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

Online Alarm Flood Classification Using Alarm Coactivations ⋆

Matthieu Lucke [*,**] **Moncef Chioua** [*] **Chriss Grimholt** [***]
Martin Hollender [*] **Nina F. Thornhill** [**]

[*] *ABB Corporate Research Germany, Wallstadter Strasse 59, 68526 Ladenburg, Germany (e-mail: matthieu.lucke@de.abb.com)*
[**] *Centre for Process Systems Engineering, Department of Chemical Engineering, Imperial College London, London SW7 2AZ, UK*
[***] *ABB Oil, Gas and Petrochemicals, Ole Deviks Vei 10, 0666 Oslo, Norway*

Abstract: Alarms indicate abnormal operation of the process plants and alarm floods constitute specific abnormal episodes that cannot be handled safely by the operators. In that regard, online alarm flood classification based on a bank of past historical episodes provides support on how to handle ongoing alarm sequences. This paper introduces a new approach based on alarm coactivations that is appropriate for the analysis of ongoing sequences. The method shows improvements when compared to an established sequence alignment approach for abnormal episode analysis of a gas oil separation plant.

Keywords: Alarm systems, Fault detection and diagnosis, Alarm flood analysis.

1. INTRODUCTION

Alarm flood episodes can be a threat to the safe and efficient operation of a plant. An alarm is an audible and/or visible means of indicating to the operator an equipment malfunction, process deviation, or abnormal condition requiring a timely response (IEC, 2014). An alarm flood is a condition during which the alarm rate is greater than the operator can effectively manage, i.e. greater then 10 alarms per 10 minutes (IEC, 2014). The research community has suggested different approaches to tackle this problem, from alarm rationalization (Beebe et al., 2013) and root cause analysis (Rodrigo et al., 2016) to pattern matching of alarm floods (Lai et al., 2017). While regular rationalization of the alarm system is the best approach to avoid alarm floods, pattern matching methods provide operators with support when the alarm floods are occurring, especially when it can be done online.

Several pattern matching methods have been proposed in the last few years to analyse alarm floods. Most of them belong to the category of Alarm Flood Sequence Alignment (AFSA) methods. Ahmed et al. (2013) suggested a pattern matching based on dynamic time warping. Cheng et al. (2013) introduced a modified version of the Smith-Waterman algorithm (MSW), a dynamic programming algorithm for sequence alignment, to measure the similarity between two alarm sequences. Lai and Chen (2017) extended the MSW algorithm to extract a pattern sequence from multiple sequences. Guo et al. (2017) suggested another implementation of the MSW to reduce the computational burden. Other sequence alignment algorithms such as the BLAST algorithm have been applied as alternative to the MSW algorithm (Hu et al., 2016).

The methods listed above have been restricted to offline pattern matching applications because the computational requirements limit their application online. Lai et al. (2017) recently proposed the first online AFSA method. However the approach of Lai et al. (2017) hides a certain complexity since it relies on computational blocks introduced in other works (Cheng et al., 2013; Lai and Chen, 2017), which require additional effort when it comes to industrial implementation. This paper presents an alternative method for alarm flood classification based on alarm coactivations, whose simplicity makes it a good candidate for online industrial applications.

Similarity analysis of alarms has been studied in several works as a tool to reduce the number of redundant and consequential alarms. Various representations of the alarm binary signals and various similarity measures have been used. Kondaveeti et al. (2012) introduced the alarm correlation maps computed from the alarm binary signals as a visualization tool to assess redundant alarms. Alarm signals are set to one at the time of activation, zero otherwise (except for a padding of five seconds before and after the activation), and the correlation is based on the Jaccard similarity index (Lesot et al., 2009). Yang et al. (2012) computed the correlation maps using Pearson's correlation function based on pseudo (continuous) signals generated from the alarm binary signals. The binary signals are defined based on activations and deactivations of the alarms, and transformed to pseudo signals using a Gaussian kernel function. Yang et al. (2013) assessed multiple similarity coefficients for binary sequences and suggested a new method

⋆ Financial support is gratefully acknowledged from the Marie Skodowska Curie Horizon 2020 EID-ITN project PROcess NeTwork Optimization for efficient and sustainable operation of Europe's process industries taking machinery condition and process performance into account PRONTO, Grant agreement No 675215.

to detect correlated alarms based on Sorgenfrei coefficient and the distribution of the correlation delay.

The present work leverages a simple representation of alarm coactivations (close to the one proposed by Kondaveeti et al. (2012)) to classify alarm flood episodes online, according to past episodes that occurred recurrently on the process plant. While AFSA methods have been successful in finding patterns in alarm flood sequences, this paper argues that the information contained in the coactivation of binary alarm signals is sufficient to set up an algorithm that can easily be implemented online, and whose accuracy is comparable to AFSA methods. Section 2 details the procedure from extraction of abnormal episodes to online classification of incoming alarm sequences. Section 3 presents the results of the approach on an industrial case and compares it with an AFSA approach, namely the MSW algorithm (Cheng et al., 2013). Section 4 analyses the outcomes of the approach, and Section 5 provides concluding remarks.

2. ABNORMAL EPISODE CLASSIFICATION USING ALARM COACTIVATION MAPS

The five main steps of the proposed alarm flood analysis procedure are described in this section. The four first steps are done offline, while the last one corresponds to the online analysis. The five steps are:

- Alarm chattering removal and extraction of the abnormal episodes of alarm flood.
- Computation of the alarm coactivation maps on the alarms generated during the abnormal episodes.
- Selection of the abnormal episode classes to monitor.
- Training of the classification model based on selected recurrent types of abnormal episodes.
- Online analysis of the incoming alarm sequences using the classification model.

2.1 Extraction of the abnormal episodes

Chattering alarms are first removed using a minimum admissible time interval of 10 minutes between two consecutive occurrences of alarms of the same type (Ahmed et al., 2013). Alarm flood episodes are selected based on the number of alarms occurring during a 10-minute sliding window computed every second. Consecutive alarm flood episodes are merged into a single interval to form an abnormal episode. The 10 minutes preceding (resp. following) the merged intervals are also included in the abnormal episode to cover the beginning (resp. end) of the alarm sequence.

2.2 Computation of the coactivation maps

The next step is to transform the alarm sequence corresponding to each abnormal episode into an Alarm Coactivation Map (ACM) to be used as feature for the classification. The work reported in this article suggests a similarity measure that can easily be computed online while capturing the dynamics of the alarm sequences. For each alarm tag defined on the process plant, the corresponding alarm binary signal is set to one when the alarm is activated (i.e. until the alarm returns to normal, noticeable difference with the implementation proposed by Kondaveeti et al. (2012)), to zero otherwise. The Jaccard similarity index (Lesot et al., 2009) is used to compute the similarity between each pair of alarm binary signals $X = (x_1, x_2...x_N)$ and $Y = (y_1, y_2...y_N)$ where N is the number of samples in the time interval of study.

$$S_{jac}(X,Y) = \max_{l \in L} \frac{a(l)}{a(l) + b(l) + c(l)} \tag{1}$$

where:

- $l \in L$ is the lag to be considered between the two signals X and Y. The set of lags L is chosen according to the dynamics of the process.
- $a(l)$ is the number of samples i where $x_i = 1$ and $y_{i+l} = 1$ over the time interval of study,
- $b(l)$ is the number of samples i where $x_i = 1$ and $y_{i+l} = 0$ over the time interval of study,
- $c(l)$ is the number of samples i where $x_i = 0$ and $y_{i+l} = 1$ over the time interval of study.

For each abnormal episode identified, an ACM is formed as the matrix $S_{jac}(X_i, X_j)$, $i, j \in [1, n]$, where the X_i represents the alarm binary signal associated with the alarm tag i out of the n unique alarm tags configured on the process plant. By convention, the similarity of an alarm signal with itself $S_{jac}(X_i, X_i)$ is set to one if the alarm is activated at some point during the time interval of the analysis, to zero otherwise. Thus, the diagonal elements of the matrix represent the aggregated vector of the alarms that appeared during the abnormal episode.

2.3 Selection of the abnormal episode classes

Alarm flood identification based on a classification model requires several occurrences of similar abnormal episodes for training of the model. In that regard, the approach targets recurrent abnormal episodes (see examples in Section 4.1). Extracting the corresponding episodes is a tedious task to be done manually by the operators, possibly with help of a clustering algorithm. The classes of abnormal episodes selected by the operators to be monitored are referred to as selected classes. The other classes are referred to as leftover (possibly unknown) classes.

2.4 Classification of the coactivation maps

The ACM classification is tested using two different approaches: a Support Vector Machine (SVM) classification (Cortes and Vapnik, 1995) and a k Nearest Neighbours (kNN) classification (Fix and Hodges, 1951). The ACMs being symmetric matrices, the classifier is applied to the lower triangular elements of the matrices, which are stacked as an input vector. The dimension of the input vector is $n(n+1)/2$, where n is the total number of alarms designed on the plant. In addition, a detection threshold is introduced to determine if a given ACM should be classified as a one of the selected classes or if it belongs to the leftover classes.

Classification with SVM A SVM classifier is a linear discriminative classifier that uses hyperplanes to separate classes. SVM is selected in this work for its good generalization properties even when dealing with small data sets, as discussed by Vapnik (1995), which is the case when dealing with abnormal episodes on a process plant. For example, the case study in Section 3 features at most a few dozens of occurrences of each class of abnormal episodes over one year of operation. The resistance to overfitting of SVM due to the use of regularization (Vapnik, 1995) is also an important criterion since the dimension of the ACM is much greater than the number of observed episodes. The SVM classification is done based on a one-vs-one approach. A k-fold cross validation is used for the choice of the kernel function and hyper parameters. The metric used for the detection is the probability of the classification result p which is expressed as the posterior class probabilities defined by Platt's probabilistic output to the SVM as described by Lin et al. (2007). The detection threshold is defined as probability threshold α.

Classification with k Nearest Neighbours A kNN classifier is a non-parametric classifier that uses the k closest training samples as local approximation to assign the class of the new sample. kNN is selected in this work to provide a consistent comparison with the AFSA approach that uses kNN classification (Lai et al., 2017). The kNN classification is done using a Euclidian distance between the input vectors. Since data is sparse due to the small number of episodes, k is set to one, which is consistent with the approach presented by Lai et al. (2017). The distance to the nearest neighbour d is used as a metric for the detection, and the detection threshold is defined as a distance threshold β.

2.5 Online analysis of incoming alarm sequences

An online implementation of the algorithm for analysis of incoming alarm sequences is presented in Figure 1. A sliding window is used to detect the periods with high alarm rates on which the analysis should focus. The alarm rate τ is defined as the number of alarms in the last 10 minute interval. Two thresholds are set: τ_2 is the alarm flood threshold (10 alarms per 10 minutes according to IEC (2014)), τ_1 is an intermediate threshold chosen such that $0 < \tau_1 \leq \tau_2$. τ_1 is introduced for the analysis to be triggered prior to the end of the abnormal episode to provide operators with early diagnostics, and can be considered as an hyper-parameter to be tuned.

The alarm rate τ is computed continuously (e.g. every second) over a sliding window of 10 minutes (600 seconds). Alternatively, the length of the sliding window can be tuned according to the dynamics of the process. When $\tau \geq \tau_1$, indicating that an alarm flood episode may be developing, an ACM is computed every time a new alarm occurs on a window covering the last 1200 seconds. When an alarm flood interval is detected (i.e. $\tau \geq \tau_2$) with a starting time s, the starting time of the abnormal episode is set to 600 seconds before s. An ACM is computed every time a new alarm occurs (in a window beginning at time $s-$ 600 seconds) until 600 seconds after the alarm rate drops below the alarm flood threshold, corresponding to the end time e. Each computed ACM is fed as an input vector to the classifier according to the two methods described in Section 2.4. The classification result is presented to the operator if the probability p returned by the SVM is greater than the detection threshold α (or alternatively

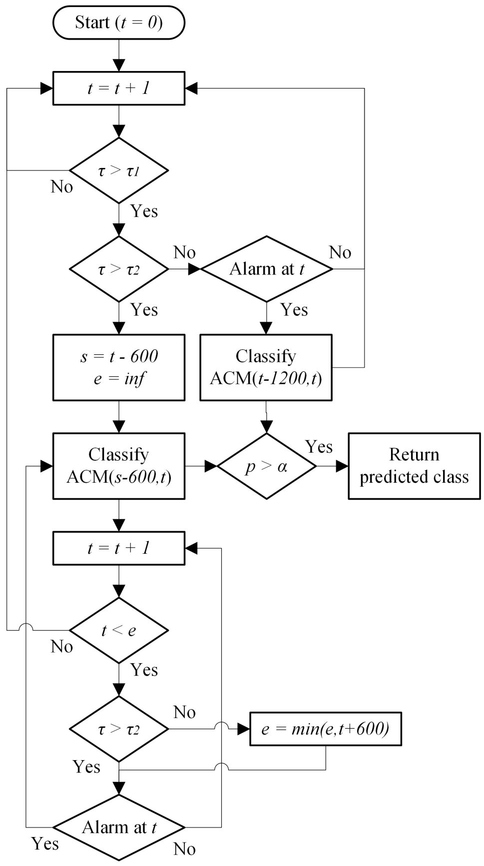

Fig. 1. Flowchart of the proposed online implementation. The times are expressed in seconds.

if the distance d returned by the 1NN is smaller than the detection threshold β).

3. INDUSTRIAL CASE STUDY

3.1 Description of the industrial case study

The industrial case study is an offshore gas oil separation plant, designed to separate crude oil, gas and condensates next to the well before export. An overview of the process is given in Figure 2. The analysis covers a period of 382 days.

The analysis focusses on a subset of alarms, namely the process alarms and trip events, which amounts to 1473 unique alarm tags on the plant. For this reason, the alarm flood threshold is lowered to 8 alarms per 10 min. Among

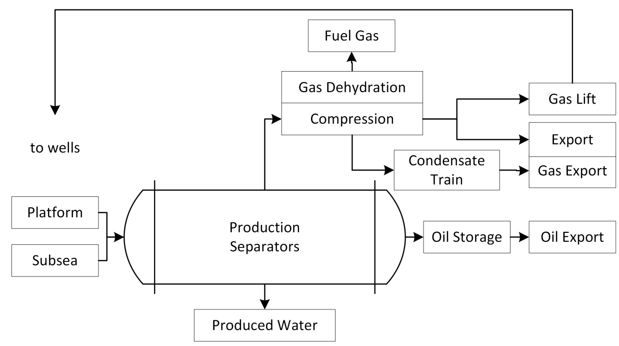

Fig. 2. Process overview of the separation plant

the identified 926 abnormal episodes, 81 correspond to five types of recurrent episodes which occurred more than five times during the period under analysis (description in Table 1).

Table 1. Description of the selected types of abnormal episodes. The length is expressed in number of alarms.

Class	Number of alarm floods	Average sequence length	Longest sequence length	Shortest sequence length
A	16	16.6	31	8
B	16	13.4	26	8
C	9	24.4	40	18
D	11	25.3	31	19
E	29	24.6	48	15

3.2 Performance and comparison with AFSA approach

The performance of the ACM approach is studied with a 1NN classifier and with a SVM classifier, and compared to the performance of a standard AFSA approach, i.e. the MSW algorithm (Cheng et al., 2013) with 1NN classifier. Optimization of the performance of the MSW approach using advanced similarity-based classification methods (Chen et al., 2009) is out of scope of this paper. The results given in this section correspond to the classification of the ACMs computed over the whole period of the abnormal episode, i.e. offline analysis. The performance is assessed based on three criteria: the False Detection Rate (FDR), the Missed Detection Rate (MDR) and the accuracy of the classification. While FDR and MDR indicate the ability of the method to differentiate selected classes from the numerous leftover classes that can occur on a process plant, the accuracy indicates the ability to identify the correct class among the five selected in Table 1.

False detection rate The FDR indicates the proportion of abnormal episodes from leftover classes that are identified as selected classes. For the FDR computation, the model is trained on the 81 selected episodes and tested on the 845 leftover episodes.

Missed detection rate The MDR indicates the proportion of abnormal episodes from selected classes that are not identified as such. For the MDR computation, the model is trained on a set containing a random half of each of the five classes (in total 42 episodes), and tested on a set with the other halves (39 episodes).

Accuracy of the classification The accuracy is computed as the average accuracy of the classification on 50 independent experiments using random splits between training and test data on the 81 labelled ACMs using half for training and half for testing. Using an average accuracy on 50 random splits mitigates the variability in the accuracy value due to the small number of samples.

Table 2 presents the FDR and MDR values for three configurations of the detection threshold which highlight the trade-off between FDR and MDR: one configuration corresponds to a priority given to a low MDR, one to a priority given to a low FDR, and one to a trade-off between MDR and FDR. Table 3 shows the results of the accuracy tests.

Both tables highlight that, in spite of the simplicity of the extracted features, the ACM approach can present a performance comparable (or slightly higher) than the MSW approach considered as benchmark in the industry. While 1NN with Euclidian distance does not seem to be an appropriate choice of classifier for the ACM approach, SVM offers a good separation space for the feature vectors. The performance of the ACM approach with SVM classifier rivals the MSW approach both in term of accuracy and in term of FDR/MDR (except for low FDR configuration). Analysis of the results are provided in Section 4.2. The following section considers the validity of the online implementation of the ACM+SVM approach.

Table 3. Average accuracy of the classification

Method	Accuracy (%)
MSW + 1NN	91.2
ACM + 1NN	90.7
ACM + SVM	92.2

3.3 Online implementation

Two criteria with regard to the online implementation of the approach are studied in this section: the computation time of the algorithm, and the ability to classify ongoing sequences.

Computation time

The response time is taken as the time between an alarm occurrence and the time when the algorithm returns a classification. The average response time of the algorithm (with SVM classification) using a C# implementation on a 64 bit Windows PC with Intel(R) Core(TM) i7-6820 2.70 GHz CPU and 16.0 GB memory is 1.5 ms, which is fully satisfactory compared to the temporal resolution of the alarms on the plant (one second). The detailed response time analysis of the MSW+1NN approach provided by Lai et al. (2017) shows that an optimized version of the algorithm for online analysis would also be applicable in this case (response time below one second).

Classification of ongoing episodes

The ability of the proposed approach to identify ongoing alarm sequences as episodes from the selected classes is evaluated through the probability p of the classification outcome that can be compared with a detection threshold α. The evolution of the probability of the classification

Table 2. Results of the FDR and MDR tests

	MSW + 1NN		ACM + 1NN		ACM + SVM	
	FDR (%)	MDR (%)	FDR (%)	MDR (%)	FDR (%)	MDR (%)
Low MDR	22.1	0.0	66.7	0.0	17.2	0.0
Trade-off	7.5	7.7	23.3	20.5	8.8	7.7
Low FDR	0.5	46.1	0.7	53.8	0.9	82.0

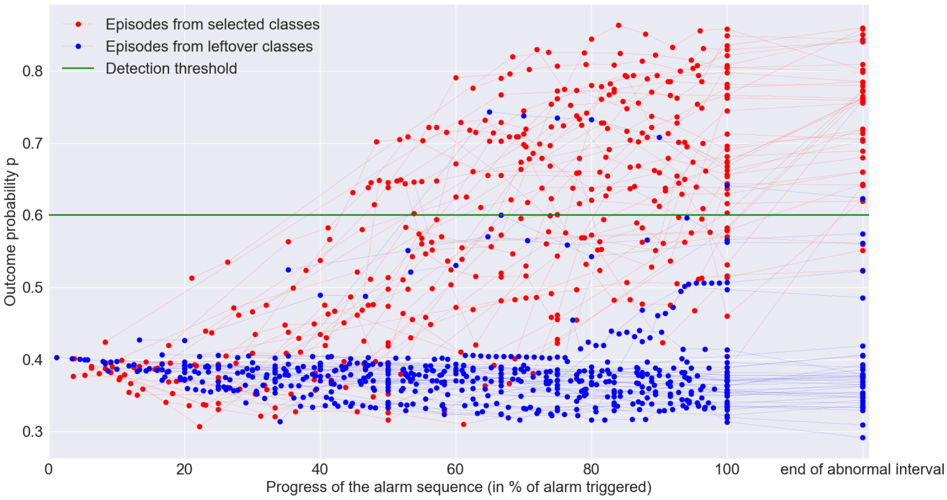

Fig. 3. Probability of the classification outcome for ongoing episodes. Each point corresponds to a classification.

outcome as the alarm sequences progress is presented in Figure 3. The model is trained with the training set used for the MDR test in Section 3.2, and tested on the same set of 39 selected episodes (in red in Figure 3). In addition, the model is tested on 39 random leftover episodes (in blue in Figure 3). The evolution of the ongoing alarm sequence is expressed in term of ratio of triggered alarms to the total number of alarms in a sequence. Each point corresponds to a classification, triggered every time a new alarm (or simultaneous alarms) occurs following the implementation in Figure 1. The final classification is given at the end of the abnormal episode interval as identified in Section 2.5, shortly after all the alarms of the sequence appeared. The classification results at the end of the abnormal episode interval correspond to the offline results (Table 2). The detection threshold corresponding to the trade-off configuration of Table 2 is plotted as a green horizontal line for reference.

4. DISCUSSION

4.1 Analysis of recurrent abnormal episodes

This case study illustrates how diagnosing recurrent episodes can present major value for the operation of the plant. Two of the five selected classes of abnormal episodes (Table 1) are analysed in this section to support this point.

Class D and class E correspond to abnormal episodes in the Produced Water Reinjection (PWRI) system (Figure 4), where used water is going through degasification before being reintroduced in the plant via two injection points. Both abnormal episodes in class D and class E manifest as pump trips in the system, leading to a low water flow going to the injection point 2 (e.g. through FCV 133). Two alarm sequences from class D are displayed in Table 4. Sequences in class D start with a low flow alarm in the pump P11 (A_FICA_130_L) shortly followed by a low flow alarm in pump P21 (A_FICA_116_L), leading to trips of both pumps. The level in the degassing drum increases quickly until high alarms for the water level (C_LICA_128A_H) and the oil level (C_LT_118_H) are successively triggered. Sequences in class E start with a low flow alarm in pump P21 followed by a trip of the pump, leading to a high level in the degassing drum. Further analysis by the operators showed that the alarm flood sequences in class D were due to a change of type of fuel used to run the pumps, whereas sequences in class E are due to a low initial value of the suction pressure (and suction flow).

Those two cases illustrate how a recurrent abnormal episode with an important impact on the operation of the plant (shutdown of the PWRI system) can be immediately identified by the proposed approach.

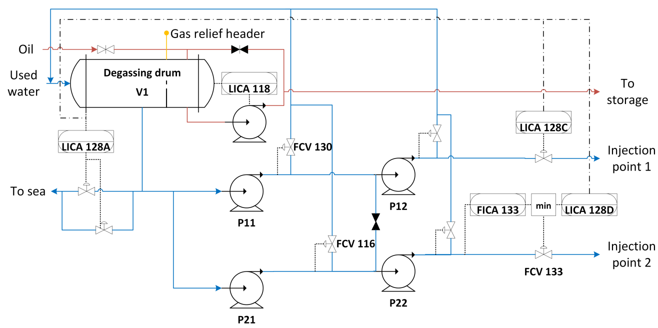

Fig. 4. Process diagram of the PWRI section. Blue indicates the water system, brown the oil system, and yellow the gas system.

4.2 Discussion of the results

Section 3.2 demonstrates that alarm coactivations can be used as features to classify abnormal events with performance comparable to the AFSA approach. While a 1NN classifier based on the Euclidian distance between the coactivation vectors struggles to separate the different classes especially when taking into account the numerous leftover classes, the SVM classifier with Gaussian kernel provides a separation space that is more appropriate to the high dimensional feature space. The performance of the ACM

Table 4. Two alarm flood sequences of class D.

Alarm type	Timestamp	Alarm type	Timestamp
H_TT_135_L	06:14:30	G_TT_134_L	22:45:40
A_FICA_130_L	06:26:47	A_PT_2051_L	23:34:17
A_FZAL_115_LL	06:26:47	A_FZT_127_LL	23:34:30
A_FICA_116_L	06:26:48	A_FICA_130_L	23:34:30
A_FZT_115_LL	06:26:48	A_FZT_114_L	23:34:31
A_FZT_114_L	06:26:48	A_FZAL_115_LL	23:34:31
A_FZT_127_LL	06:26:48	A_FICA_116_L	23:34:31
A_PT_121_L	06:26:49	A_FZT_115_LL	23:34:32
B_P81_TRIP	06:26:49	A_FZT_114_LL	23:34:32
A_P21_TRIP	06:26:59	A_PZT_2260_LL	23:34:32
A_P11_TRIP	06:27:01	A_PT_121_L	23:34:32
C_LICA_128A_H	06:27:14	B_P81_TRIP	23:34:33
C_LT_118_H	06:27:23	A_P21_TRIP	23:34:43
E_PDIC_129_H	06:27:31	A_P11_TRIP	23:34:44
D_PZT_119A_L	06:28:58	A_PZT_117_L	23:34:46
A_PZT_124_H	06:33:42	C_LICA_128A_H	23:34:57
A_PZT_124_HH	06:34:49	C_LT_118_H	23:35:05
D_PT_131_L	06:35:42	A_PZT_2054_L	23:36:35
D_PT_132_L	06:36:12	D_PZT_119B_L	23:37:02
D_PT_131_LL	06:37:14	A_PZT_124_H	23:37:34
D_PT_132_LL	06:38:48	A_PZT_2054_LL	23:37:47
A_FICA_116_L	06:44:50	D_PT_131_L	23:42:44
		D_PT_131_LL	23:43:32
		D_PT_132_L	23:44:26
		D_PT_132_LL	23:47:16
		D_PZT_119A_L	23:50:09
		D_PZT_119B_L	23:50:09

approach with SVM classification rivals the one of the AFSA approach both in term of accuracy and detection rates for the low MDR and trade-off configurations, which correspond to usual industrial applications. The high value of the MDR for ACM+SVM in the low FDR configuration shows the limit of the detection threshold based on the posterior probability of the classification outcome. This configuration corresponds to a very high probability threshold α (here 0.8), which means that only the episodes with a very high classification probability outcome will be classified as one of the selected classes. The low FDR configuration is not adapted to industrial systems where a certain variability in the content and dynamics of the sequences can be observed. With the same perspective, the accuracy of all three methods does not reach 100% since sequences from different classes have a number of alarms in common.

Section 3.3 validates the applicability of the ACM+SVM approach online following the implementation suggested in Section 2.5. Figure 3 illustrates how the probability of the classification increases as additional alarms appears for the episodes corresponding to the selected classes, and how it reaches the detection threshold before the end of the abnormal episode interval. On the contrary, the probability stays low for leftover classes.

5. CONCLUSION

The proposed classification approach based on alarm coactivations allows efficient analysis of abnormal episodes. The method is particularly adapted for online industrial applications where its performance compares well with traditional AFSA approaches while keeping a low computational burden. In addition, the simplicity of the approach increases greatly its applicability to industrial cases.

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.

Beebe, D., Ferrer, S., and Logerot, D. (2013). The connection of peak alarm rates to plant incidents and what you can do to minimize. *Process Safety Progress*, 32, 72–77.

Chen, Y., Garcia, E.K., Gupta, M.R., Rahimi, A., and Cazzanti, L. (2009). Similarity-based classification: Concepts and algorithms. *Journal of Machine Learning Research*, 10, 747–776.

Cheng, Y., Izadi, I., and Chen, T. (2013). Pattern matching of alarm flood sequences by a modified Smith-Waterman algorithm. *Chemical Engineering Research and Design*, 91, 1085–1094.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273–297.

Fix, E. and Hodges, J.L. (1951). Discriminatory analysis - nonparametric discrimination: consistency properties. *USAF School of Aviation Medicine, Randolph Field, Texas.*

Guo, C., Hu, W., Lai, S., Yang, F., and Chen, T. (2017). An accelerated alignment method for analyzing time sequences of industrial alarm floods. *Journal of Process Control*, 57, 102–115.

Hu, W., Wang, J., and Chen, T. (2016). A local alignment approach to similarity analysis of industrial alarm flood sequences. *Control Engineering Practice*, 55, 13–25.

IEC (International Electrotechnical Commission) (2014). *Management of Alarms Systems for the Process Industries. IEC 62682.* IEC.

Kondaveeti, S.R., Izadi, I., Shah, S.L., Black, T., and Chen, T. (2012). Graphical tools for routine assessment of industrial alarm systems. *Computers and Chemical Engineering*, 46, 39–47.

Lai, S. and Chen, T. (2017). A method for pattern mining in multiple alarm flood sequences. *Chemical Engineering Research and Design*, 117, 831–839.

Lai, S., Yang, F., and Chen, T. (2017). Online pattern matching and prediction of incoming alarm floods. *Journal of Process Control*, 56, 69–78.

Lesot, M.J., Rifqi, M., and Benhadda, H. (2009). Similarity measures for binary and numerical data: a survey. *International Journal of Knowledge Engineering and Soft Data Paradigms*, 1, 63–84.

Lin, H.T., Lin, C.J., and Weng, R.C. (2007). A note on Platt's probabilistic outputs for support vector machines. *Machine Learning*, 68, 267–276.

Rodrigo, V., Chioua, M., Hagglund, T., and Hollender, M. (2016). Causal analysis for alarm flood reduction. In *Proceedings of 11th DYCOPS, Trondheim, Norway, June 6-8, 2016*, 723–728.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory.* Springer-Verlag, New York.

Yang, F., Shah, S.L., Xiao, D., and Chen, T. (2012). Improved correlation analysis and visualization of industrial alarm data. *ISA Transactions*, 51, 499–506.

Yang, Z., Wang, J., and Chen, T. (2013). Detection of correlated alarms based on similarity coefficients of binary data. *IEEE Transactions on Automation Science and Engineering*, 10, 1014–1025.