Logical processing of sequential alarms for safer plant operation

Ryuto Tanakaa and Masaru Nodaa\*

*aDepartment of Chemical Engineering, Fukuoka University, Fukuoka 814-0180, Japan*

mnoda@fukuoka-u.ac.jp

Abstract

Logical alarm processing is a technique for processing signals from alarm sensors to generate more meaningful alarms for plant operators. This technique involves grouping and suppressing redundant alarms, and eclipsing of alarms on the same variable etc. While it can enhance the operational value of alarms, inappropriate deployment of logical alarm processing can result in safety hazards during plant operations. In this study, we propose a novel approach for logical alarm processing that focuses on suppressing sequential alarms. Sequential alarms refer to sets of alarms that occur in succession within a short period of time after the first waring alarm indicating an abnormality. These types of alarms reduce the ability of plant operators to cope with operation abnormalities as critical alarms are often lost in numerous other correlated ones. The proposed method comprises three parts: identifying sequential alarms in plant operation data through dot matrix analysis, detecting the occurrence of sequential alarms in real-time by matching patterns of alarms during plant operation, and suppressing unnecessary alarms in the detected sequential alarms. Application of the method to simulated operational data for a distillation column demonstrated that its capability to decrease the number of unnecessary alarms displayed to a plant operator.

**Keywords**: Logical Alarm Processing; Sequential Alarms; Dot Matrix Analysis, Plant Operation Data

* 1. Introduction

Advancements in distributed control systems (DCS) in the chemical industry have made it possible to inexpensively and easily install numerous alarms in DCS. While most alarms help operators to detect and identify abnormalities, some do not (Hollifield *et al.*, 2006). A poor alarm system might cause sequential alarms, which are a series of alarms that occur in succession within a short period of time after the first waring alarm indicating a plant abnormality. These sequential alarms reduce the operators’ ability to cope with plant abnormalities because the critical alarms get buried under many unnecessary ones. To ensure safe plant operations, it is very important to suppress sequential alarms during plant operation.

Logical alarm processing is a technique for processing signals from alarm sensors to generate more meaningful alarms for plant operators, such as grouping alarms, suppression of redundant alarms, eclipsing of several alarms on the same variable, suppression of alarms according to plant operation mode (International Electrotechnical Commission, 2022). For instance, the suppressing relevant alarms during startup operation is appropriate as it may be difficult to avoid transiently exceeding the high limits that apply in steady operation. The utilization of logical alarm processing is highly advantageous, but it should be stressed that: the operator should be kept informed when logical processing is removing alarms from the display, any logical processing should be done in a manner which minimizes the possibility of error (Engineering Equipment & Material Users’ Association, 2013).

In this study, we propose a new logical alarm processing method to suppress sequential alarms. In a sequential alarm, the alarms following the first few alarms do not provide useful information to the operator and thus should be suppressed. The proposed method consists of three parts, a database construction of sequential alarms using plant operation data, detection of the occurrence of sequential alarms during the plant operation, and the suppression of sequential alarms. The effectiveness of the proposed method is demonstrated through a case study.

* 1. Logical Alarm Processing of Sequential Alarms
		1. Plant Operation Data

The plant operation data recorded in DCS generally includes the timing and tag information of the occurred alarms, as presented in Table 1. The notation A*k* in Table 1 represents the alarm associated with the *k*th tag.

Table 1 Example plant operation data

|  |  |
| --- | --- |
| Date – Time | Tag  |
| 2024/01/01 – 10:10:15 | A3 |
| 2024/01/01 – 10:12:36 | A1 |
| 2024/01/01 – 10:13:42 | A2 |
| 2024/01/01 – 10:14:58 | A3 |
| 2024/01/01 – 10:45:08 | A4 |
| 2024/01/01 – 10:51:21 | A1 |
| 2024/01/01 – 11:13:33 | A2 |
| 2024/01/01 – 11:31:21 | A1 |
| 2024/01/01 – 11:32:43 | A2 |
| 2024/01/01 – 11:33:18 | A3 |
| 2024/01/01 – 11:34:59 | A4 |
| 2024/01/01 – 11:45:30 | A2 |

Plant operation data is characterized by these tags and the order of alarm occurrence. Conversion of the plant-operation data in Table 1 by placing them in order by occurrence time gives alarm sequence *S* in Eq.(1):

*S* = A3, A1, A2, A3, A4, A1, A2, A1, A2 , A3 , A4 , A2 (1)

* + 1. Identification of Sequential Alarms by Dot Matrix Analysis

Repeated alarm patterns in plant operation data can be classified as sequential alarms. Thus, the problem of identifying sequential alarms in plant operation data is formulated as the problem of searching for repeated subsequences *Si* of alarms in alarm sequence *S*. Previously, we proposed an identification method for sequential alarms buried in noisy plant operation data using dot matrix analysis (Wang *et al.*, 2017). The dot matrix analysis is originally a sequence alignment method for identifying similar regions in DNA or RNA, which may be a consequence of functional, structural, or evolutionary relationships between sequences (Mount, 2004)

In the dot matrix analysis of the alarm sequence *S* in Eq.(1), *S* is listed up the left side and across the bottom of the graph, as shown in Figure 1. The sequence comparison starts with the first alarm on the horizontal axis. The comparison moves across the graph in the first row, and a dot is placed in every column where the alarm is the same. Then, the second alarm in the sequence is compared to the entire sequence, and a dot is placed in the second row wherever a match occurs. This procedure is repeated until the graph is entirely filled with dots.

The diagonal lines in Figure 1 display the locations of the repeated regions in *S*. The longest diagonal line, running from the lower left to the upper right, does not represent a sequential alarm as it reflects the comparison of a sequence with itself. In contrast, two diagonal lines with four dots indicate the occurrence of repeated sequential alarms “A1, A2, A3, A4” twice during the plant operation. Dots that do not align on a diagonal row or align on a diagonal row with less than three dots represent random matches unrelated to the sequential alarms.



Figure 1 Example of single-sequence dot matrix analysis

The identified sequential alarms are used in constructing a database for suppressing sequential alarms during plant operation, as shown in Table 2. The time period of sequential alarms △*t*1 in Table 2 is period between the first and the last alarms in the sequential alarm *S*1.

Table 2 Example of sequential alarms identified using dot matrix analysis

|  |  |  |
| --- | --- | --- |
| Sequential alarms | Alarm sequence | Time period of sequential alarms |
| *S*1 | A1, A2, A3, A4 | △*t*1 |

* + 1. Detection and Suppression of Sequential Alarms

Sequential alarm *Si* identified through the dot matrix analysis is divided into a pattern matching part and a suppression part. The patter matching part with the first few alarms in *Si* is used to detect the occurrence of sequential alarms during the plant operation, while the suppression part is the remaining alarms in *Si*. Once the pattern matching part of *Si*is found during plant operation, the method detects the occurrence of *Si* and suppresses the alarms in the suppression part of *Si*. A size of the matching part *N*d should be the minimum number enough to distinguish all sequential alarms identified by the dot matrix analysis.

Table 3 shows an example of database for detecting the occurrence of *S*1 in plant operation, where *N*d is selected as two. When “A1,A2” is detected during plant operation, “A3” and “A4” are suppressed for the suppression time △*t*1.

Table 3 Example of database for detecting the occurrence of sequential alarms

|  |  |  |  |
| --- | --- | --- | --- |
| Sequential alarms | Matching part | Suppression part | Suppression time of sequential alarms |
| *S*1 | A1,A2 | A3, A4 | △*t*1 |

* 1. Case Study

The proposed method was applied to simulation data obtained from a distillation column. There are four high alarms in the DCS: column top temperature, column bottom temperature, reflux flow rate, and steam flow rate, which are denoted by A1-A4 as shown in Table 4. The alarms A1-A4 are activated when corresponding process variables exceed their high alarm thresholds, respectively.

Table 4 Installed alarms in distillation column

|  |  |
| --- | --- |
| Alarms | Description |
| A1 | High alarm of column top temperature |
| A2 | High alarm of column bottom temperature |
| A3 | High alarm of reflux flow rate |
| A4 | High alarm of steam flow rate |

During the process simulation of 480 minutes, we intentionally induced four types of malfunctions. The plant operation data recorded a total of 143 alarm occurrences, which are illustrated in Figure 2.

The dot matrix analysis was applied to the plant operation data in Figure 2. Eight sequential alarms *S*1–*S*8 in Table 5 were identified. One of the longest sequential alarms *S*1 occurred twice in 480 minutes, which includes ten alarms with a 6.6-minute time span between the first and the last alarms A2. Any alarms following the first alarm A2 in *S*1 do not provide useful information to the operator, therefore they should be suppressed through logical alarm processing. The total number of alarms in the eight sequential alarms was 116, which accounts for 81% of the number of alarms that occurred in 480 minutes. This indicates that it is possible to reduce the number of alarms by 81% at most through the logical alarm processing.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *S* = | $$A\_{3},$$ | $$A\_{4},$$ | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{4},$$ | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{1},$$ |
|  | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{4,}$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{3},$$ | $$A\_{1},$$ |
|  | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{1},$$ | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{3},$$ |
|  | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{1},$$ | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{3},$$ |
|  | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{4},$$ | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{1},$$ |
|  | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{1},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{1},$$ |
|  | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{2},$$ | $$A\_{3},$$ |
|  | $$A\_{1},$$ | $$A\_{2},$$ | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{4},$$ | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{4},$$ | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{4},$$ | $$A\_{1},$$ |
|  | $$A\_{2},$$ | $$A\_{1},$$ | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{4},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{1},$$ |
|  | $$A\_{1},$$ | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ |
|  | $$A\_{3},$$ | $$A\_{1},$$ | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{4},$$ | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{2},$$ | $$A\_{3},$$ |
|  | $$A\_{2},$$ | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{1},$$ | $$A\_{3}$$ |  |

Figure 2 Simulated plant operation data

Table 5 Identified sequential alarms by dot matrix analysis

|  |  |  |  |
| --- | --- | --- | --- |
| *S*i | Alarm sequence | Freq. | Time length of sequential alarms [min] |
| *S*1 | A2, A3, A4, A1, A2, A3, A4, A1, A3, A2 | 2 | 6.6 |
| *S*2 | A3, A4, A2, A3, A4, A2, A3, A4, A1, A2 | 2 | 10.2 |
| *S*3 | A1, A2, A3, A4, A3, A2, A4, A1, A3, A2 | 2 | 8.3 |
| *S*4 | A4, A1, A2, A3, A2, A1, A3, A1 | 2 | 7.4 |
| *S*5 | A4, A1, A3, A2, A4, A1, A3, A2 | 2 | 4.7 |
| *S*6 | A3, A4, A2, A3 | 2 | 3.8 |
| *S*7 | A2, A3, A1, A2 | 2 | 3.5 |
| *S*8 | A1, A3, A2, A4 | 2 | 2.7 |

The database for supressing sequential alarms was constructed based on a list of identified sequential alarms using the dot matrix analysis in Table 5, as detailed in Table 6. The size of pattern matching part *N*d was selected as three, which is the minimum required number of alarms to identify the sequential alarms except for *S*6. It is impossible to distinguish *S*2 and *S*6 when *N*d is three as the sequence of the first three alarms of *S*6 are same as that of *S*2. Nevertheless, *N*d cannot be more than four, because the number of alarms in *S*6-*S*8 is four. For instance, when the alarm sequence “A2, A3, A4” is detected during the plant operation, alarms A1, A2, A3, and A4 are suppressed for 6.6 min through the logical alarm processing.

Table 6 Database for suppression of sequential alarms

|  |  |  |  |
| --- | --- | --- | --- |
| *S*i | Matching part | Suppression part | Suppression time [min] |
| *S*1*S*2*S*3*S*4*S*5*S*7*S*8 | A2, A3, A4A3, A4, A2A1, A2, A3A4, A1, A2A4, A1, A3A2, A3, A1A1, A3, A2 | A1, A2, A3, A4A1, A2, A3, A4A1, A2, A3, A4A1, A2, A3A1, A2, A3, A4A2A4 | 6.610.28.37.44.73.52.7 |

The proposed logical alarm processing method was applied to the plant operation data in Figure 2. Figure 3 reveals the alarms displayed to the operator with the dotted box denoting suppressed alarms that were not notified by the proposed method. This suggests that 55% reduction in alarms is feasible without sacrificing the first three alarms, which provide useful information to the operator.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *S* = | $$A\_{3},$$ | $$A\_{4},$$ | $$A\_{2},$$ |  |  |  |  |  |  |  |  |  |
|  |  | $$A\_{3},$$ | $$A\_{4},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{3},$$ |  |  |  |  |
|  |  | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{2},$$ |  |  |  |  |  | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{3},$$ |
|  |  |  | $$A\_{1},$$ | $$A\_{2},$$ | $$A\_{3},$$ |  |  |  |  |  |  |  |
|  |  |  |  | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{3},$$ |  |  |  |
|  |  |  | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{1},$$ |  | $$A\_{4},$$ | $$A\_{1},$$ |
|  | $$A\_{3},$$ |  |  |  |  |  | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{2},$$ |  | $$A\_{2},$$ | $$A\_{3},$$ |
|  | $$A\_{1},$$ |  | $$A\_{1},$$ | $$A\_{3},$$ | $$A\_{4},$$ | $$A\_{2},$$ |  |  |  |  |  |  |
|  |  |  |  |  | $$A\_{4},$$ | $$A\_{3},$$ | $$A\_{2},$$ | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{3},$$ |  |  |
|  |  | $$A\_{2},$$ | $$A\_{3},$$ | $$A\_{4},$$ |  |  |  |  |  |  |  | $$A\_{4},$$ |
|  | $$A\_{3},$$ | $$A\_{1},$$ | $$A\_{2},$$ | $$A\_{3},$$ |  |  |  |  | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{2},$$ |  |
|  |  |  |  |  |  |  | $$A\_{4},$$ | $$A\_{1},$$ | $$A\_{3},$$ |  |  |  |

Figure 3 Suppression of Sequential Alarms by logical alarm processing

Conclusions

We developed a new logical alarm processing method to prevent sequential alarms. Our method includes three parts: constructing a database construction of sequential alarms by analysing plant operation data using the dot matrix analysis, detecting of the occurrence of sequential alarms during the plant operation by pattern matching, and suppressing sequential alarms. Application of the proposed method to simulated plant operation data of a distillation column showed that the proposed method can effectively suppress sequential alarms while ensuring that critical alarms are displayed to the operator.

Acknowledgement

This work was supported by JSPS KAKENHI Grant Numbers JP20K05213 and JP23K04491.

References

B. R. Hollifield and E. Habibi, 2006, The Alarm Management Handbook A Comprehensive Guide, PAS, Houston

International Electrotechnical Commission (IEC), 2022, IEC 62682 Management of Alarm Systems for the Process Industries 2nd edition, IEC, Geneva

Engineering Equipment & Material Users’ Association (EEMUA), 2013, Alarm Systems – a Guide to Design, Management and Procurement, EEMUA Publication No.191 3rd Edition, EEMUA, London

Z. Wang and M. Noda, 2017, Identification of Repeated Sequential Alarms in Noisy Plant Operation Data Using Dot Matrix Method with Sliding Window, *Journal of Chemical Engineering of Japan,* 50(6), 445-449

D. W. Mount, 2004, Bioinformatics Sequence and Genome Analysis Second Edition, Cold Spring Harbor Laboratory Press, New York