Preprints, 10th IFAC International Symposium on
Advanced Control of Chemical Processes
Shenyang, Liaoning, China, July 25-27, 2018

Extraction and Graphical Representation of Operator Responses to Multivariate Alarms in Industrial Facilities ⋆

Wenkai Hu * **Ahmad W. Al-Dabbagh** * **David Li** * **Tongwen Chen** *

* *Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada T6G 1H9*
e-mail: {wenkai, aaldabba, shicheng, tchen}@ualberta.ca

Abstract: Expert knowledge is an important factor to achieve operational effectiveness. This work focuses on mining such knowledge on operator responses to alarms, and examining the relations between the responses and alarms from the historical Alarm & Event logs, which are commonly available in modern industrial facilities. The process mining is adapted and applied to construct dependency matrices, based on which workflow models of the operator responses to alarms are discovered. Also, a new framework for graphical representation of operator responses is proposed to give a better visualization of the extracted workflow models. To demonstrate the effectiveness of the method, an industrial case study is presented.

Keywords: Alarm systems, industrial facilities, process discovery, operator responses

1. INTRODUCTION

Modern industrial facilities are usually very large in scale and complex in process operation. The interconnection of a large number of components, such as vessels, pipes, actuators, sensors, and control modules, forms paths for the propagation of abnormalities, which usually result in multiple alarm annunciations and even alarm floods. In practice, too many alarms may overload operators and make them fail to respond to alarms in a timely and corrective manner. For example, as mentioned in (Mattiasson, 1999), it is highly unlikely for operators to read through an alarm response manual during an alarm flood that requires immediate attention. If theses alarms are not addressed timely, the abnormal operation can lead to unexpected plant shut-down, economic loss and even loss of life as seen in many plant failures (Goel et al., 2017). Thus, there is a high demand to develop tools to assist operators in taking timely and corrective responses to address alarms.

Computerized control and information systems, such as the Distributed Control Systems (DCSs) and the Supervisory Control And Data Acquisition (SCADA) systems, have been broadly deployed in large-scale industrial facilities, making the collection of large volumes of process operation data and events an easy task. Such historical data contains valuable information about process operation status, propagation of abnormalities, and operator actions in response to alarms. Extracting knowledge from historical data would be helpful for decision support during process monitoring (Goel et al., 2017). The existing research mainly focused on analyzing alarm data to tackle alarm floods and nuisance alarms (Schleburg et al., 2013; Hu et al., 2017), whereas historical events related to operator responses are not sufficiently investigated.

The human factor is crucial in alarm monitoring in plentiful studies in the area of ergonomics (Nimmo, 2002; Stanton and Baber, 2008; Adhitya et al., 2014). The goal of any alarm system is to effectively display the state of the plant to the operator, and the operator should be able to clearly identify which alarm requires immediate attentions and how to fix the abnormalities. Historical records about such interactions between alarm systems and operators make it possible to find out how operators respond to alarms and how operational procedures are conducted routinely. The workflow mining, or also known as process mining (Van der Aalst et al., 2004), provides solutions to gain such insights from the historical data. By extracting all variations of procedures and turning them into understandable models, the real execution can be discovered.

This paper attempts to adopt the workflow mining into the field of alarm management to learn how operators respond to alarms. Some related studies are: Dasani et al. (2015) created workflow models from event messages from a boiler operation and applied conformance checking to the resulting model to extract unique findings. Hu et al. (2016) applied process discovery to text based messages to capture responses for univariate alarms and displayed the results using petri-nets. Based on these ideas, this work explores workflow mining of operator actions in response to mutivariate alarms. The major contributions are to propose: 1) a systematic method to capture relationships between operator responses and alarms, and 2) a new graphical visualization framework to represent the extracted workflow models. Such workflow models can be used to assist operators in decision making. As a result, operators can take actions promptly and correctly to tackle abnor-

⋆ This work was supported by the Natural Sciences and Engineering Research Council of Canada via the CRD program.

Table 1. Common operator actions in industrial facilities.

Type	Examples
Primary actions	1) acknowledging, shelving, or suppressing alarms; 2) regulating pumps, valves, motors or fans; 3) adjusting parameters, set points, or alarm limits; 4) switching operating states and control modules.
Secondary actions	1) communicating with other workers; 2) scheduling tasks contemporaneously; 3) checking operation manuals or notes; 4) thinking about causes and solutions.

malities by referring to the corrective responses distilled from the historical data.

The remainder of the paper is organized as follows. Section 2 gives a brief introduction to common operator responses and an industrial event log of alarm messages and operator actions. In Section 3, a variant of the heuristics miner is used and adapted to capture the dependencies between alarms and operator responses. A new framework for graphical representation of workflow models is also proposed in this section. Section 4 presents an industrial case study to demonstrate the effectiveness of the proposed methods, followed by concluding remarks in Section 5.

2. PRELIMINARIES

This section introduces common operator responses to handle alarms in industrial facilities, and discusses the historical data involving both alarm messages and operator response events.

2.1 Operator Responses to Alarms

Alarms are generated by alarm systems in the form of visual or audible notifications to warn operators of abnormalities. Accordingly, operators take actions to respond to the alarms and bring the system back to the normal state. As summarized in (Rothenberg, 2009; Hu et al., 2016), there are two major types of operator actions (shown in Table 1) in industrial facilities.

Primary actions are usually deterministic and can be easily historized and used for off-line analysis. By contrast, secondary actions are uncertain and usually performed before or after certain primary actions. Secondary actions are considered as non-standard and non-documented actions and usually cannot be historized by computerized control and information systems. In view of that, this work only studies primary actions, which are usually available in the historical data. To evaluate which actions are associated or important to an alarm, metrics are needed to measure the dependencies between actions and alarms. This problem is addressed in the next section.

Fig. 1 presents a single tank system, where the input flow rate is controlled by the pump and the output flow rate is controlled by the valve. If the ratio between the pressure of the pump and the position of the value exceeds the normal operating range, the level in the tank may become too high or too low, and this may trigger an alarm. Therefore in this system, three different variables are inter-related. In the scenario where the level is above the alarm limit, several

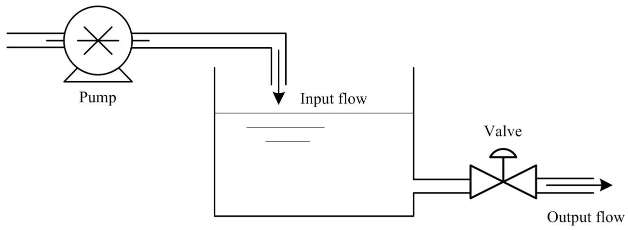

Fig. 1. A single tank system.

operating procedures could be executed by the operator to handle such an alarm:

(1) Increasing the opening of the valve at the bottom of the tank;
(2) Reducing the speed of the pump or shutting down the pump, while keeping the valve open;
(3) Increasing the alarm limit of the tank fluid level.

In this paper, the goal is to determine which actions are commonly used and which actions are more effective in addressing alarms. Such knowledge is usually hidden in the historical data, namely, the Alarm & Event (A&E) logs in the next subsection.

2.2 Alarm & Event Logs

An A&E log is a structured database of time stamped textual messages for events, such as the alarm state transitions and operator responses (Hu et al., 2016, 2017). The common data attributes in an A&E log include the time stamp, tag name, alarm identifier, event type, plant area, priority, state transition, and description. Among them, configuration attributes, such as the tag name, alarm identifier, event type, plant area, and priority are fixed and uniquely identify a process or an instrument. By contrast, other data attributes, such as the time stamp and state transition, may vary with time and indicate what events happened and when they happened.

The data attributes enable the exploitation for multiple purposes in off-line analysis. Since the goal of this paper is to extract information about operator responses to alarms, only related data attributes are used. More specifically, in this paper an event is referred to as a 3-tuple, consisting of three data attributes, including the time stamp, event type, and activity, which are described as follows:

(1) Time stamp defines when exactly an event happened in the studied time period and also defines the orders of events.
(2) Event type defines whether the event is related to an alarm or a response.
(3) Activity defines the state transition, such as an alarm occurrence (ALM) and return-to-normal (RTN) of a specific alarm, or a change of a specific operator response, (e.g., opening or closing a valve).

In some cases, unprocessed A&E logs (namely, raw files) may have many imperfections, such as missing records, inconsistent time formats, and improper descriptions (Suriadi et al., 2017). Thus, prior to extraction of information, such imperfections should be resolved, and therefore, a processed A&E logs is more useful.

3. METHODOLOGY

This section presents the process mining technique which extracts workflow models of operator responses to multivariate alarms from historical A&E logs, and a new framework for graphical representation of operator responses.

3.1 Preliminaries on Process Mining

As introduced in (Van der Aalst, 2011a), the process mining is a relatively young research area sitting between machine learning and data mining. The basic idea of process mining is to discover, monitor, and improve real processes by extracting knowledge from historical event logs. It consists of 3 main tasks, including process discovery, conformance checking, and process enhancement (Van der Aalst, 2011b). Fueled by the availability of large amounts of event data, process mining has been broadly applied to solve real world problems. Meanwhile, a variety of process mining algorithms and softwares have been developed. Some of the fundamental algorithms are, but not limited to, the alpha miner (Van der Aalst et al., 2004), heuristics miner (Weijters et al., 2006), fuzzy miner (Günther and Van der Aalst, 2007), and genetic miner (de Medeiros et al., 2007). Over the past few years, many revised versions of these algorithms have been developed for performance improvement or special applications (Weijters and Ribeiro, 2011; Wen et al., 2006; de Medeiros et al., 2003).

In industrial facilities, there could be different ways to respond to the same alarms and clear the abnormal situations. Such dependencies between alarms and responses are stored in the A&E logs. In this work, the heuristic miner algorithm is exploited to extract workflow models of operator actions in response to multivariate alarms. Compared to the most common process mining algorithm, namely, the α-algorithm, the heuristic miner is more robust to noises (Weijters et al., 2006; Weijters and Ribeiro, 2011). Moreover, using this algorithm, it is not necessary to segment the A&E log into traces of events, which is a major difference to the univariate analysis case in (Hu et al., 2016), where A&E log needs to be segmented into many traces and the "head" or "tail" of each trace of events should be identified from the data. Thus, this property makes the heuristic miner more appropriate for the case with multivariate alarms.

3.2 Construction of Dependency Relations

According to (Weijters et al., 2006; Weijters and Ribeiro, 2011), an event log should be analyzed for causal dependencies, which are usually reflected by the orders of events (alarms or responses) in the A&E log file. Thus, some basic notations are introduced first. Given an A&E log $\mathbb{D} = \{x_1\ x_2\ x_3\ \ldots\ x_N\}$ defined on the event domain $\mathcal{X}$ (including alarms and responses), the following log-based relations are defined:

- Direct successor: $A > B$ iff there exists $i \in \{1, 2, \ldots, N-1\}$ such that $x_i = A$ and $x_{i+1} = B$ within $\mathbb{D}$;
- Length-two loops: $A \gg B$ iff there exsists $i \in \{1, 2, \ldots, T-2\}$ such that $t_i = A$, $t_{i+1} = B$ and $t_{i+2} = A$ within $\mathbb{D}$;
- Direct or indirect successor: $A \ggg B$ iff there exsists $i < j$ and $i, j \in \{1, 2, \ldots, N\}$ such that $t_i = A$ and $t_j = B$ within $\mathbb{D}$.

where "iff" abbreviates for "if and only if", and N indicates the number of events in $\mathbb{D}$.

Dependency is then defined as the relation that the presence of certain events (alarms and responses) implies the presence of other events (alarms and responses). To be more specific, within the heuristics miner algorithm, five types of dependencies are defined (Weijters et al., 2006; Weijters and Ribeiro, 2011):

Direct dependency: the occurrence of one event directly causes another event to occur. The direct dependency metric is given by

$$A \to B = \frac{|A > B| - |B > A|}{|A > B| + |B > A| + 1} \tag{1}$$

where $|\cdot|$ indicates the frequency of a specific log-based relation. A value close to 1 indicates a strong bi-variate relationship between A and B.

Self dependency: the occurrence of the same event consecutively within an event log. The self dependency metric is given by

$$A \to A = \frac{|A > A|}{\||A > A| + 1} \tag{2}$$

where this type of dependency usually occurs when the same alarm is raised multiple times in succession, or one operator response repeats multiple times.

Length-two dependency: the occurrence of A followed by B is followed again by A (ABA). The length-two dependency metric is given by

$$A \to_2 B = \frac{|A \gg B| - |B \gg A|}{|A \gg B| + |B \gg A| + 1} \tag{3}$$

where this type of dependency can be regarded as a feedback system.

Long distance dependency: similar to direct dependency except, A is followed by B within a window size. The long distance dependency metric is given by

$$A \to_l B = \frac{|A \ggg B| - |B \ggg A|}{|A \ggg B| + |B \ggg A| + 1} \tag{4}$$

AND/XOR dependency: If an event has multiple direct dependencies, those events must be either AND or XOR dependencies. For example, A is followed by B and A is followed by C. Then, an AND relationship may suggest that both B and C occur after A. XOR dependency may suggest that A followed by B and A followed by C are two different dependencies with no relation. The AND/XOR dependency metric is given by

$$A \to B \wedge C = \frac{||B > C| - |C > B||}{|A > B| + |A > C|} \tag{5}$$

Thresholds to imply the above dependencies can be predefined and adjusted by the users based the application requirements. High restrictions will result in over generalized or partial models, while low restrictions will result in unwanted relationships. The user's expert knowledge about the process and interpretation skills can greatly improve the results.

3.3 Implementation of the Heuristics Miner

The key to construct a workflow model using the heuristic miner algorithm is to extract frequency based matrices representing dependency relations and build dependency graphs based on these matrices (Weijters et al., 2006; Weijters and Ribeiro, 2011). Given M unique events (including all unique alarms and responses), the dependency graphs are contracted with the size $M \times M$. The procedures to create dependency matrices are:

Step 1 Check if each unique event meets the minimum observation count. If this condition is not met, these events will not be considered for the remaining steps.
Step 2 Create three dependency graphs, namely, direct dependency graph, length-two dependency graph, and long distance dependency graph, using the log-based relations and metrics in Section 3.2.
Step 3 Check for any negative dependencies and remove them from the dependency matrices.

Next, the following procedures are followed to mine multivariate relationships from the constructed dependency graphs:

Step 1 Find the highest row and column values from all three dependency matrices.
Step 2 Check whether each of the highest row or column values meet the threshold requirements. If this condition is met, then select all dependencies in the respective row or column with dependencies greater than the highest row/column minus the relative to best threshold.

3.4 Graphical Representation

The detected relations of operator responses to multivariate alarms can be represented by workflow models, which are graphical representations in different forms (de Medeiros et al., 2003). In this paper, a new framework for the graphical representation of workflow models of operator responses to alarms is proposed. In this new framework, two main parts are defined:

(1) Nodes: each node indicates a unique event, such as the occurrence of an unique alarm, its return-to-normal, or an associated alarm response. In this framework, nodes related to alarm occurrences or return-to-normal events, are representing by triangle symbols. Nodes related to operator responses or other related events are represented by ellipse symbols.
(2) Edges: each edge indicates the dependency between two events. Three types of edges are defined in this graphical representation: a green solid arrow indicating the forward dependency, a red solid arrow indicating the reverse or feedback dependency, and a dotted arrow indicating a portion of the process model which is connected to the other portion of the process model. In most cases, a well structured or ideal process model should not contain self loops and reverse dependencies. Dashed lines are useful to display large and complex process models.

The following list presents the graphical notations for the representation of workflow models of operator responses to multivariate alarms:

▷ A hollow triangle connecting to the start of an edge, represents the occurrence of an alarm.
◁ A hollow triangle connecting to the end of an edge, represents the return-to-normal event of an alarm.
▶ A solid triangle connecting to the start of an edge indicates the repeated occurrence of an alarm.
◀ A solid triangle connecting to the end of an edge indicates repeated return-to-normal event of an alarm.
$\longrightarrow$ A green solid arrow represents an edge connecting two events in a forward direction.
$\longleftarrow$ A red solid arrow represents an edge connecting two events in a feedback direction.
$\cdots$ A dotted line (either a green or red arrow) connects two partial process models.
⬭ An hollow ellipse represents a unique operator response, and a solid ellipse indicates a repeated operator response.

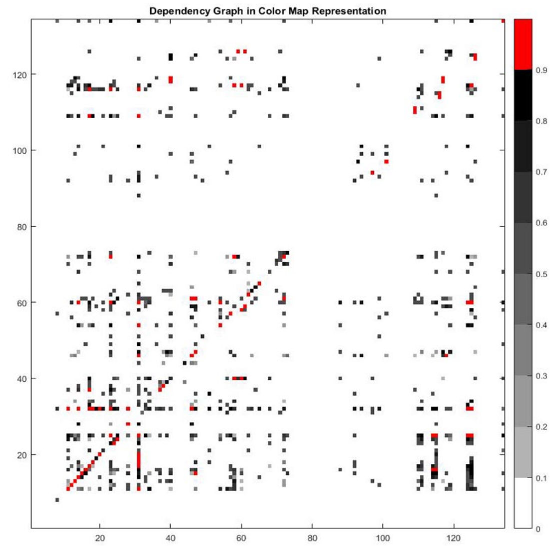

Fig. 2. Color map of the direct dependency matrix. A darker color indicates a higher dependency value. A red color indicates that the dependency is no less than the predefined threshold of 0.9.

To distinguish between different alarms or responses, different colors are used for the above symbols. Since the occurrence and return-to-normal events usually appear in pair for the same alarm tag, the same color should be used for such alarm events. For instance, we can use yellow, orange, and red to denote the "high", "high high", and "very high" alarms for the ALM and RTN of a specific variable. Meanwhile, we can use light blue, blue, and dark blue to denote the "low", "low low", and "very low" alarms for the ALM and RTN of a specific variable.

4. INDUSTRIAL CASE STUDY

The case study involves a real A&E log file, which consisted of 13917 events. The number of unique events including alarms and response was 134, indicating that the average frequency of each event is 104. Using the heuristic miner, the dependency matrices are constructed with a size of 134×134. In order to find significant dependencies between alarms and response, a large threshold of 0.9, is applied to filter out weak dependencies and then construct dependency graphs.

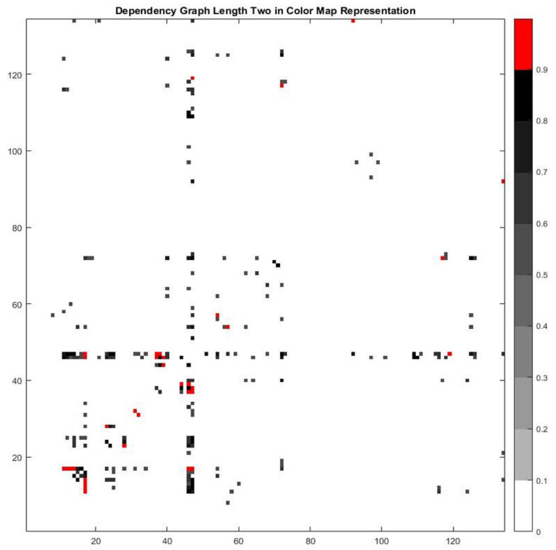

Fig. 3. Color map of the length-two dependency matrix.

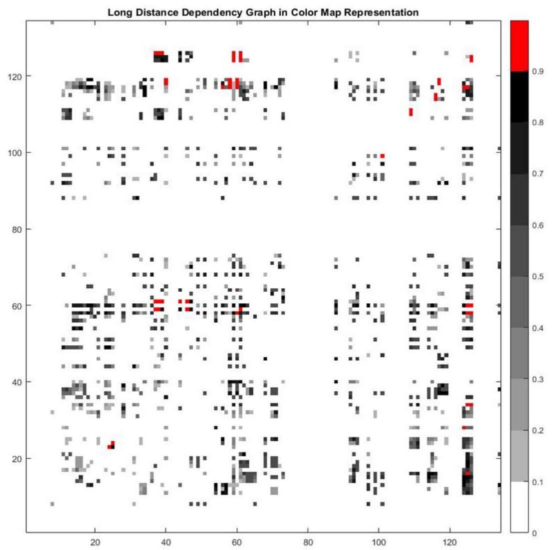

Fig. 4. Color map of the long distance dependency matrix.

The calculated direct dependency matrix, length-two dependency matrix, and long distance dependency matrix are presented using color maps in Figs. 2, 3, and 4. The color of each pixel in the color map indicates the dependency value from the row variable to the column variable. The diagonal indicates the dependency between one variable and itself. The red color indicates that the dependency value is no less than the threshold of 0.9. From the direct dependency matrix in Fig. 2, there is a high number of self-dependencies as can be observed at the diagonal, where quite a few pixels have the red color, indicating significant dependencies. From the length-two dependency matrix in Fig. 3, it can be seen that fewer significant dependencies are found compared to the direct dependency matrix. From the long distance dependency matrix in Fig. 4, a large number of long distance dependencies are found and quite a few are significant, i.e., the dependency values are no less than the predefined threshold.

Based on the significant dependencies found from the dependency matrices, workflow models of operator responses to multivariate alarms can be constructed. Fig. 5 presented four examples of the extracted work flow models, which are displayed using the new graphical representation. The dependencies are automatically detected using the process mining method in this work; the workflow graphs in Fig. 5 are manually drawn based on these dependencies.

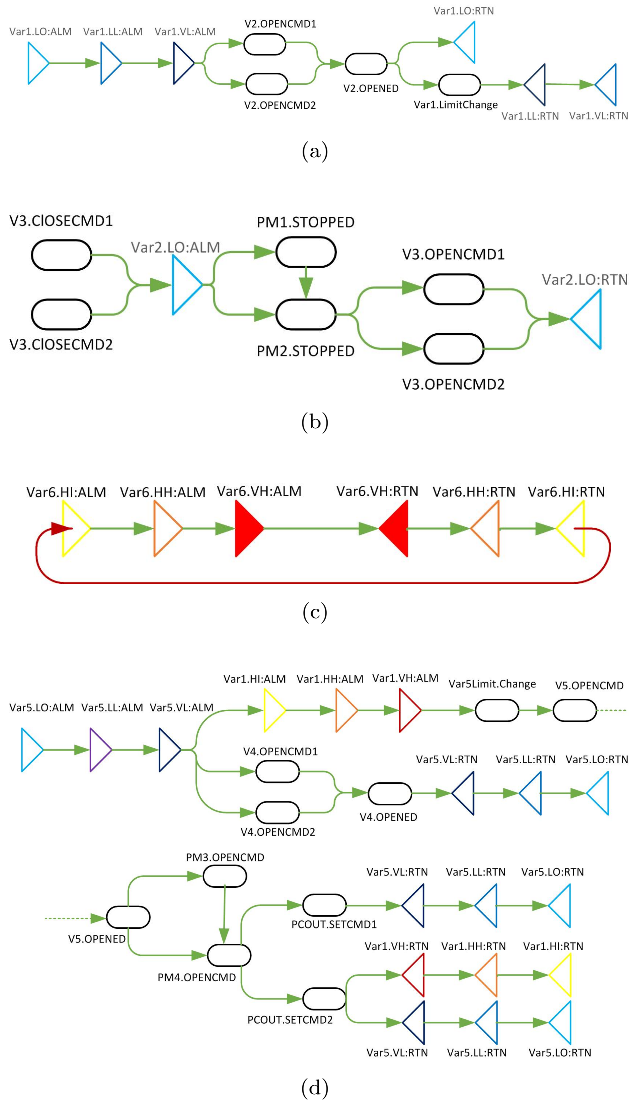

Fig. 5. Constructed workflow models based on the new graphical representations.

- In Fig. 5a, the triangles of different colors indicate three alarms, namely, "Var1.LO", "Var1.LL" and "Var1.VL", which correspond to "low", "low low", and "very low" alarm limits of the same process variable "Var1". As discovered from the A&E log, commands to open the valve "V2" were usually made by operators through two ways indicated by "V2.OPENCMD1" and "V2.OPENCMD2". Then, the valve "V2" opened, leading to the the clearance of "Var1.LO". Meanwhile, an operator changed the alarm limits, which cleared the alarms "Var1.LL" and "Var1.VL".
- From Fig. 5b, it can be seen that the alarm "Var2.LO" was triggered by the command "V3.CLOSECMD1" or "V3.CLOSECMD1" that was used to close valve "V3". Then, Pumps "PM1" and "PM2" stopped.

After that, an operator made commands to open valve "V3", which led to the clearance of the alarm "Var2.LO". This example demonstrates that the method in this work is not only capable of discovering workflow models for multivariate alarms, but also capable of finding those for univariate alarms.

- Fig. 5c presents a special case, where no operator actions or system changes appeared in the extracted workflow model. The alarms "Var6.HI", "Var6.HH", and "Var6.VH" occurred in a sequential order and cleared in a reverse order. After that "Var6.HI" alarm happened again, indicating the repeating of these alarms without the intervention of operators. As indicated by the solid symbols, "Var6.VH:ALM" and "Var6.VH:RTN" were likely to repeat by themself.
- The last example in Fig. 5d presents a more complex case, with six alarm variables and ten alarm responses. Since this graphical model is too large, it is decomposed into two parts, connected by a dotted arrow. Compared to the previous examples, there are two new actions, namely, "PCOUT.SETCMD1" and "PCOUT.SETCMD1", which indicate the operator respones to change the parameters related to the pump control loop.

5. CONCLUSION

This paper presents a framework for the extraction and graphical representation of operator actions in response to multivariate alarms. Process mining is adapted and applied to extract meaningful workflow models, which represent the knowledge of how operators make actions to respond to alarms. Meanwhile, a new framework of graphical representation of workflow models is proposed for a better visualization of the extracted results. Eventually, the results can be used to assist operators in decision making, and thus prevent the reproduction of human errors when responding to alarms. The workflow models captured from the skilled operators can also be used for operator training programs. Further, there remain many research opportunities in this area. Some possible problems deserving efforts include: how to validate the captured the workflow models based on the process knowledge or alarm response manuals, how to incorporate more information, such as the response time, and how to evaluate which response is the best when there are multiple paths of operator actions in response to the same alarms.

REFERENCES

Adhitya, A., Cheng, S.F., Lee, Z., and Srinivasan, R. (2014). Quantifying the effectiveness of an alarm management system through human factors studies. *Computers & Chemical Engineering*, 67, 1–12.

Dasani, S., Shah, S.L., Chen, T., Funnell, J., and Pollard, R.W. (2015). Monitoring safety of process operations using industrial workflows. *IFAC-PapersOnLine*, 48(8), 451–456.

de Medeiros, A.K.A., Weijters, A.J.M.M., and Van der Aalst, W.M.P. (2007). Genetic process mining: an experimental evaluation. *Data Mining and Knowledge Discovery*, 14(2), 245–304.

de Medeiros, A.K.A., Van der Aalst, W.M.P., and Weijters, A.J.M.M. (2003). Workflow mining: Current status and future directions. In *OTM Confederated International Conferences on the Move to Meaningful Internet Systems*, 389–406. Springer.

Goel, P., Datta, A., and Mannan, M.S. (2017). Industrial alarm systems: Challenges and opportunities. *Journal of Loss Prevention in the Process Industries*, 50(Part A), 23–36.

Günther, C.W. and Van der Aalst, W.M.P. (2007). Fuzzy mining-adaptive process simplification based on multi-perspective metrics. In *International Conference on Business Process Management*, 328–343. Springer.

Hu, W., Al-Dabbagh, A.W., Chen, T., and Shah, S.L. (2016). Process discovery of operator actions in response to univariate alarms. *IFAC-PapersOnLine*, 49(7), 1026–1031.

Hu, W., Chen, T., and Shah, S.L. (2017). Discovering association rules of mode-dependent alarms from alarm and event logs. *IEEE Transactions on Control Systems Technology*, PP(99), 1–13.

Mattiasson, C. (1999). The alarm system from the operator's perspective. In *International Conference on Human Interfaces in Control Rooms, Cockpits and Command Centres*, 217–221.

Nimmo, I. (2002). It's time to consider human factors in alarm management. *Chemical Engineering Progress*, 98(11), 30–38.

Rothenberg, D. (2009). *Alarm Management for Process Control: A Best-practice Guide for Design, Implementation, and Use of Industrial Alarm Systems*. NewYork: Momentum Press.

Schleburg, M., Christiansen, L., Thornhill, N.F., and Fay, A. (2013). A combined analysis of plant connectivity and alarm logs to reduce the number of alerts in an automation system. *Journal of process control*, 23(6), 839–851.

Stanton, N.A. and Baber, C. (2008). Modelling of human alarm handling response times: a case study of the ladbroke grove rail accident in the uk. *Ergonomics*, 51(4), 423–440.

Suriadi, S., Andrews, R., ter Hofstede, A.H., and Wynn, M.T. (2017). Event log imperfection patterns for process mining: Towards a systematic approach to cleaning event logs. *Information Systems*, 64, 132–150.

Van der Aalst, W.M.P. (2011a). Data mining. In *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer Berlin Heidelberg.

Van der Aalst, W.M.P. (2011b). Getting the data. In *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer Berlin Heidelberg.

Van der Aalst, W.M.P., Weijters, T., and Maruster, L. (2004). Workflow mining: discovering process models from event logs. *IEEE Transactions on Knowledge and Data Engineering*, 16(9), 1128–1142.

Weijters, A.J.M.M. and Ribeiro, J.T.S. (2011). Flexible heuristics miner (fhm). In *IEEE Symposium on Computational Intelligence and Data Mining*, 310–317.

Weijters, A.J.M.M., van Der Aalst, W.M.P., and De Medeiros, A.A. (2006). Process mining with the heuristics miner-algorithm. *Technische Universiteit Eindhoven, Tech. Rep. WP*, 166, 1–34.

Wen, L., Wang, J., and Sun, J. (2006). Detecting implicit dependencies between tasks from event logs. In *Asia-Pacific Web Conference*, 591–603. Springer.