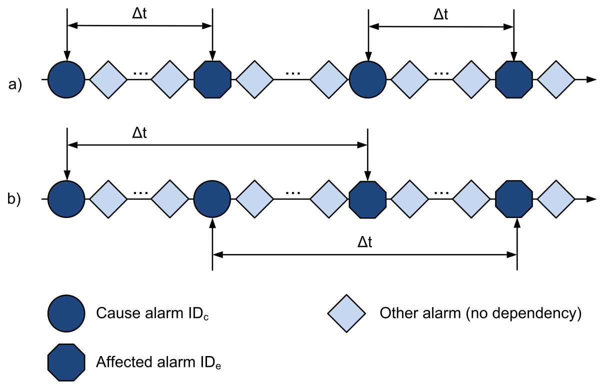

Fig. 1. Two examples of possible temporal relationships between alarm in an alarm log which may occurs

However, in real data sets, initiating alarms can also occur in columns (see Fig. 1 (b)), e.g. flicker alarms. If there is a temporal relationship with a large time interval and if only the next occurrence of the caused alarm (first alarm of the column of caused alarms) is taken into account, the dependency will be missed. To capture these relationships, also time intervals, in which temporal dependencies are assumed, must be known. Hence, alarms with "ID_e" in the regarded time interval can be investigated and relationships can be found. This information leads to the formulation of a condition for temporal dependencies, which is explained in the next subsection.

3.3 Condition to determine temporal dependencies

Supposing there is a temporal dependency after a time interval Δt, one can define a condition including a possible deviation as follows:

$$|t_i + \Delta t - t_j| < \epsilon \tag{3}$$

Where t_i corresponds to an alarm with $c_i = ID_c$ and respectively t_j to an alarm with $c_j = ID_e$. That means the two alarms a_i and a_j of the log follow each other with a time interval Δt within a maximal deviation ϵ. In case this condition holds for a certain amount of occurrences of a_i, a temporal dependency is assumed for the alarms with the IDs "ID_c" and "ID_e". Fig. 2 illustrates the condition graphically.

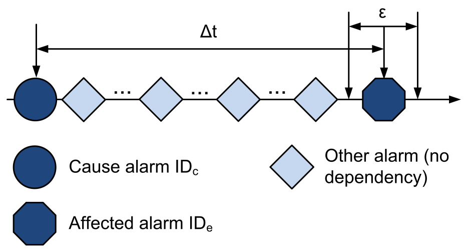

Fig. 2. Alarm log with two dependent alarms within time Δt and deviation ϵ

This condition can now be used to formulate a conditional probability of a temporal relationship between alarms with "ID_c" and "ID_e". The conditional probability is defined as follows:

$$P(a_i|a_j) = \frac{P(a_i \wedge a_j)}{P(a_j)} \tag{4}$$

The probability of two random alarms a_i and a_j can then be formulated as the probability of occurrence of "ID_c" and "ID_e" within time interval Δt with a maximum deviation ϵ divided by the probability of occurrence of "ID_c".

In case the calculated probability is higher than a certain threshold "P_{thr}", there is a temporal dependency between alarms with "ID_c" and "ID_e" in the time interval Δt with a maximum deviation ϵ. The threshold value "P_{thr}" has to be chosen manually, e.g based on the dead-times and/or speed of the process.

Thus the conditional probability can be formulated in a mathematical function depending on the occurrences of alarms in the log.

$$P(ID_c, ID_e, \epsilon, \Delta t) = \frac{N_1}{N_2} \tag{5}$$

$$\begin{aligned} N_1 = |\{c_i|(t_i, c_i) \in A_{log} \wedge \\ (t_j, c_j) \in A_{log} \wedge \\ c_i = ID_c \wedge c_j = ID_e \wedge \\ |t_i + \Delta t - t_j| < \epsilon)\}| \end{aligned} \tag{6}$$

$$N_2 = |\{c_i|(t_i, c_i) \in S \wedge c_i = ID_c\}| \tag{7}$$

This function only describes a relationship between two observed alarms in a particular time interval Δt and a certain deviation ϵ. Evaluating all possible combinations of alarms to calculate all time intervals and deviations leads to a high computational effort.

Therefore only alarms which have a certain minimal occurrence "min_{occ}" in the log have to be considered. Alarms that occur rarely in corresponding log are ignored by the algorithm. This does not mean that rarely occurring alarms are unimportant but, furthermore, the results of the proposed approach are not significant in this case because there are not enough samples.

The next challenge is to choose an appropriate time interval for the considered alarm tuples. Since the time distribution of the alarm around a certain alarm is unknown, an estimation of the time interval is a challenging task. A promising method is to analyze many time intervals, which cover a whole time frame after the occurrence of the initial alarm (alarm with ID "ID_c") in the log. In this contribution, time intervals in uniform intervals are chosen using a histogram. The bins depend on the deviation ϵ.

$$\Delta t_i = \epsilon + 2 \cdot \epsilon \cdot i, \quad i = 0...M, \quad M = \left\lceil \frac{t_{max} - \epsilon}{2 \cdot \epsilon} \right\rceil \tag{8}$$

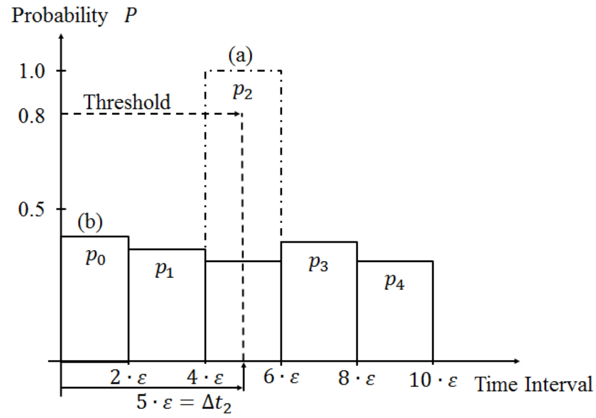

Fig. 3. Possible discrete derivations of probabilities of an alarm event tuple (a) Ideal temporal dependence (b) Uniformly distributed time intervals between events with "ID_c" and "ID_e"

By choosing these time intervals, every point in time until t_{max} after the occurrence of the initial alarm is exposed. Thus no temporal dependence will be missed. To find temporal relationships, the maximal time that has to be analyzed is t_{max}. However, this approach only regards a finite set of time intervals. Temporal dependencies which have not exactly one of the time intervals Δt_i can also be found. This depends on the real deviation of the time intervals of the related alarms around the caused alarm .

The next analysis step is to compute for every relevant alarm tuples the corresponding probabilities for all time intervals Δt_i. The probabilities can be obtained by iterating over the overall alarm log and by counting the occurrences of following alarms in certain time intervals. The results can then be stored in a discrete deviation of probabilities:

$$P(i) = p_i, \quad p_i \in P = p_0, p_1, ..., p_m \tag{9}$$

$$P(i) = P(ID_c, ID_e, \epsilon, \Delta t_i) \tag{10}$$

This deviation can be visualized by a histogram. Fig. 3 (a - dashed line) shows the bar chart of an ideal temporal dependency. The probability of the caused alarm occurring in the time interval $[4 \cdot \epsilon, 6 \cdot \epsilon]$ is 1, which means that an alarm always follows an initiating alarm within a specific time 5ϵ and time deviation 4ϵ to 6ϵ. This detected dependency is significant, since there is an element of the amount of interval probabilities, which is higher than the pre-defined probability threshold "P_{thr}" (in this case $P_{thr} = 0.8$). Fig. 3 (b - bold line) shows the bar chart of another discrete deviation. Here, a nearly uniformly distributed alarm type with "ID_e" is assumed. In contrast to Fig. 3 (a), it occurs that the time between alarms fits into several bins of the bar chart and, therefore, does not reach the pre-defined threshold. In this case no dependency is detected using the presented approach, because the calculated time dependency is too inaccurate.

There is also the possibility that more than one element of the discrete probability distribution can exceed the given threshold. This can for example happen, if the time frame is large enough to detect the same temporal dependence twice. The bar chart would have 2 peaks at different time intervals. However, a tuple of alarms can only have one temporal dependency. This is the dependency with the lowest time interval Δt_j.

3.4 Developed algorithm and computational complexity

The developed concept was implemented by an algorithm. For each tuple of regarded alarms it iterates the alarm time series. If an alarm event with "ID_c" is found, all following alarms with "ID_e" that occur in the specified time window are investigated. They are then accumulated according to their time interval and placed in an array. Knowing the occurrences of alarms with "ID_c", the discrete probability distribution can be calculated.

Assuming a medial amount of alarms with "ID_e" to be investigated when an event with "ID_c" occurs, the effort of computing a discrete probability deviation for a tuple of alarms is linear to the number of alarms in the time series. Furthermore, the number of tuples to be examined is quadratic to the amount of different unique alarms types in the time series. The overall computational complexity of the algorithm can then be estimated:

$$complexity(N, Q) = O(N, Q^2) \tag{11}$$

N: Amount of alarms in the time series
Q: Amount of unique alarms types to be investigated

The computational complexity and the number of alarm types to analyze must be kept low. Thus only tuples with alarms that have a certain minimal occurrence in the time series are investigated. An appropriate approach to minimize the amount of alarms is to consider important event types, which are essential for related events, e.g. deleting visualization alarms or notification alarms without meaningful information about the manufacturing plant. The size of the time frame also affects the computational complexity linearly. The choice of the deviation has no effect on the computational effort.

4. EVALUATION AND DISCUSSION

The developed concept was evaluated with eight different alarm logs from industry plants. The algorithm was applied on alarm logs from continuous (process control industry), discrete (manufacturing industry) and hybrid processes. Table 1 gives a summary of the characteristics of the alarm logs.

The data have been collected over several days during the run-time of every plant. The shown alarms have strong variations in the number of investigated alarms, alarm types and mean times between the alarms. However, despite the regarded time series are from very different processes, the quality of the results is highly affected by the chosen algorithm parameters (e.g. t_{max}, ϵ). During the evaluation, several experiences on the choice of these parameters were made. They are described later in this section. The results of the developed algorithm contain information about the time interval, deviation, probability and number of observations of each temporal dependency. For visualization purposes, the temporal dependencies can be illustrated as directed graphs. Fig. 4 shows an example of such a graph. The nodes contain alarm IDs. Alarms

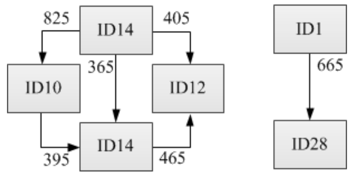

Fig. 4. Directed graphs illustrating temporal relationships found in an alarm time series. Excerpt of a bigger graph structure

are connected with edges. At the edge the mean time of a temporal relationship between two alarms is shown. Fig. 4 shows an excerpt of a bigger graph structure, visualizing temporal dependencies in a plant. The graph shows only relationships with a minimum occurrences of 100 in the alarm log and an incidence probability of at least 0.8. The chosen maximal deviation ϵ is 10 seconds.

The above mentioned graphs have been reviewed by plants' operators which are experts on plants' behavior. The experts pointed out that the algorithm is able to detect causalities between failures they did not know before or they detected only recently. For instance erroneous inputs have been made by the operator that affected the overall process and raised other alarms. This information can be used for a re-design of the operator interface, to avoid these erroneous inputs by the operator. Additionally the algorithm points out alarms that occur first and lead to plant shutdowns, e.g. failure of axes. These causal dependencies can be used to improve the intervention recommendations for the operator, to inform the operator earlier. Interrelations between logged alarms were discovered (e.g. incorrect inputs by the operator, which raise alarms due to failure of axes, which then leads to plant's shutdown) but due to the statistical approach, some random relationships were also recognized. Even if there are some significant alarm dependencies, there are still some major drawbacks.

The detected time intervals between alarms in most alarm logs have strong variations. They can vary from seconds to hours as it is the case in nearly every analyzed process. The reason for that is the high non-determinism of the plant's behavior due to the operators' behavior and several other influencing parameters (environment, produced product etc.) . Additionally, the alarm log is not pre-processed and, hence, also alarms occurring during maintenance are stored in the alarm log, which distort the statistical alarm sequence recognition. Furthermore, the proposed approach expects input parameters, which are used for the

Table 1. Analyzed alarm time series with industry segments.

		Plant (Alarm Log) Number							
		1	*2*	*3*	*4*	*5*	*6*	*7*	*8*
Alarm Log Properties	*Process kind*	D	D	C	C	C	H	H	H
	Number of alarms	50T	10T	25T	29T	5T	50T	9T	10T
	Number of unique alarm types	373	37	537	427	223	426	200	37
	Recording time (days)	59,00	66.9	18,00	14,30	34.56	61.3	9,10	66.9
	Mean time between alarms (seconds)	102,00	33,00	34,40	41,90	21.35	65,00	41.8	33.4

Legend: C: Continuous Processes, D: Discrete Processes, H: Hybrid Processes alarm sequence recognition. These parameters cannot be assumed to be static. For instance, a causal dependent failure between a pump and the filling level of a vessel depends on the volume of a tank and the volume/length of the pipe. Hence, proposed parameters have to be set a priori, e.g. by using engineering data.

Hence, the temporal information gained from this algorithm must be combined with knowledge about the particular processes. For example, a process flow model can be connected with the temporal dependencies to discover relationships in the occurrences of abnormal process parameters. This can be used to derive changes in the plant's functionality to avoid these erroneous situations. Furthermore the discovered dependencies can be used to predict soon arising alarms at certain times. Alarms that occur in direct temporal relations can also be removed from the alarm system to reduce operators' workload.

The interviewed operators mentioned that alarms are raised during downtime of plant sections, i.e.during maintenance. These alarms are not disabled and, hence, are illustrated to the operator. These alarms do not include important information and may hide important alarms. Unfortunately, these alarms are still recorded and complicate the recognition of dependencies. Further challenges are longer time intervals, where huge delays between cause and effects can lead to high deviations in the alarm event times. Hence, temporal relationships cannot be detected anymore in longer time intervals.

5. CONCLUSION AND OUTLOOK

This paper presents an algorithm to find causal dependencies of alarm within alarm logs, recorded during the run-time of industrial plants. A temporal dependencies are determined by analyzing the temporal dependency between two alarms, which occur several times in the alarm log, based on statistical approaches. In this contribution important parameters for the analysis are defined. The algorithm can find relationships even in longer time intervals by regarding multiple time intervals at once. For the evaluation an algorithm with linear computational effort has been developed and applied on eight different alarm logs from different industry segments. The results pointed out causal dependent alarms, e.g. based on erroneous inputs of operators or based on the process/product flow. The benefits and major drawback of parameter estimation for the algorithm are discussed. The gained information can be used to make alarm management systems more effective or to discover dependencies in the erroneous behavior of a plant. This can be used to reduce operators' workload, to draw further conclusions about the erroneous behavior of the plant processes or the configuration of the alarm management configuration. In addition, the temporal dependencies can be used to forecast several critical alarm events before they occur. This knowledge can be used to avoid those events and thus improve the performance of the whole plant process.

Due to the drawback of the algorithm, future research will focus on combining further process information, like plants' layout and additional parameters about plants' operation mode, to improve the results of the proposed algorithm.

REFERENCES

Ahmed, K., Izadi, I., Chen, T., Joe, D., and Burton, T. (2013). Similarity Analysis of Industrial Alarm Flood Data. *IEEE Transactions on Automation Science and Engineering*, 10(2), 452–457.

Boghey, R. and Singh, S. (2013). Sequential Pattern Mining: A Survey on Approaches. In *International Conference on Communication Systems and Network Technologies (CSNT)*, 670–674.

Bouandas, K. and Osmani, A. (2007). Mining Association Rules in Temporal Sequences. In *Symposium on Computational Intelligence and Data Mining*, 610–615.

EEMUA (2013). *Alarm systems a guide to design, management and procurement*. EEMUA publication No. 191, 3rd edition.

Esling, P. and Agon, C. (2012). Time-series data mining. *ACM Computing Surveys*, 45(1), 1–34.

Folmer, J., Pantförder, D., and Vogel-Heuser, B. (2011). An analytical alarm flood reduction to reduce operator's workload. In J. Jacko (ed.), *Human-Computer Interaction. Users and Applications*, volume 6764 of *Lecture Notes in Computer Science*, 297–306. Springer Berlin Heidelberg.

Folmer, J. and Vogel-Heuser, B. (2012). Computing dependent industrial alarms for alarm flood reduction. In *9th IEEE International Multi-Conference on Systems, Signals and Devices (SSD)*, 1–6.

Fu, T.c. (2011). A review on time series data mining. *Engineering Applications of Artificial Intelligence*, 24(1), 164–181.

Höppner, F. and Klawonn, F. (2002). Finding informative rules in interval sequences. *Intelligent Data Analysis*, 6(3), 237–255.

IEC 62424 (2013). Representation of process control engineering - request in p&i diagrams and data exchange between p&id tools and pce-cae tools.

Li, Y., Ning, P., Wang, X.S., and Jajodia, S. (2001). Discovering calendar-based temporal association rules. In *Temporal Representation and Reasoning, 2001. Proceedings. Eighth International Symposium on*, 111–118.

Mannila, H., Toivonen, H., and Verkamo, A.I. (1995). Discovering frequent episodes in sequences Extended abstract. In *1st Conference on Knowledge Discovery and Data Mining*.

Pantforder, D. and Vogel-Heuser, B. (2009). Benefit and evaluation of interactive 3d process data visualization in operator training of plant manufacturing industry. In *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on*, 824–829.

Schluter, T. and Conrad, S. (2011). About the analysis of time series with temporal association rule mining. In *IEEE Symposium on Computational Intelligence and Data Mining*, 325–332.

Thuan, N.D., Toan, N.G., and Tuan, N.L.V. (2012). An Approach Mining Cyclic Association Rules in E-Commerce. In *15th International Conference on Network-Based Information Systems*, 408–411.

Zhao, Q. and Bhowmick, S.S. (2003). Sequential pattern mining: A survey. *ITechnical Report, Nayang Technological University Singapore*, 1–26.