Discovering Systematic Relations between Alarms for Alarm Flows Reduction

Yannick Laumonier, Jean-Marc Faure, Jean-Jacques Lesage, Hervé Sabot

Discovering Systematic Relations between Alarms for Alarm Flows Reduction

Y. Laumonier, J.-M. Faure, *member IEEE*, J.-J. Lesage, *member IEEE,* and H. Sabot

Abstract— **Alarm systems play an important role for the safe and efficient operation of modern industrial plants. However, in most of industrial alarm systems, alarm flows cannot always be correctly managed by the operators as they often turn into alarm floods, sequences of numerous alarms occurring in a short period of time.**

To reduce the alarm flows, this paper focuses on detection of redundant alarms that could be removed. This objective is met by, first, looking for two systematic causality relations between alarms, called domination and mutual dependency, in the alarm log. Once these relations discovered, alarms removal requires an expert analysis. To ease this analysis, the discovered relations are depicted in the form of Petri nets. The proposed method has been applied on a dataset coming from power industry and provided by General Electric. The results of this case study show the benefits of the approach.

Keywords— *Industrial alarm systems; Alarm rationalization; Discrete Event Systems; Petri nets*

I. INTRODUCTION

An alarm is defined in the ANSI/ISA 18.2 standard (which is detailed in [1]) as "An audible and/or visible means of indicating to the operator an equipment malfunction, process deviation or abnormal condition requiring a timely response". Alarm systems play a primordial role for efficient and safe operation of modern industrial plants such as power plants, refineries, petrochemical facilities and manufacturing systems. However, the use of both computer assisted design tools and wireless sensors technologies makes so easy the monitoring of numerous variables that the number of alarms has drastically increased in the past years. Consequently, excessive alarm flows (called alarm floods), i.e. sequences of numerous alarms occurring in a short period of time, often happen today. As operators cannot react to these floods in the due time, off-specification products may be produced, the performances of the plant may be lessened and even, above all, hazardous events may occur.

To tackle this issue, the Engineering Equipment and Materials Users Association (EEMUA) has defined metrics in terms of numbers of alarms during a given period of time and proposed maximal values for these metrics. The maximal values of the average number of alarms per day and of the number of alarms in a 10 min time are respectively equal to 144 and 10 in this recommendation, for instance. Unfortunately, these values are strongly overtaken in industrial

systems where these metrics may reach thousands and hundreds. Hence, alarm number reduction is a current crucial industrial problem. It is particularly crucial because, on one hand, the number of alarms must be strongly reduced, and, on the other hand, an alarm carries an information about the dysfunctional behavior of the plant. Then, as the reduction of the number of alarms must not lessen safety and availability of the monitored system, focus must be put on detection of *redundant* alarms.

Our first contribution in this domain is presented in [2] where a pattern mining algorithm on alarm sequences detected frequent patterns composed of adjacent alarms. Once the frequent patterns detected, they were validated by an expert to check whether some alarms of the patterns were really redundant and should be removed. This approach may indeed lead to a significant reduction of the alarm number but it provides numerous patterns, which implies a long expert analysis, and it is not able to detect patterns composed of non-adjacent alarms.

To improve the detection and elimination of redundant alarms, this paper proposes a new approach which is based on modeling and analysis techniques of discrete event systems (DES). First, two systematic relations, termed domination and mutual dependency, between the alarms of a sequence are automatically discovered. Second, the discovered relations are analyzed by an expert to validate redundancies. This analysis relies on the representation of the relations in the form of Petri nets (PNs) to ease understanding.

The outline of the paper is the following: An overview of related works is presented at Section 2. Fundamental definitions and the problem statement are given in Section 3. Section 4 focuses on the systematic relations between alarms while Section 5 presents the construction of Petri net models from these relations. A case study which is based on real data coming from a thermal power plant is performed in Section 6. Finally, concluding remarks and prospects are drawn up in the last section.

II. RELATED WORKS

Alarm flows avoidance (or reduction) is an up-to-date concern in industry and is one of the objectives of what is generally called *alarm rationalization*. Alarm rationalization is defined by ANSI/ISA 18.2 as "The process of ensuring an alarm meets the requirements set forth in the alarm philosophy,

Y. Laumonier is with General Electric Digital Foundry Europe and with LURPA, ENS Cachan, Univ. Paris-Sud, Université Paris-Saclay, 94235 Cachan, FRANCE (e-mail: yannick.laumonier@ens-cachan.fr).

J.-M. Faure is with LURPA, ENS Cachan, Univ. Paris-Sud, Supméca, Université Paris-Saclay, 94235 Cachan, FRANCE (e-mail: jean-marc.faure@ens-cachan.fr).

J.-J. Lesage, is with LURPA, ENS Cachan, Univ. Paris-Sud, Université Paris-Saclay, 94235 Cachan, FRANCE (e-mail: jean-jacques.lesage@ens-cachan.fr).

H. Sabot is with General Electric Digital Foundry Europe, 75002 Paris, FRANCE (e-mail: herve.sabot@ge.com).

including the tasks of prioritization, classification, settings determination, and documentation". In this context, generic guidelines to reduce the operator load and improve the way the alarms are proposed in [3].

Several academic contributions to this issue deserve also to be mentioned. A good survey of these works is given in [4]. The aim of the works which are referred to in this overview is to avoid a wrong configuration of the alarms or the identify relations between alarms or to identify the root causes of the occurring alarms.

For instance, temporal relations between alarm occurrences are looked for in [5] in order to build a graph which models the statistically related alarms. [6] proposes a multi-temporal sequence mining-based algorithm to detect related alarms and develop suppression rules. [7] uses Bayesian networks to model the causes of alarms and to help the expert interpretation thanks to a graphical representation.

However, it must be noted that all these contributions are based on statistical or probabilistic models that require a careful expert analysis. Discovery of systematic (i.e. always true) relations provides more meaningful results which are easier to interpret. This is, for instance, the case in [8] where the behavior of a process is represented in the form of a Petri net by analyzing event logs with process mining techniques. This observation justifies our approach.

III. DEFINITIONS & PROBLEM STATEMENT

Alarms can be seen as Boolean variables whose value is 1 when the corresponding alarm is active and 0 when it is inactive. The occurrence of an alarm can therefore be seen as the rising edge of the associated variable and the disappearance of an alarm as the falling edge of this variable. Alarms appearances are always the result of anomalous evolutions of the process whereas their disappearances can be due to the process evolution or to be the consequence of an operator acknowledgment. Therefore, since the goal of the work is to find systematic relations between alarms generated by the process, only the alarm appearances are considered. To meet this objective, discrete event systems (DES) paradigms and techniques are used; they are recalled below.

A. Definitions

Alarm occurrences, noted a_i, are considered as rising edge events whose occurrences are instantaneous and asynchronous. Hence, simultaneous alarm occurrences are not possible. Moreover, physical time is not considered; only the order of alarm occurrences is kept to build an alarm sequence. Fig.1 shows a short example of a succession of alarms occurrences generated by a system. It is extracted from an industrial case study and will be treated in section 4.
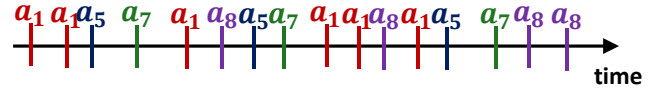

Figure 1: Example of an alarm sequence

Definition 1: Alarm sequence

An alarm sequence is an ordered succession of alarm occurrences.

The length of an alarm sequence S is noted $|S|$. The succession of alarms depicted on Fig. 1 is an alarm sequence $S = a_1 a_1 a_5 a_7 a_1 \ldots$ of length 16 that contains for example six occurrences of a_1 and three occurrences of a_7.

Definition 2: Alphabet of alarms

The set of all alarms a_i whose occurrences constitute a sequence S is called the alphabet of the sequence S and is noted Σ_S.

$$\Sigma_S = \{ a_1, a_2, \ldots, a_i, \ldots, a_{|\Sigma_S|} \} \tag{1}$$

The alphabet of the alarm sequence presented at Fig.1 is $\Sigma_S = \{ a_1, a_5, a_7, a_8 \}$. Σ_S^* is the set of all possible sequences that can be defined on the alphabet Σ_S.

Definition 3: Projection of a sequence

The projection of a sequence S on an alphabet Σ, $Proj_\Sigma(S)$, consists of replacing all $a_i \notin \Sigma$ by the identity element of the concatenation[1] operation ε [10].

If $S = a_1 a_1 a_5 a_7 a_1 a_8 a_5 a_7 a_1 a_1 a_8 a_1 a_5 a_7 a_8 a_8$ and $\Sigma = \{ a_1, a_7 \}$, then:

$$Proj_\Sigma(S) = a_1 a_1 \varepsilon a_7 a_1 \varepsilon \varepsilon a_7 a_1 a_1 \varepsilon a_1 \varepsilon a_7 \varepsilon \varepsilon$$
$$= a_1 a_1 a_7 a_1 a_7 a_1 a_1 a_1 a_7 \tag{2}$$

B. Problem statement

By using the definitions and notations given above, the problem of alarm flow reduction can now be reformulated as follows: given an alarm sequence S observed during the operation of an industrial process $\mathcal{P}$, the goal is to find a reduced alarm sequence $\tilde{S}$ such that $|\tilde{S}| \ll |S|$, with $\Sigma_{\tilde{S}} \subseteq \Sigma_S$, and such that this reduction does not induce a loss of safety on $\mathcal{P}$.

To meet this goal, two sub-problems can be successively dealt with:

- The determination of all alarms that could be suppressed to go from S to $\tilde{S}$. For that, redundant information are looked for in Σ_S under the form of systematic relations linking some of the alarms, in such a way that the suppression of those alarms implies no loss of information related to the dysfunctional behavior of the system.

---

[1] It is reminded that the concatenation of two sequences is a non-commutative operation that builds a sequence which is composed of the first sequence followed by the second one.

- The choice of alarms that can be suppressed without loss of safety. This step requires an expertise of the process $\mathcal{P}$ and is not developed in the paper. Nevertheless, to ease the analysis of the expert, it is shown in section V that the relations discovered in the previous step can easily be modeled by Petri nets.

An alarm sequence observed during a long period of operation of a process is not only a very long succession of alarm occurrences, but also contains lots of information related to the history of the dysfunctional behavior of the process. The goal of the first step of our approach is to extract from S systematic relations between some alarms that express redundant dysfunctional information. For that, systematic causality relationships between alarms are searched under the form of sufficient conditions such as "it is sufficient to observe a to know that a AND b" or "it is sufficient to observe a OR b to know that a AND b". The formal definitions of these relations are given in the next section.

IV. SYSTEMATIC RELATIONS BETWEEN ALARMS

The two causality relations presented in this section have been first presented in [9] for DES identification purposes and have been adapted to the current issue.

A. Domination

Definition 4: Domination

Let S be an alarm sequence defining an alphabet Σ_S and a_i and a_j be two different alarms of Σ_S that define two singleton alphabets $\{a_i\}$ et $\{a_j\}$.

a_i dominates a_j, noted $a_i \rightarrow a_j$, iff the projection of S on $\{a_i\} \cup \{a_j\}$ is an alternation between a non-null length sequence of occurrences of a_i and only one occurrence of a_j:

$$Proj_{\{a_i\} \cup \{a_j\}}(S) = \sigma_{i,1} a_j \sigma_{i,2} a_j \sigma_{i,3} a_j \sigma_{i,4} \ldots \quad (3)$$

with $\forall\, k \in \mathbb{N}^*, \begin{cases} \sigma_{i,k} \in \{a_i\}^* \\ |\sigma_{i,k}| \geq 1 \end{cases}$

a_i dominates a_j implies that before every occurrence of a_j at least one occurrence of a_i has been observed. Therefore, the observation of a_j is sufficient to know that the dysfunctional behavior of $\mathcal{P}$ has been $a_i a_j$. Thus, all a_i occurring before an occurrence of a_j in $Proj_{\{a_i\} \cup \{a_j\}}(S)$ can be suppressed in S.

Example 1: When applying this domination relations search on the sequence of Fig. 1:

$S = a_1 a_1 a_5 a_7 a_1 a_8 a_5 a_7 a_1 a_1 a_8 a_1 a_5 a_7 a_8 a_8$, for the two alarms a_1 and a_5, it comes:

$$Proj_{\Sigma_i \cup \Sigma_j}(S) = a_1 a_1 a_5 \varepsilon a_1 \varepsilon a_5 \varepsilon a_1 a_1 \varepsilon a_1 a_5 \varepsilon \varepsilon \varepsilon$$

$$= a_1 a_1 a_5 a_1 a_5 a_1 a_1 a_1 a_5$$

$$= \sigma_{i,1}\, a_5 \sigma_{i,2} a_5\, \sigma_{i,3} a_5$$

$$\Rightarrow a_1 \rightarrow a_5$$

As $a_1 \rightarrow a_5$, the reduced sequence $\tilde{S}$ is obtained from the original sequence by removing every occurrence of a_1 which is followed by one occurrence of a_5.

$$\tilde{S} = a_1 \cancel{a_1} a_5 a_7 \cancel{a_1} a_8 a_5 a_7 a_1 a_1 a_8 \cancel{a_1} a_5 a_7 a_8 a_8$$

$$= a_1 a_5 a_7 a_8 a_5 a_7 a_1 a_1 a_8 a_5 a_7 a_8 a_8$$

The physical interpretation of the domination relation can be explained by taking the example of the process $\mathcal{P}$ on which S has been observed. In this process, a_1 is an alarm meaning that a "High pressure level" of a fluid is reached, and a_5 means that this fluid reached a "Very high pressure level". The relation "a_1 dominates a_5" expresses the fact that before reaching a "very high" level, the pressure has to reach the "high" pressure level. Giving to an operator the information that a "high" and then a "very high" pressure level has been reached does not give more information than only signaling that a "very high" pressure level has been reached. Therefore, the alarm "High pressure level" can be removed from the alarm sequence when it precedes the alarm "Very high pressure level" without loss of information related to the dysfunctional behavior of the process.

The search of domination relations has to be performed for all couples $(\{a_i\}, \{a_j\})$ to obtain the shortest reduced sequence $\tilde{S}$. In the sequence given in example 1, three domination relations are found: $a_1 \rightarrow a_5$, $a_1 \rightarrow a_7$, and $a_5 \rightarrow a_7$.

B. Mutual dependency

Definition 5: Mutual Dependency

Let S be an alarm sequence defining an alphabet Σ_S, and a_i and a_j be two different alarms of Σ_S, that define two singleton alphabets $\{a_i\}$ et $\{a_j\}$.

a_i and a_j are mutually dependent, noted $a_i \leftrightarrows a_j$, iff the projection of S on $\{a_i\} \cup \{a_j\}$ is an alternation between a single occurrence of a_i and a single occurrence of a_j. The mutual dependency is defined by (4) and is similar to a two-way domination:

$$\begin{cases} Proj_{\{a_i\} \cup \{a_j\}}(S) = a_i a_j a_i a_j a_i \ldots \\ \qquad\qquad or \\ Proj_{\{a_i\} \cup \{a_j\}}(S) = a_j a_i a_j a_i a_j \ldots \end{cases} \quad (4)$$

a_i and a_j are mutually dependent implies that before or after every occurrence of a_i (resp. a_j) one and only one occurrence of a_j (resp. a_i) is observed. Therefore, the observation of a_i or a_j is sufficient to know that the dysfunctional behavior of $\mathcal{P}$ is $a_i a_j$ (or $a_j a_i$). In that case, one of the two alarms a_i or a_j can be removed from S. For example, it is possible to choose to keep the first alarm occurring in the sequence $Proj_{\{a_i\} \cup \{a_j\}}(S)$.

Example 2: When applying this mutual dependency relations search on the sequence of Fig. 1:

$S = a_1 a_1 a_5 a_7 a_1 a_8 a_5 a_7 a_1 a_1 a_8 a_1 a_5 a_7 a_8 a_8$, for the two alarms a_5 and a_7, it comes:

$$Proj_{\{a_5\} \cup \{a_7\}}(S) = \varepsilon\varepsilon a_5 a_7 \varepsilon\varepsilon a_5 a_7 \varepsilon\varepsilon\varepsilon\varepsilon a_5 a_7 \varepsilon\varepsilon$$

$$= a_5 a_7 a_5 a_7 a_5 a_7$$

$$\Rightarrow a_5 \leftrightarrows a_7$$

As $a_5 \leftrightarrows a_7$, a_5 or a_7 can be suppressed to obtain a reduced $\tilde{S}$. a_5 being the first alarm to occur, only this alarm is kept. This gives the following reduced sequence:

$$\tilde{S} = a_1 a_1 a_5 \cancel{a_7} a_1 a_8 a_5 \cancel{a_7} a_1 a_1 a_8 a_1 a_5 \cancel{a_7} a_8 a_8$$

$$= a_1 a_1 a_5 a_1 a_8 a_5 a_1 a_1 a_8 a_1 a_5 a_8 a_8$$

The physical interpretation of the domination relation can be explained by taking the example of the process $\mathscr{P}$ on which S has been observed. In this process, a_5 is an alarm meaning that a "Very high pressure level" of a fluid is reached and a_7 that the "Exit flow is too fast". The fact that a_5 and a_7 are mutually dependent reflects the fact that the exit flow is too fast when the pressure reaches the "very high" level, and vice versa. Both alarms give therefore the same information to the operator, but with different words. Therefore, one of those alarms can be suppressed. For example it makes sense to keep the first alarm that occurs as it allows the problem to be pointed out sooner, but it also makes sense as the "very high" pressure can be considered as the cause of the exit flow "too fast".

Such an expert analysis is needed to confirm the suppression of alarms, once all domination and mutual dependency relations have been identified. A global view of all relations between alarms is indeed mandatory in order to guarantee consistency of the removing decisions. For example, if we find in S:

$$\begin{cases} a_1 \rightarrow a_5 \ (1) \\ a_5 \leftrightarrows a_7 \ (2) \end{cases}$$

From (1) we deduce that a_1 can be suppressed when it occurs before a_5.

Then, by considering only (2), we may deduce that either a_5 or a_7 can be suppressed. However, but taking into account also (1), it is no more possible to remove a_5 without modifying the observed dysfunctional behavior (the domination relation would be forgotten in that case).

To provide this global view of the relation between alarms, the next section shows how the domination and mutual dependency relations can be translated into a graphical and formal representation: autonomous Petri nets.

V. TRANSLATION INTO PETRI NETS

This section first recalls the definition and the notations of autonomous Petri nets (PN), then shows how the systematic relations can be translated into Petri nets.

A. Autonomous Petri Net Definitions and Notations

An autonomous Petri net is a directed bipartite graph with two kinds of vertices: places and transitions. Places are drawn as circles and transitions are drawn as bars. Directed arcs go from a place to a transition or vice versa.

Formally, an ordinary Petri net is defined by a 4-tuple: $PN = (P, T, I, O)$ where:

- $P = \{p_1, p_2, \dots, p_{|P|}\}$ is a finite set of places,
- $T = \{t_1, t_2, \dots, t_{|T|}\}$ is a finite set of transitions,
- $\begin{cases} P \cap T = \emptyset \\ P \cup T \neq \emptyset \end{cases}$
- $I: P \times T \rightarrow \{0,1\}$ is a function representing the existence of arcs going from places to transitions,
- $O: T \times P \rightarrow \{0,1\}$ is a function representing the existence of arcs going from transitions to places,

A marking function $M: P \rightarrow Z^+$ (Z^+ being the set of nonnegative integers) assigns to each place of P a number of tokens; the marking of a PN is usually expressed as a $|P|$-entry vector. M_0 denotes the initial marking of a marked $PN = (P, T, I, O, M_0)$. In a PN, a transition t_j is enabled at marking M_k if $\forall p_i \in P$, $M_k(p_i) \geq I(p_i, t_j)$, an enabled transition t_j can be fired, reaching a new marking M_{k+1}. This behavior is represented as $M_k \overset{t_j}{\rightarrow} M_{k+1}$. The new marking can be computed as $M_{k+1} = M_k + (O - I)u_k$, where $u_k(i) = 0$, $i \neq j$, and $u_k(j) = 1$; this equation is called the PN state equation. The set of all possible reachable markings from M_0 by firing only enabled transitions is called the reachability set of a marked PN.

In a PN, a source transition is a transition without any input place; it is unconditionally enabled. A sink transition is a transition without any output place; its firing consumes but does not create any token.

B. Modeling Relations between Alarms by Petri Nets

Each one of the domination and mutual dependency properties expresses a sequentiality or a concurrency relationship between alarms that can be represented by a generic PN structure, called a Petri net pattern, in which alarms are associated to the transitions.

- Fig. 2a. is an example of the generic PN pattern representing a domination relation (in this case $a_1 \rightarrow a_5$). This PN expresses that the sink transition a_5 (the dominated alarm) can only be fired after the source transition a_1 (the dominant alarm) has been fired.
- Fig. 2b is an example of the generic PN pattern representing a mutual dependency (in this case $a_5 \leftrightarrows a_7$). This PN expresses an iterative sequence of firing of the transition a_5 and the transition a_7. It must be noted that the initial marking of the place P16 imposes that the first enabled transition is a_5, translating that a_5 is the first alarm occurring in the projection of the sequence S on $\{a_5\} \cup \{a_7\}$.

Let now consider the set of all relations between alarms found in the sequence S given in Fig.1:

$$\begin{cases} a_1 \rightarrow a_5 \ (a) \\ a_1 \rightarrow a_7 \ (b) \\ a_5 \leftrightarrows a_7 \ (c) \\ no \ relation \ for \ a_8 \ (d) \end{cases}$$

The translation of these four relations in PN is given in Fig 2c. This PN is composed of two patterns of domination
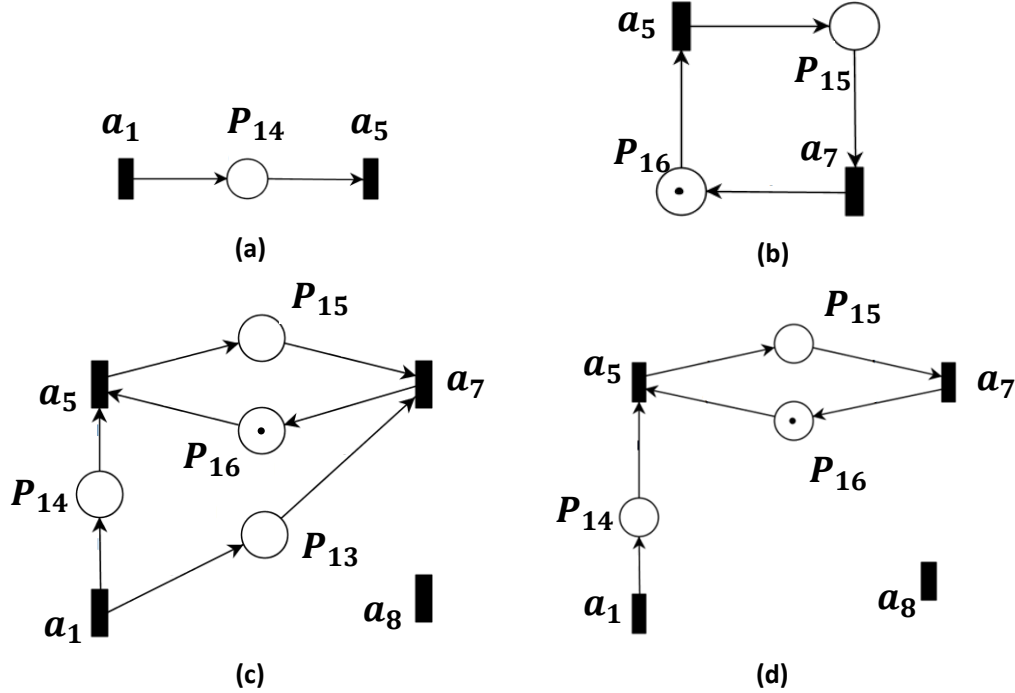
Figure 2: a) PN pattern of a domination relation, (b) PN pattern of a mutual dependency relation
(c) PN representing all relations found in S defined in Fig. 1, (d) PN representing all relations found in S after simplification

relation (relations a and b above) and one pattern of mutual dependency (relation $\underline{c}$ above), connected by fusing the two transitions related to the same alarm (a_5). Alarm a_8 does not have any relation with the other alarms: it is therefore represented as a source and sink transition which is not connected to the rest of the PN. This means that a_8 is independent of the others alarms and does not need this analysis. Finally, this PN can be simplified, by observing that place P13 is an implicit place, i.e. a place whose suppression does not change the reachability set of the PN [11]. The resulting simplified PN is given in Fig. 2d. It provides a graphic global view of all domination and mutual dependency relations found in S and is therefore well adapted to the expert subsequent work that consist in making choices of which alarms can be removed without loss of safety. An example of such analysis and its benefits in terms of alarm flow reduction is given in the next section.

VI. INDUSTRIAL EXAMPLE

The method that is presented in this paper consist in the discovery of systematic relations between alarms, followed by the representation of the discovered relations in the form of Petri net patterns that can be analysed later on by experts to remove redundant alarm occurrences. This method has been applied to a real alarm log coming from the power industry.

This alarm log has been recorded during one month to guarantee that the obtained results are representative enough. It includes 55,000 alarm occurrences and is made up of 489 alarms ($|S| = 55{,}000$ and $|\Sigma s| = 489$).

A. Global results

The algorithm to discover systematic relations between alarms must detect, for every couple of alarms, whether a systematic, i.e. for the whole sequence, domination or mutual dependency relationship exists or not. This is done by browsing the sequence for each alarm couple to look for those relations. The theoretical complexity of this algorithm is therefore $\mathcal{O}(|\Sigma_S|^2|S|)$. Nevertheless, this algorithm scales well on the example. The global analysis of the considered alarm sequence was performed in 45 minutes, a quite acceptable duration for such an example in industrial context.

The automatic analysis of the considered alarm log by this algorithm returns 36 patterns (containing 141 alarms presenting some relations between them) and 348 independent alarms (alarms associated to transitions which are both source and sink, like a8 in the previous section). The most frequent pattern is composed of 2 alarms and occurs 1,051 times while the least frequent pattern is composed of 4 alarms and occurs only 11 times.

Using these identified patterns, expert analysis allows a 5% reduction of the whole alarm sequence length. This is not the most significant metrics because the number of alarms per time unit is not constant during the observation time. Therefore, the same analysis must be done during the time period when the operators are the most solicited. The result is that a 38% reduction of the length of the alarm sub-sequences was achieved during the 10 minutes periods with the largest number of alarms and constitute really an alarm flood. Hence, alarm flow reduction is real.

B. Comparison to previous work

The method presented in this paper can be also compared on the basis of this example to a previous work which is detailed in [2]. The selected approach was based on the pattern mining algorithm *AprioriAll* [12] and was aiming at extracting from an alarm sequence frequent patterns composed of *adjacent* alarms from an alarm sequence. Related alarms were obtained from expert analysis of these frequent patterns. Both methods share the same hypotheses: alarms as events whose occurrences are instantaneous and asynchronous and only logical time is considered. Both start from an alarm sequence, too. Therefore, they can be reasonably compared (Table I).

TABLE I. COMPARISON BETWEEN SYSTEMATIC RELATIONS DISCOVERY AND FREQUENT PATTERNS DETECTION

	Systematic relations discovery	Frequent patterns detection
Computation time	45 minutes	2 hours 45 minutes
Number of discovered patterns	41 patterns	More than 1,500 patterns
Expert analysis	Guided by the Petri net patterns	Only based on the patterns frequency

The computation time of the discovery of systematic relations is more than three times smaller than the one taken by the detection of frequent patterns. Nevertheless, both are acceptable in the industry. The main benefits of the new approach are, in our opinion, the following ones:

1. It provides to the expert a rather small number of patterns. Expert analysis is therefore more focused.
2. These patterns are represented in a graphical manner which eases expert analysis.
3. It allows to detect related alarms that are not adjacent.

In other words, frequent patterns detection yields a large set of patterns that are usually composed of a small number (2 or 3) of alarms and is difficult to analyze by an expert to remove the redundant alarms. On the opposite, systematic relations discovery provides a small set of patterns which model systematic (and not only frequent) relations and is easier to analyze. Nevertheless, the strength of this approach is also its limitation. As the patterns represent systematic relations, a rare alarm occurrence may prevent from detecting a relation by interfering with the natural order of the alarm occurrences.

VII. CONCLUSION

Alarm systems play an important role for safety and efficiency of industrial plants. Unfortunately, in modern systems, the operators are often overwhelmed by the number of alarms and cannot react in the due time. Alarm flows reduction is therefore a crucial industrial issue.

To meet this objective, this paper has shown how to automatically discover in alarm logs systematic domination and mutual dependency relations between alarms. These relations can be directly translated into Petri net patterns which are later on analyzed by an expert in order to detect related alarms and identify alarms occurrences that need the same response from the operator. These redundant occurrences can then be removed during future operation. Scalability and efficiency of the proposed approach has been illustrated on an industrial-size case study.

Implementation of this approach in the GE Industrial Internet of Things Predix© environment is currently in progress. Future works aim at extending the relations discovery by considering physical time (and not only logical time). Alarm occurrences are indeed generally time-stamped while these information are neglected in our current approach. Introducing them will surely allow a more accurate analysis of the alarm sequences.

REFERENCES

[1] ANSI/ISA-18.2-2016, "Management of Alarm Systems for the Process Industries". 2009

[2] Y. Laumonier, J.-M. Faure, J.-J. Lesage, and H. Sabot. "Towards Alarm Flood Reduction". 22nd International Conference on Emerging Technologies and Factory Automation (ETFA'17), IEEE, Limassol, 2017.

[3] J. Koene, and V. Hiranmayee. "Alarm management and rationalization." Proceedings of the Third International conference on loss prevention, Singapore, 2000.

[4] J. Wang, et al. "An Overview of Industrial Alarm Systems: Main Causes for Alarm Overloading, Research Status, and Open Problems". IEEE Transactions on Automation Science and Engineering **13(2)**, pp. 1045-1061, 2016.

[5] J. Kinghorst, M.F. Pirehgalin and B. Vogel-Heuser. "Graph-based Grouping of Statistical Dependent Alarms in Automated Production Systems". IFAC-PapersOnLine, **51(24)**, pp.395-400, 2018.

[6] G. Dörgő, J. Abonyi. "Sequence Mining based Alarm Suppression". IEEE Access, **PP(99)**, pp. 1-1, 2018

[7] P. Wunderlich, and O. Niggemann. "Structure learning methods for Bayesian networks to reduce alarm floods by identifying the root cause". 22nd International Conference on Emerging Technologies and Factory Automation (ETFA'17), IEEE, Limassol, 2017.

[8] W.M.P. Van Der Aalst, and B. F. Van Dongen. "Discovering Petri nets from event logs." In Transactions on Petri Nets and Other Models of Concurrency VII, Springer, pp. 372-422, 2013.

[9] J. Saives, G. Faraut, and J.-J. Lesage. "Identification of discrete event systems unobservable behaviour by Petri nets using language projections". European In Control Conference (ECC), IEEE, Linz, pp. 464-471, 2015.

[10] R. David, and H. Alla. "Discrete, continuous, and hybrid Petri nets". Springer Science & Business Media, 550 p., 2010.

[11] M. Silva. "Introducing Petri nets." In Practice of Petri Nets in manufacturing, Springer, pp. 1-62, 1993.

[12] R. Agrawal, and R. Srikant. "Mining sequential patterns". Data Engineering. Proceedings of the Eleventh International Conference on. IEEE, Taipei, pp.3-14, 1995.