

# Study of Alarm Filtering Method Based on Multivariate Process Condition Data

Huang JianWen

Guangdong Gower Grid Company of China Southern Power  
Grid, DongFeng East Road No. 757  
Yuexiu District of Guangzhou Gity, Guangdong Province  
China

Peng ZeWu\*

Guangdong Gower Grid Company of China Southern Power  
Grid, DongFeng East Road No. 757  
Yuexiu District of Guangzhou Gity, Guangdong Province  
China  
[\\*952898435@qq.com](mailto:*952898435@qq.com)

**Abstract**—In this paper, through the analysis of core demands related multivariate process condition data, the obtained features and classification results are used to calculate the alarm statistical index. By evaluating alarm presence rate and effective alarm rate, alarm classification filtering is achieved, where the upper layer applications include the pre-/post-fault alarm statistical tool and the alarm classification filtering tool. The speeding fault alarm of concrete pumping truck is taken as an example to analyze the results of alarm filtering using the alarm filtering framework developed in this work.

**Keywords**—process monitoring, time series, pattern discovery, alarm filtering

## I. INTRODUCTION

In the field of engineering machinery, alarm events can monitor and feedback the status of equipment in real time, and thus plays an important role in ensuring the normal operation of equipment[1,2]. However, due to the numerous types of alarms, unclear inherent causes, frequent occurrence, and partially lack of warning effectiveness, etc., alarms often lead to unnecessary nuisance to the operator. The operator generally only pay attention to alarms of equipment failure causing huge impact, but would often be misled by the frequent alarms of low impact, and thus falsely report a fault[3].

## II. OPERATION DEMAND ANALYSIS

Alarm filtering is to retain the alarms of high value from the frequently occurred alarms, and discard the alarms of low value[4]. An ideal alarm filtering tool should be able to identify and classify the importance and impact of alarms from different parts of the equipment, thereby automatically assisting the operator to retain high-impact alarms and discard low-impact alarms. The use of such a tool can greatly improve alarming efficacy and rationality, by accurately predicting failures of each part of the device and greatly reducing the false alarm rate[5].

This paper presents an alarm filtering model. Through the use of alarm features and alarm classification from multivariate process condition data, in conjunction with alarm statistical rules, this model can filter the low-impact alarm classes. One of the main goals is to design and develop the statistical tool of pre-/post-fault alarms, which enables the users to check the

fault repair data of each part of the device, as well as the statistical indicator summary and detailed log for each statistical indicator of the alarms before and after repair. With the help of this model, users can discover the part of the device that is highly relevant to the alarm and perform further analysis accordingly. The second goal is to design and develop an alarm classification filtering tool, which enables the users to classify the alarm statistical indicators according to the alarm classification, and subsequently identify the impact of the alarm class by analysis of the change in the statistical indicators of the alarm before and after fault repair. In this way, the low-impact alarms will be identified and discarded. Eventually, the impact of each alarm class under various statistical indicators on each part of the equipment will be sorted, helping the user timely spot potential equipment fault based on alarms.

Using the results from alarm filtering, users can easily identify the alarms with high impact, so as to further identify the possible type(s) of equipment failure. When the device frequently alarm, the user can first classify the alarm using the alarm classification tool. For the frequently occurring alarm class, the statistical indicators can be used to analyze possible equipment failure[6]. If a particular failure appears frequently in the analysis result, the user can perform systematic troubleshooting at the location of failure, or check the operations related to the failure. With this method, we cannot only detect the potential equipment failure in time and ensure healthy condition of the equipment, but also improve accuracy of troubleshooting and reduce the number of fault false alarms[7]. The alarm filtering method proposed in this paper distinguishes the impact of different alarm class through the equipment's fault repair data, thereby discarding the low-impact ones[8].

## III. OPERATION DEMAND ANALYSIS

An alarm filtering framework is constructed to filter the frequently occurring alarms, in which working conditions are identified using alarm relevant working condition data and the degrees of impact of the alarms are evaluated using alarm maintenance data. The overall structure of the alarm filtering framework is shown in Fig. 1. It consists of three parts: alarm feature extraction engine based on multivariate process condition data, alarm classification model, and alarm filtering

model. The alarm feature extraction engine is responsible for extracting the features of the alarms by incorporating multivariate process condition data from alarm database and process condition database. It is the inlet of alarm filtering framework, receiving and processing real-time collected alarm data. It is also the underlying foundation of the alarm classification model and the alarm filtering model.

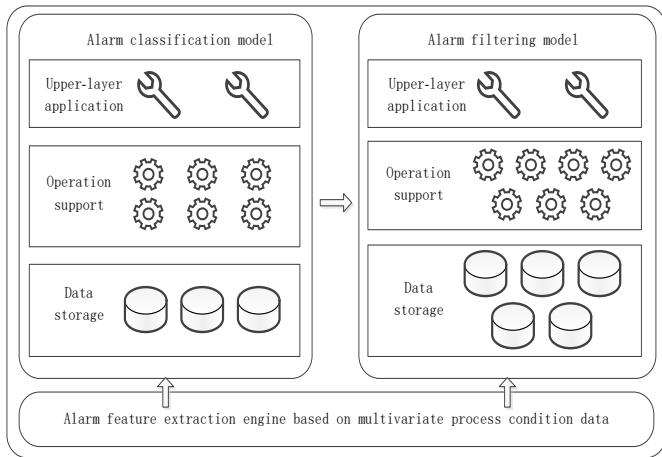


Fig. 1. The integral structure diagram of the developed alarm filtering framework

Alarm filtering model, which uses multivariate process condition data-based alarm features and alarm classification results to filter alarms, is the outlet of the alarm filtering framework. It is responsible for exporting the filtered results of the alarms. Alarm filtering model includes three layers, from bottom to top: data storage layer, operation support layer, and upper application layer. The data storage layer includes 5 databases; the operation support layer includes 7 components; and the upper application layer includes two tool applications.

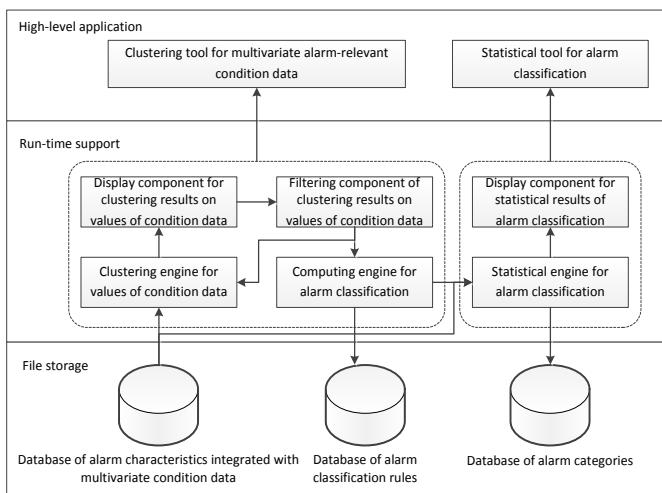


Fig. 2. The integral structure diagram of the developed alarm filtering framework

Fig. 2 is the alarm filtering model framework. The integral structure of the model includes three layers, from bottom to top: data storage layer, operation support layer, and upper application layer. Data storage layer is the basis layer of the

model, responsible for storage of input data and output data for the entire model, through Oracle database and file storage. Operation support layer is the middle layer of the model, which is responsible for connecting data storage layer and upper application layer. Its main task is to provide a series of calculation engines and intermediate components, offering support for upper layer calculation, processing, and results display using input data from file storage layer. Upper application layer is the application layer of the model, which directly interacts with users and provides users with the interfaces for parameter setting and applications. It is responsible for calling the calculation engines of operation support layer and intermediate components for corresponding calculations, processing, and displaying in accordance with user commands.

In the data storage layer, fault repair database is stored through Oracle database, which records the repair service orders of all the pump trucks of a machinery manufacture from Oct 1st, 2012 to June 30, 2013. Fault repair records of pump trucks of model 12BC5313 can be extracted from this database. The alarm statistical index rules and another three databases are stored using file storage. They are multivariate process condition data-based alarm feature database, alarm statistical index database, alarm statistical index summary database, and alarm class database. Alarm statistical index rules are the statistical rules for alarming. They are used in statistical analysis of the alarms, the results from which are stored in the alarm statistical index database.

The upper-layer application library consists of two tools: pre-/post-fault alarm statistical tool and alarm class filtering tool. The pre-/post-fault alarm statistical tool uses the alarm statistical index summary component and the detailed information of alarm statistical indexes to display the alarm statistical indexes of each part before and after the equipment failure. The alarm class filtering tool uses the alarm statistical index classification engine, the alarm class statistical index variation analysis engine, and the alarm class filtering result display component, to identify and discard the alarms of low impact, and display the filtered results.

These three layers together comprise the alarm filtering model framework, which is able to identify the degrees of impact of alarm classes, as well as to achieve the objective of discarding the low-impact alarms. Next, we will start with the pre-/post-fault alarm statistical tool and the alarm class filtering tool, by giving the details of their workflow and the components needed in the operation support layer.

#### A. Design of Pre-/Post-fault Alarm Statistical Tool

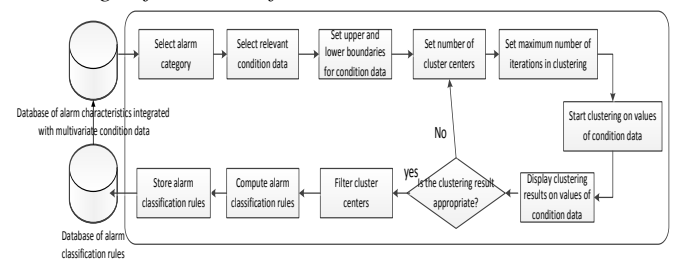


Fig. 3. Workflow of the pre-/post-fault alarm statistical tool

Fig.3 shows the workflow of the alarm-relevant multivariate process condition data clustering tool. First, select the fault repair data of the specified equipment from the fault repair database and extract the corresponding fault repair time point. Then, set the extraction time range of the alarm before and after the fault. Select the class of alarm from the multivariate process condition data-based alarm feature database and extract corresponding alarms before and after the fault. Next, for each failure, use the alarm statistical index rules to calculate the statistical indexes of the alarms before and after the fault, respectively. Store the results in alarm statistical index database. Subsequently, follow the equipment parts to compile all the calculation results, and store them in the alarm statistical index summary database.

Finally, via the alarm statistical index summary displaying component, display the summary of pre- and post-fault alarm statistical indexes of each part of the equipment for users. The users can sort and display these results according to different alarm statistical index summaries. If a particular part is chosen, the alarm statistical index detailing component can display the entire fault repair data set of this part. Moreover, for each fault repair, it can display the corresponding alarm statistical indexes before and after the fault.

### B. Design of Alarm Classification Filtering Tool

Fig. 4 shows the workflow of the alarm classification filtering tool. As a first step, select the equipment part from the alarm statistical index database and extract the corresponding statistical indexes. Then, select all or partial fault data of this equipment part and extract the corresponding alarm statistical indexes. Next, select the frequently occurring alarm classes from the alarm class database and use them to classify all the alarm statistical indexes, obtaining the alarm classification statistical index. Compare the change of alarm class statistical indexes before and after the fault. Alarm classes can thus be filtered by calculating the degrees of impact of the alarms based on the index variation results. Finally, display the filtered results of alarm classes.

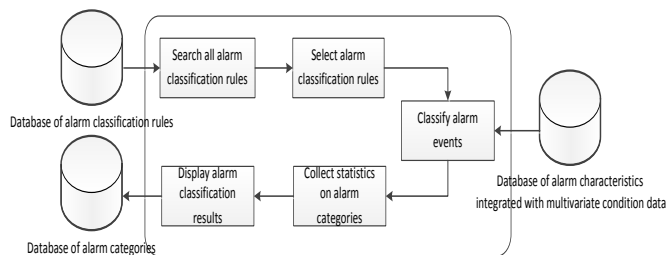


Fig. 4. Workflow of the alarm classification filtering tool.

### C. Alarm Classification Statistical Engine

Before and after the fault, the alarm statistical engine will first use fault repair data to calculate all the corresponding pre- and post-fault alarm statistical indexes, totaling 10 statistical indexes. They include total alarm count  $totalC$ , total alarm duration  $totalD$ , maximum alarm count per day  $maxC$ , and maximum alarm duration per day  $maxD$ . For each statistical index  $statIndex$ ,  $statIndex_{before}$  represents the statistical index before the fault, whereas  $statIndex_{after}$  represents

the statistical index after the fault. In addition, by calculating the total alarm count before the fault  $totalC_{before}$  and the total alarm count after the fault  $totalC_{after}$ , an effective alarm count reduction rate can be obtained, which is defined as  $minusRateC = (totalC_{before} - totalC_{after}) / totalC_{before}$ . By calculating the total alarm duration before the fault  $totalD_{before}$  and the total alarm duration time after the fault  $totalD_{after}$ , an effective alarm duration reduction rate can also be obtained, which is defined as  $minusRateD = (totalD_{before} - totalD_{after}) / totalD_{before}$ .

After obtaining the alarm statistical indexes corresponding to each fault repair, specific values can be viewed from the display component of alarm statistical index details, which can assist analyze whether each fault repair is closely related to the frequently occurring alarm. After obtaining the summary of each alarm statistical index for every part of the equipment, different alarm statistical indexes summaries can be chosen as the sorting rule. All the equipment parts can be sorted in a descending order of their values. The sorting results can be displayed by the alarm statistical index summary displaying component.

The algorithm for alarm class filtering is described here. First, for each evaluation index, by setting the corresponding filtering threshold, an alarm class can be discarded if its evaluation index is below the threshold; display the remaining classes. Then, calculate the intersections of all alarm classes filtered by the evaluation indexes, and display it as the final result of alarm class filtering.

## IV. SYSTEM IMPLEMENTATION AND RESULT ANALYSIS

On the basis of aforementioned multivariate process condition data-based alarm feature extraction method, as well as the alarm classification model and alarm filtering model design, along with the alarm data and process condition data stored in Cassandra database and the fault repair data stored in Oracle database, now let us turn our attention to the implementation of the alarm filtering framework. We will introduce the details of the upper layer application and analyze the alarm filtering results. The alarm filtering framework is realized using Java in this work. Input data are stored using Cassandra database and Oracle database, whereas output data are stored using the means of file storage.

The purpose of this experiment is to use the proposed alarm filtering system to filter the speeding fault alarms from pump trucks. The studied equipment objects are 583 12BC5313-type pump trucks from a subsidiary of Guangdong Power Grid., The over speed fault alarm-related process condition data and fault repair data within a 9-month period from October 1, 2012 to June 30, 2013 are adopted in the present analysis. The fault repair data are utilized to filter the speeding fault alarms. Firstly, the fault repair data are used to calculate each statistical index. Then the 10 most frequent alarms are filtered based on alarm presence rate and effective alarm rate, and the filtered results are subsequently analyzed. In this experiment, the issue of insufficient water pump pressure is taken as an example of fault for analysis.

Water pump is the power supply component of the water washing system of hydraulic pump truck. The potential causes for pump failure include damage, wear, and insufficient pressure. This experiment aims to analyze the fault of

insufficient pressure. 8 pump trucks are adopted for calculation and analysis, 6 of which have the fault repair data for insufficient pump pressure.

TABLE I. ALARM PRESENCE RATES AND EFFECTIVE ALARM RATES OF ALL ALARMS FROM WATER PUMPS

	<i>totalC</i>	<i>totalD</i>	<i>maxC</i>	<i>maxD</i>	<i>maxDayByC</i>	<i>maxDayByD</i>
Alarm presence rate	1	1	1	1	1	1
Effective alarm rate	0.38	0.62	0.38	0.62	0.38	0.62

TABLE II. Alarm PRESENCE rates of different alarm classes under various statistical indexes for water pumps

	<i>totalC</i>	<i>totalD</i>	<i>maxC</i>	<i>maxD</i>	<i>maxDayByC</i>	<i>maxDayByD</i>
AAAA	0.75	0.75	0.75	0.75	0.75	0.25
BBAA	0.25	0.25	0.25	0.25	0.25	0.12
CCAA	0.12	0.12	0.12	0.12	0.12	0
DDAA	0.62	0.62	0.38	0.38	0.62	0.5
BAAA	0.25	0.25	0.25	0.25	0.25	0.12
DAAA	0.12	0.12	0.12	0.12	0.12	0.12
ADAA	0.38	0.38	0.38	0.38	0.38	0.38
DDBA	0.12	0.12	0.12	0.12	0.12	0
CAAA	0.25	0.25	0.25	0.25	0.25	0.12
DDAB	0	0	0	0	0	0

TABLE III. EFFECTIVE alarm rates of class AAA under different statistical indexes

	<i>totalC</i>	<i>totalD</i>	<i>maxC</i>	<i>maxD</i>	<i>maxDayByC</i>	<i>maxDayByD</i>
AAAA	0.17	0.17	0.17	0.17	0.83	1

TABLE 1 shows the alarm presence rates and effective alarm rates of all speeding fault alarms under each statistical index for water pump. We can see that the alarm presence rates under various statistical parameters all are 1, which demonstrates that speeding fault alarm does set off each time before the water pump shows insufficient pressure. Since from the overall effective alarm rate distribution it is not possible to determine whether speeding fault alarm has impact on water pump, it is necessary to calculate and analyze the top 10 most frequently occurring alarm classes.

TABLE 2 shows the alarm presence rates of different alarm classes under different statistical indexes for water pump. It can be seen that under different statistical indexes, only the alarm presence rates of class AAAA are close to or slightly above 0.7, which indicates that this class of alarms is likely to happen before the fault. Now let us continue the analysis on the effective alarm rates of this class.

TABLE 3 shows the effective alarm rates of class AAAA under various statistical parameters for water pump. It appears that the effective alarm rates of this class under majority of the statistical parameters are very low. Therefore, after filtering, the top 10 most frequently occurring alarms are all eliminated, indicating that speeding fault alarm has no correlation with the water pump. That is, it is not feasible to determine whether water pump pressure is insufficient via the speeding fault alarm.

## V. CONCLUSION

In this paper, the filtering method for frequently occurring alarms is studied. An alarm filtering framework is proposed. This framework classifies alarms according to working conditions with the help of relevant multivariate process condition data, and distinguishes the degree of impact of alarms using the fault repair data. Thus, the frequently occurring alarms can be filtered. That is, alarms with higher degrees of impact will be retained. The user will also be informed of these alarms, which are indicative of possible equipment failure. The alarm filtering model is proposed. The pre-/post-fault alarm statistical tool and the alarm classification tool are also developed. Using the model, users can check the fault repair data of each part of the equipment, as well as the summary of statistical indexes and detailed information of each index before and after the fault alarms. By adopting the alarm classification results, users can also classify the alarm statistical indexes and thereby conduct filtering and analysis for each class through the use of alarm presence rates and effective alarm rates. Finally, with the obtained evaluation criteria for different statistical indexes, various alarm classes can be employed to predict possible equipment failure. In the case study performed in this work, by filtering the frequently occurring speeding fault alarms, we conclude that speeding fault alarm is completely not correlated with insufficient water pump pressure.

In this paper, the cluster classification of process condition data is conducted through calculation and analysis of the data, without the involvement of mechanism knowledge. As a result, the clustering results obtained may be of little practical value. The future clustering algorithm for process condition data can proceed from this perspective.

This study only uses the fault repair data to identify the degree of impact of the alarm, without considering the false alarm data. Follow-up study can thus further analyze the change of statistical indexes before and after the false alarm to determine the impact of alarms, so as to make filtering more accurately.

#### REFERENCES

- [1] Rothenberg D H. Alarm Management for Process Control: A Best-Practice Guide for Design, Implementation, and Use of Industrial Alarm System[M]. Momentum Press, 2009: 3-4.
- [2] Hollifield B R, Habibi E. The alarm management handbook: a comprehensive guide[M]. Pas, 2006.
- [3] Bergquist T, Ahnlund J, Larsson J E. Alarm reduction in industrial process control[C]//Emerging Technologies and Factory Automation, 2003. Proceedings. ETFA'03. IEEE Conference. IEEE, 2003, 2: 58-65.
- [4] Kondaveeti S R, Shah S, Izadi I. Application of multivariate statistics for efficient alarm generation[C]//Fault Detection, Supervision and Safety of Technical Processes. 2009: 657-662.
- [5] HU You-le. Analytical systems design and application on working conditions data based on statistical indicators [Master thesis]. Beijing: School of Software, Tsinghua University, 2014.
- [6] YAO Xin-xin, LIU Ying-bo, ZHAO Jiong, et al. Device group-oriented method for abnormal floor data detecting [J]. Computer Integrated Manufacturing Systems, 2013, 19(12): 2993-3001.
- [7] EEMUA 191, "Alarm Systems – A Guide to Design, Management and Procurement", 2nd Edition, 2007.
- [8] ZANG Hao, LI Hongguang, YANG Fan et al. Analysis and improvement of traditional assessment methods for process industrial alarm systems [J]. CIESC Journal, 2014, 65(11): 4459-4464.