Control Engineering Practice 45 (2015) 110–122

Contents lists available at ScienceDirect

Control Engineering Practice

journal homepage: www.elsevier.com/locate/conengprac

A weighted dissimilarity index to isolate faults during alarm floods

S. Charbonnier [a,*], N. Bouchair [b], P. Gayet [b]

[a] Gipsa-Lab, Univ. Grenoble Alpes & CNRS, Grenoble, France
[b] CERN, Geneva, Switzerland

ARTICLE INFO

Article history:
Received 26 January 2015
Received in revised form
7 September 2015
Accepted 8 September 2015
Available online 27 September 2015

Keywords:
Fault recognition
Pattern matching
Dissimilarity index
Alarm vectors
Decision support

ABSTRACT

A fault-isolation method based on pattern matching using the alarm lists raised by the SCADA system during an alarm flood is proposed. A training set composed of faults is used to create fault templates. Alarm vectors generated by unknown faults are classified by comparing them with the fault templates using an original weighted dissimilarity index that increases the influence of the few alarms relevant to diagnose the fault. Different decision strategies are proposed to support the operator in his decision making. The performances are evaluated on two sets of data: an artificial set and a set obtained from a highly realistic simulator of the CERN Large Hadron Collider process connected to the real CERN SCADA system.

1. Introduction

Modern industrial processes are increasingly complex. They are composed of many controlled systems equipped with a large number of sensors and actuators. To ensure system productivity and safety, constant monitoring is required of operators. One of their goals is to detect occurring faults. To assist operators, supervision systems, often based on SCADA (Supervisory Control and Data Acquisition) solutions, are implemented. A SCADA system is made of different layers: a field layer formed of sensors and actuators, a control layer where control algorithms are processed and alarms are generated; and a supervision layer where valuable information, such as sensor values and alarms, is displayed to the operators.

The control system triggers alarms to draw the attention of the operator. Alarms inform the operator that a process variable is out of the ranges determined to be acceptable by preset thresholds or that an equipment is not functioning properly. Alarms are very easy to configure in modern control systems and thus many of them are integrated in the control systems for safety reasons. Because of the high number of alarms that can be raised and because industrial systems are formed of inter-connected parts, the occurrence of a fault usually raises not one single alarm referring to the occurring fault but tens or hundreds of alarms appearing in a short period of time (Beebe, Ferrer & Logero, 2013). This situation is known as an alarm flood. From ISA standards 18.2 (Management of Alarm Systems for the Process Industries, 2009), an alarm flood occurs when the control system raises more than 10 alarms in 10 min. This flood of information overwhelms the operator because it exceeds his response capability. However, the alarm list should be analyzed in depth by the operator to make a diagnosis of the fault. It is, therefore, important to develop methods that can provide operators with hints on the source of the problem to enable them to select the appropriate recovery actions.

The aim of this paper is to present algorithmic solutions to assist the operator in his decision making when confronted with an alarm flood. The goal is to develop a system that makes a diagnostic of the process – i.e. tells the operator which fault caused the alarm flood – by analyzing the list of alarms raised during the flood. The method proposed should not be dedicated to a specific process but should be adaptable to any process.

1.1. Fault isolation methods using discrete events

One way to help the operator is to reduce the large amount of information delivered by changing alarm parameters so as to decrease the number of alarm raised, by summarizing the alarms or by eliminating useless ones. This way, the operator can focus only on the relevant data. This topic, summarized under the categories of alarm rationalization, alarm correlation techniques or alarm management, has been widely researched in recent decades (Salah, Maciá-Fernández & Díaz-Verdejo, 2013; Izadi, Shah & Shook, 2009).

To reduce the number of alarms produced, some authors propose to develop moving average filters to smooth the signal before setting a detection threshold (Cheng, Izadi & Chen, 2013), others

* Corresponding author.
E-mail address: Sylvie.Charbonnier@gipsa-lab.grenoble-inp.fr (S. Charbonnier).

http://dx.doi.org/10.1016/j.conengprac.2015.09.004

propose to adjust the alarm settings (threshold, dead-band, and delay-timers) so as to optimize the alarm production i.e. obtain the best trade-off between false alarm rate, missed detection rate and detection delay (Yang, Shah & Xiao, 2010; Kondaveeti, Izadi, Shah & Chen, 2011; Naghoosi, Izadi & Chen, 2011; Adnan, Izadi & Chen, 2011). To reduce the amount of information displayed to the operator, another solution is to detect irrelevant alarms raised by the alarm system. Irrelevant alarms are chattering alarms (Kondaveeti, Izadi, Shah, Shook & Kadali, 2013; Naghoosi, Izadi & Cheng, 2011) – alarms that altern between normal and abnormal states during a short period of time – and related alarms (Folmer & Vogel-Heuser, 2012). Two alarms are related when one is just the consequence of the other, when they systematically appear in a short period of time or when they are related to the same event. The methods look for alarms that are correlated in time i.e. they systematically appear with some given event or alarm in a given period of time (Folmer, Schuricht & Vogel-Heuser, 2014). Graphical tools may also be used to discover visually related alarms (Kondaveeti, Izadi, Shah & Black, 2010; Yang, Shah, Xiao & Chen, 2012).

Alarms, or, more generally, discrete events, may be grouped into larger classes by clustering methods (Chen & Lee, 2011) or cleaned up by detecting frequent patterns using data-mining methods (Mannila & Toivonen, 1997; Dousson & Vu Duon, 1999).

The following is focused on the more specific task of fault isolation, which consists of finding the exact cause of the alarm flood. Fault isolation has long interested the automatic control community. However, most of the methods proposed suppose that an in-depth knowledge of the system is available. The knowledge may be summed up by expert rules (Bernard & Durocher, 1994) or by a model of the system. The model may represent the normal behavior of the system. In this case, a deviation from the model, obtained by comparing the lists of discrete events (inputs and outputs) generated by the system being monitored and by the model, gives information on the fact that a fault occurred (Roth, Lesage & Litz, 2009). A model may also represent the behavior of the system when a specific fault occurs, which enables the diagnosis to be made directly by comparing the sequences of events (Danancher, Roth, Lesage & Litz, 2011). Various representations may be used to model a discrete event system such as finite automata, Petri nets (Mansour, Wahab & Soliman, 2013; Cabasino, Giua, Pocci & Seatzu, 2011), fault trees (Philippot, Sayed-Mouchaweh, Carré-Ménétrier & Riera, 2011; Hurdle, Bartlett & Andrews, 2009), templates (Pandalai & Holloway, 2000; Palshikar & Khemani, 1999), and chronicles (Cordier & Dousson, 2000). However, the elaboration of a model may be a difficult task when the process is complex and requires the involvement of experts. Moreover, the model is dedicated to a given system and cannot be used on any other system.

Data-driven approaches are an alternative to model-based approaches. In this case, information on the process is provided by a set of historical data which are used to make decisions. Data may be represented either by a vector of alarms or by a sequence of alarms. In a vector, alarms are not ordered in time: only the fact that they occurred is considered. In a sequence the alarms are ordered by their time of appearance.

When data are represented by a vector of alarms, pattern recognition methods may be used. Classifiers are learned from the data and used to assign a fault class to the alarm flood (Ganyun, Haozhong, Haibao & Lixin, 2005; AL-Jumah & Arslan, 1998). For instance, (Negnevitsky & Pavlovsky, 2005) used three connected multiple-layer perceptrons to detect failures in protection relays and (Chen, Qiu, Feng, Tavner & Song, 2011) used a multiple layer perceptron to detect pitch fault in a wind turbine using SCADA alarms. Charkaoui (2005) used decision trees to detect failures in cars using alarms from the off-board computer. Another solution that does not require a training phase is to compare the vector of alarms to expertized cases using scalar products (Fritzen, Montagner Zauk, Cardoso, de Lima Oliveira & Bassi de Araújo, 2012), the Hamming distance (Yemini, Kliger & Mozes, 1996), the Jaccard distance (Ahmed, Izadi, Chen, Joe & Burton, 2013), or a more complex similarity measure (Charkaoui, Dubuisson, Ambroise & Boatas, 2005).

When data are represented by sequences, the order of appearance is considered and this information is used in the methods proposed. Some solutions consist in the data automatically generating a model that explains the sequences of the observed alarms. Bayesian networks, which learn the causal relationships between alarms on the occurrence of a fault and represent them using an acyclic graph, are a popular method (Yamaguchi, Inagaki & Suzuki, 2012; Chen, Tavner & Feng, 2012). Hidden Markov models represent a system as a sequence of states where specific alarms may be generated. Abductive reasoning networks describe the cause-effect relationships between faults and alarms with an acyclic graph (Sun, Guo, Zhang & Zhang, 2012). Petri nets may also be generated from a pool of data (Lefebvre & Leclercq, 2011). However, these methods require a large amount of data to uncover the statistical relationships represented by the models. These data are not always available – particularly data recorded during fault situations, which are hopefully relatively rare.

Competing methods are based on sequence-matching algorithms. These methods, widely used in bio-informatics to compare gene sequences, are based on a similarity measure. Two sequences can be compared with these algorithms. First, they are optimally aligned, then a distance between the aligned sequences is calculated. These methods were used to detect abnormalities in a sequence of events (Chandola, Banerjee & Kumar, 2012; Budalakoti, Srivastava & Otey, 2009), to cluster sequence floods of alarms in a chemical process (Cheng, Izadi & Chen, 2013; Ahmed et al., 2013) and to isolate faults from alarm floods (Charbonnier, Bouchair & Gayet, 2014).

1.2. Approach proposed

The approach presented in this paper is a data driven one that can be applied to various systems, as opposed to model-based methods, which suffer of a lack of adaptability because they require an in-depth knowledge of the process to be able to build the model. It is based on a pattern-matching approach and is less greedy in data than automatic model-elaboration methods or classifiers that need training. The alarm flood generated by an unknown fault is compared to a set of alarm lists stored in a case base by means of a weighted dissimilarity measure. Each alarm list of the case base represents a flood that was recorded in the system on the occurrence of a fault, diagnosed by an expert. The alarms that form the list are the alarms raised by the control system from the beginning to the end of the alarm flood. The lists are represented by a vector of alarms and not by a sequence of alarms, which means that the order of appearance is not used by the method. The idea of the method proposed is that very few of the alarms that form the alarm flood are relevant to diagnose the fault. These alarms are relevant because they are systematically raised on the occurrence on the fault but never raised on the occurrence of another fault (or never raised on the occurrence of the fault and always raised on the occurrence of another fault). A diagnosis could be made by focusing on only these specific alarms, regardless of the other alarms and of their order of appearance. Therefore, a method that can pick out these few relevant alarms and use them in the diagnosis process should be able to make an accurate diagnosis. To extract the relevance of each alarm to a fault, several examples of the same fault are supposed available and summarized into a fault template.

Some papers in the literature propose methods to compare