Control Engineering Practice 55 (2016) 13–25

Contents lists available at ScienceDirect

Control Engineering Practice

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

# A local alignment approach to similarity analysis of industrial alarm flood sequences ☆

Wenkai Hu [a], Jiandong Wang [b,*], Tongwen Chen [a]

[a] Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Alberta, Canada
[b] College of Engineering, Peking University, Beijing, China

ARTICLE INFO

Article history:
Received 16 November 2015
Received in revised form
1 April 2016
Accepted 30 May 2016
Available online 30 June 2016

Keywords:
Industrial alarm systems
Alarm floods
Alarm priority
Local sequence alignment

ABSTRACT

Similar alarm sequence alignment algorithms have been used to find similar alarm floods in the historical database for the prediction and prevention of alarm floods. However, the existing modified Smith–Waterman (SW) algorithm has a high computation complexity, preventing its online applications within a tolerable computation time period. This paper proposes a new local alignment algorithm, based on the basic local alignment search tool (BLAST). The novelty of the proposed algorithm is three-fold. First, a priority-based similarity scoring strategy makes the proposed algorithm more sensitive to alarms having higher alarm priorities. Second, a set-based pre-matching mechanism avoids unnecessary computations by excluding all irrelevant alarm floods and alarm tags. Third, the seeding and extending steps of the conventional BLAST are adapted for alarm floods, which reduce the searching space significantly. Owing to the novelties, the proposed algorithm is much faster in computation and provides a higher alignment accuracy than the SW algorithm. The efficiency of the proposed algorithm is demonstrated by industrial case studies based on the historical alarm floods from an oil conversion plant.



## 1. Introduction

An alarm flood refers to a situation during which the alarm rate is too high, exceeding the ability of industrial plant operators to manage occurred alarms in a prompt manner. The industrial standard ANSI/ISA-18.2 (ISA 18.02, 2009) says that an alarm flood is present when more than 10 alarms occur within 10-min time period. Alarm floods could be caused by many factors, e.g., abnormal situations, improper alarm system design, and operating state transitions. In practice, alarm floods should be limited to less than 1% of the total time period that an industrial alarm system is in operation (EEMUA-191, 2013). However, alarm floods often appear in the existing alarm systems (Beebe, Ferrer, & Logerot, 2013).

In the presence of alarm floods, a large amount of annunciated alarms may not be manageable by operators. As a result, critical alarms may be overlooked, and various negative consequences could arise due to lack of responses to these critical alarms. The consequences include making a dangerous process situation much worse, increasing the risk of process upsets, or even deteriorating to serious accidents. Moreover, when the alarm rate is too high, operators have no choice but to ignore many of the annunciated alarms. In this case, the designed functionality of alarm systems is partially or even completely lost. Thus, the equalities “floods=incidents=loss” are likely to be valid (Beebe et al., 2013). As an example, 275 alarms had to be handled by two operators during the 10.7 min before the explosion accident occurred in the Texaco Refinery at Milford Haven (HSE, 1997); apparently, the operators failed in handling these alarms due to such a high alarm rate.

Some research work has been carried out on alarm system rationalization in order to reduce the occurrence number of alarm floods or alleviate the severity of alarm floods. Plant connectivity and alarm logs were combined to reduce the number of alarms by grouping alarm messages associated with a common root cause (Schleburg, Christiansen, Thornhill, & Fay, 2013). Based on the dependencies between fault events and the precedence of alarm messages, a dynamic fault tree was developed to generate filtering rules for false alarms (Simeu-Abazi, Lefebvre, & Derain, 2011). To remove alarm floods or to mitigate their effects, two advanced alarm handling techniques were presented: the state-based alarming reduces alarm messages by suppressing alarms for different process states and the alarm load shedding strategy displays

☆ This work was partially supported by the Natural Sciences and Engineering Research Council of Canada under Grant No. CRDPJ-446412-12, the National Natural Science Foundation of China under Grant No. 61433001, and the 111 Project of China (B08015). A preliminary version of the work was published in the Proceedings of the International Symposium on Advanced Control of Chemical Processes in Whistler, Canada, June 7–10, 2015.

* Corresponding author.
E-mail addresses: wenkai@ualberta.ca (W. Hu), jiandong@pku.edu.cn (J. Wang), tchen@ualberta.ca (T. Chen).

http://dx.doi.org/10.1016/j.conengprac.2016.05.021
0967-0661/

only the most critical alarms during alarm floods (Jerhotova, Sikora, & Stluka, 2013). To avoid alarm floods during transition of operating states, a dynamic alarm management strategy was provided by dynamically changing alarm limits based on a priori knowledge of transitions (Zhu, Shu, Zhao, & Yang, 2013). Because alarm floods are usually the results of a primary event and its consequential events (Timms, 2009), an alarm flood is often composed of a series of consequential alarms. Thus, the guideline EEMUA-191 (Brown, 2003; EEMUA-191, 2013) is recommended to group consequential alarms to reduce the number of alarms during alarm floods. To avoid alarm flooding and find out root alarms, Wang, Li, Huang, and Su (2015) proposed a method to identify consequential alarms with the combination of alarm similarity analysis and process variable causality inference. Based on the concept of frequent pattern mining, a criteria-based alarm flood pattern recognition method was utilized to reduce the overload of alarm information during alarm floods by identifying alarm sequences of causally dependent notifications and displaying them as a single piece of information (Vogel-Heuser, Schütz, & Folmer, 2015). Most-frequent alarm sequences and causal alarms consolidating the alarm sequences were identified to redesign alarm systems for reducing alarm floods (Folmer, Schuricht, & Vogel-Heuser, 2014; Folmer & Vogel-Heuser, 2012). Based on the pattern mining techniques, sequential alarms were found and used for alarm rationalization to remove redundant alarms, identify bad actors, and establish an effective alarm system (Císař, Hošt'álková, & Stluka, 2009; Kordic, Lam, Xiao, & Li, 2007). To help operators to concentrate on most important alarms during alarm floods, an alarm prioritization system was presented to prioritize alarms by calculating the severity of each alarm based on fuzzy-logic rules (Foong, Sulaiman, Rambli, & Abdullah, 2009). Some new alarm presentation techniques were proposed to ease operators the understanding of alarm floods by showing alarmsin time series together with short alarm descriptions (Laberge, Bullermer, Tolsma, & Reising, 2014).

Another research topic is on the prediction and prevention of alarm floods. That is, by comparing an incoming alarm sequence with potentially similar alarm floods in the historical database, it is possible to achieve an early warning of abnormalities, predict the cause of the incoming alarm flood, and take proactive operational actions to prevent the occurrence of an alarm flood and its negative consequences. To the best of our knowledge, the contemporary studies on the prediction and prevention of alarm floods are limited to the very first step of finding similar alarm flood sequences, which are different from above-mentioned extracting consequential alarms. Ahmed, Izadi, Chen, Joe, and Burton (2013) exploited a dynamic time warping (DTW) algorithm to find common sequences among alarm floods. Cheng, Izadi, and Chen (2013) developed a modified Smith–Waterman (SW) algorithm for the local sequence alignment of alarm floods with incorporation of time stamp information.

As the state-of-art, however, the existing methods suffer from the following limitations: (i) The DTW and SW algorithms in Ahmed et al. (2013) and Cheng et al. (2013) align all alarms even if two alarm floods share almost no common alarm tags; such a computation should be avoided. (ii) The computation complexity of the SW algorithm is too high, which may prevent it from online prediction of upcoming alarm floods. (iii) As important attributes of alarm variables, alarm priorities have not been considered by the SW algorithm yet, while it is an intuitively reasonable choice to weight more on alarms with higher priorities in the similar alarm sequence alignment. Motivated by addressing the above limitations, this paper proposes a new local alignment algorithm to find similar alarm flood sequences, based on the basic local alignment search tool (BLAST) formulated in Altschul, Gish, Miller, Myers, and Lipman (1990) and Altschul et al. (1997). Comparing with the modified SW algorithm, the proposed algorithm is much faster in computation and provides a higher alignment accuracy for similar alarm sequences. These improvements are owing to the following three novelties: (i) A set-based pre-matching mechanism is introduced to exclude the comparison between alarm floods with few common alarm tags, and to exclude irrelevant alarm tags in order to avoid their distractions on the subsequent alarm sequence alignment. (ii) The seeding and extending steps of the conventional BLAST are adapted for alarm floods, where only regions of high similarities are preserved, so that the searching space is reduced significantly. (iii) A priority-based similarity scoring strategy is developed so that the proposed algorithm is more sensitive to alarms having higher alarm priorities.

The rest of the paper is organized as follows. Section 2 defines a priority-based similarity scoring strategy. The novel local alignment algorithm is proposed in Section 3. Section 4 presents industrial case studies to illustrate the effectiveness of the proposed algorithm and make a comparison with the modified SW algorithm. Section 5 concludes the paper.

## 2. Similarity scores for alarm floods

This section introduces the mathematical representations of alarm floods and defines a priority-based similarity scoring strategy for comparing two alarm floods.

### 2.1. Representations of alarm floods

Alarm floods are composed of a series of chronologically sorted alarms, each of which generally includes a variety of attributes, such as the tag name, alarm identifier, time stamp, alarm priority, and process description (Kondaveeti et al., 2013). The tag name is the label of a process variable (including both analog and digital variables) associated with an alarm; the alarm identifier describes the alarm type, e.g. PVHI indicates an analog variable exceeding a high alarm limit. A tag name and an identifier jointly compose a unique alarm tag. For instance, if the alarm identifier PVHI is applicable to the tag name 1PT01, then 1PT01.PVHI is a unique alarm tag. The time stamp marks the time instant that an alarm occurs or clears. The alarm priority indicates the importance of an alarm. Hence, an alarm flood $X$ can be described as

$$X = \langle x_1, x_2, \ldots, x_M \rangle, \tag{1}$$

where the symbol $\langle \cdot \rangle$ indicates a sequence, the length $M$ is the total number of occurred alarms in $X$, and the element $x_i$ indicates the $i$-th alarm occurred in the chronological order. We represent $x_i$ by a tuple with three attributes:

$$x_i = (e_i, t_i, p_i). \tag{2}$$

Here $e_i$ is the alarm tag of $x_i$, and $t_i$ and $p_i$ are the corresponding time stamp and alarm priority, respectively. For the ease of computation, the alarm tag $e_i$ is better in a numerical form so that we map all distinct alarm tags in words to numerical symbols. Thus, a numerical alphabet can be constructed as

$$\Sigma = \{1, 2, \ldots, V\}, \tag{3}$$

where $V$ represents the size of the alphabet, equal to the total number of distinct alarm tags. Clearly, the functional relationship between $\Sigma$ and all distinct alarm tags is bijective.

Table 1 shows an industrial example of an alarm flood: the first column lists the alarm tags in the chronological order, the second column gives the numerical symbols of alarm tags, the third column indicates the time stamps, and the last column presents the alarm priorities.