University of Alberta

Advanced Analysis and Redesign of Industrial Alarm Systems

by

Sandeep Reddy Kondaveeti

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Process Control

Department of Chemical and Materials Engineering

Spring 2013
Edmonton, Alberta

DEDICATED TO MY TEACHERS

who taught me more by what they are...
than what they said

Abstract

In the process industry, the alarm system acts as a layer of protection between the Basic Process Control System (BPCS) and the Safety Instrumented System (SIS). The BPCS is designed for automatic regulation of day to day process operation and SIS comes into picture when emergency shutdown is required. There are specific standards worldwide that define how BPCS and SIS system are to be designed and expected to work. However, not many well defined standards are available for design and management of industrial alarm systems mainly due to the heavy involvement of human factors. Alarm management has gained complexity mainly due to increasing size of process plants and also due to the Distributed Control System (DCS) that presents little motivation for limiting the number of variables on which alarms can be configured.

Alarm management lifecycle for maintaining an efficient alarm system as suggested by the International Society of Automation standards (ISA SP18.02) is discussed in this work with emphasis on monitoring and assessment and design stages. In this work, novel tools for assessment of alarm system based on routinely collected alarm event data are proposed and demonstrated. The primary focus of these tools is to identify nuisance alarms such as chattering and redundant alarms. Alarm event data is represented mathematically and indices are proposed to calculate the extent of similarity between two alarms

and also to estimate the extent of chattering in an alarm.

Two of the most commonly used techniques for reducing alarm chatter, delay timers and latches are discussed in detail. Effect of varying the size of on-delay, off-delay timers and latches on the accuracy of detection is discussed in the Receiver Operating Characteristic (ROC) framework from a theoretical view point by modeling them using Markov chains. Use of Return to Normal (RTN) information in addition to alarm events information in designing delay timers is also discussed. Finally, advantages of multivariate techniques such as Principal Components Analysis (PCA) based T^2 and Q statistic as opposed to univariate monitoring are discussed in the same framework using simulation examples and an industrial case study.

Acknowledgements

It is a pleasant feeling to thank everyone who contributed to this study and made it a life changing experience for me. Firstly, I would like to express my deep gratitude to my supervisor Prof. Sirish Shah for his vision, attitude, patient guidance and constant encouragement. Dr. Shah has given me ample freedom, time and exposure to industrial setting which I believe were crucial for developing this thesis. I am very fortunate to have spent significant time with someone like Dr.Shah with vast knowledge and experience. I truly admire and would like to emulate some of his great qualities such as being extremely patient, positive, respectful and determined.

I would like to thank Prof. Shankar Narasimhan for exposing me at an early stage in my career to industrial data analysis and other projects involving advanced process control. I am indebted to my classmate, friend and mentor Dr. Ramsharan Rangarajan for pushing me to pursue graduate studies when I was uncertain about what was in store. My sincere thanks to Dr. Arun Tangirala, Prof. Biao Huang, Prof. Sirish Shah, Dr. Amos Ben-Zvi, Dr. Jong Min Lee, Dr. Nilanjan Ray and Dr. Richard Sutton for their passion to teaching some of the advanced courses relevant to this thesis.

I would like to thank Dr. Iman Izadi for his guidance, encouragement and like-mindedness. We had numerous discussions that were instrumental in developing some important ideas in this thesis. Many thanks to Prof. Tongwen Chen for leading the study and providing constructive feedback on some of the work. Special thanks to Dr. Dave Shook and Dr. Ramesh Kadali for their support and invaluable guidance especially on human factors in industrial operations due to which the dissertation has significant practical application. I must also thank Trevor Hrycay, Rajeev Varma, Jamie Dang, Tim Black, Tim Burton and David Joe for taking time in supporting this study.

I feel extremely fortunate to be part of a powerhouse, the Computer Process Control (CPC) group. The knowledge gained through technical presentations/discussions and the friendships developed during my stint with members of this association will last forever. I also enjoyed being a member of the Advances in Alarm Management and Design group and wish the group a huge success moving forward. I would like to specially thank the following group members for their support and friendship: Fan Yang, Prakash, Harigopal, Sankar, Phanindra, Mridul, Enayet, Venkat, Karteek, Saneej, Pabbi, Abhijit, Barath, Fei Qi, Tim, Salim, Shoukat, Hailei, Mohammad Iqbal, Rumana, Anuj, Yashaswi, Samidh, Aditya, Yue, Ping, Naseeb, Amin, Kabir, Elham and Sridhar. I would like to thank my friends in Edmonton, Hari, Varghese, John, Ananth, Anand, Ravi and Sriram for the wonderful memories I had at Garneau towers and during several outings that made me feel at home. I would like to take this opportunity to also thank my friends Suman, Mirza and Dhanno for tolerating my idiosyncrasies.

I would like to acknowledge the Department of Chemical and Materials Engineering, University of Alberta for providing me the opportunity to pursue my doctoral degree. I also gratefully acknowledge the financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC), Matrikon Inc., Suncor Energy Inc. and the Alberta Innovates Technology Futures (AITF). I would like to thank Lily Laser, Marion Pritchard and Betty Daniel for their help with various administrative requests.

I would like to express my earnest gratitude to my parents and both sisters Supriya (brother in-law Praveen) and Saveena (brother in-law Raghu) who made many sacrifices to raise me and give me this education. I am eternally grateful to my wife, Nishu for her caring love, support and patience without which the thesis wouldn't have been possible.

Contents

List of Tables

List of Figures

1
Introduction

This chapter includes the background and motivation for this relatively new area of research related to alarm management. The objectives and scope of this work are presented followed by a chapter-wise overview of the thesis.

1.1 Background

Application of control technology in process industries is becoming increasingly automated and more complex due to fierce competition, increasing governmental regulations and narrowing and stringent customer specifications. Over the years, advances in control technology and its implementation has lead to reduction of not only the number of operating personnel but also their effort in carrying out repetitive tasks which is an important contributing factor to human errors when such tasks are carried out manually.

1.1.1 Importance of safe process operation

The role of the control room operator is changing with time. In the past, process operators had better sense of the process as they performed tasks such as moving the valve positions manually after observing the tank levels, line temperatures, vessel pressures, etc. Modern plants typically have several control rooms where a small number of operators monitor thousands of control loops over a wide array of process units.

Safety is an important requirement for automatic control systems design. The best way to achieve safe plant operation is to have inherently safe processes, operating on correctly designed and maintained equipment. Total inherent safety, however, is not always achievable in practice. Typical plants involve a spectrum of risks ranging from minor operational problems to those that have the potential to impact the health and safety of workers or the public. Therefore, protection layers should be provided to prevent or mitigate each potentially hazardous event.

1.1.2 Independent layers of protection

Events with more serious potential hazards may require more layers of protection than events with lesser potential impact. A properly designed Basic Process Control System (BPCS), operated by trained and alert operators, is one of the first lines of defense, beyond sound process design, in preventing incident-initiating events from ever occurring. Protection layers beyond the BPCS include relief systems, dump systems, as well as SIS and accident mitigation systems (including emergency response procedures). Typical protection layers in modern chemical process plants are shown in Table 1.1.

Chemical plants typically operate steadily but are subject to constant distur-

Table 1.1: Independent protection layers for process safety [2]

IPL Number	Components
1	Process design
2	Basic controls, process alarms & operator supervision
3	Critical alarms, operator supervision and manual intervention
4	Automatic SIS
5	Physical protection (relief devices)
6	Physical protection (Containment dikes)
7	Plant emergency response
8	Community emergency response

bances in many forms such as varying feed rate, changing ambient temperature, etc. Automatic control systems typically act to mitigate these disturbances to keep the plant close to desired operating regime.

1.2 Role of alarms in the control room

The settings in the basic process control system are intended to keep production within economical operating regime. Alarms and alerts are configured on key process variables and must be activated once the operation deviates from this economical operation. The priority of alarms must depend on how far the operation is from the economical operation and near the safety critical regions of operation. The lower priority alarms (also known as alerts) prompt the control room operator to make small adjustments to the process that include process throughput changes, process condition changes that involve controller setpoint changes and possibly field adjustments. Most alarms are configured to announce process operations in unsafe regime. These alarms prompt the control room operator to take over or monitor the offending automatic controllers and stabilize the process operation before the safety systems are activated to shutdown the operation partly or fully. Thus, alarms play a key role in announcing abnormalities in process operation to the Control Room Operators (CROs). Alarm systems form an integral part of the operator interface in many process industries that involve continuous, batch and discrete processes. The way an alarm is announced to the control room operator depends on its priority. For instance a low priority alarms may only give visible indication (for example via blinking lights) as opposed to the high priority alarms which are almost always audible.

1.2.1 Motivation

Inefficient alarms systems have been a significant cause of industrial incidents and serious accidents [1]. Often either CROs are kept unaware of abnormal conditions due to the failure of appropriate alarms to activate or they may not have activated with sufficient time to permit the CRO to react effectively. Alarm flooding is another scenario where the CROs are overwhelmed with too many alarms thereby distracting the CROs attention from concentrating on critical alarms/issues. In both cases, the very tool that was supposed to warn and guide the operator during an abnormal situation not only failed to do either but created a distraction and additional stress. Eventually, this interference leads to escalation of seriousness by causing what started out as a minor manageable upset to produce major accidents, some progressing to serious disasters. With industries competing hard for optimal production, proper alarm management has never been more crucial. Improperly managed alarm systems are significant contributors to unplanned downtime, which costs plants mote than $20 Billion each year in the US based petrochemical industries alone [6]. An unplanned shutdown can wipe out all the benefits realized from advanced process control strategies.

1.2.2 Major incidents

The following incidents are often cited in literature as motivations for pursuing efficient alarm management.

Three Mile Island nuclear accident: March 28, 1979

Small but measurable amounts of radioactive material was leaked into the air after a series of failures and improper operator actions. There were no direct casualties as a result of this accident. The accident, however, led to serious economic and public relation consequences, and the cleanup process was slow and costly. There is a general consensus that the accident was exacerbated by wrong decisions made because the operators were overwhelmed with alarm information, much of it irrelevant, misleading or incorrect.

Chernobyl Diaster: April 26, 1986

The incident lead to 56 direct deaths and approximately 4000 indirect deaths due to cancer. The accident at the Chernobyl nuclear power plant was the worst nuclear accident in history. The unstable state of the nuclear reactor was not reflected in any way on the operational alarm system for this plant leading to improper judgment by the operators.

The explosion at Milford Haven Oil refinery: July 24, 1994

The explosion caused plant damages that cost over $100 million to repair and very substantial loss of production. Fortunately, the incident occurred on a Sunday, lunch time when the site was fairly empty. 26 people sustained minor injuries. Alarm system shortcomings were the major contributor to this incident. Due to heavy rain and lightening, there were many disturbances to the plant operation. For several hours the operators were loaded with alarms estimated at a rate higher than an alarm every 2 or 3 seconds. During this period the operators failed to identify a sticky valve that eventually lead to an accumulation of liquid into a flare knockout drum which overflowed and resulted in the explosion.

Channel Tunnel Fire: November 18, 1996

The initial fire started, before the train entered the Channel Tunnel. Initial warnings from the fire alarm systems were not acted on as staff awaited confirmation that there was a fire onboard. Subsequent mistakes were made by staff in the Rail Control Centre which made the situation worse. These staff was overloaded with information and was trying to use badly designed alarm systems and procedures.

BP refinery explosion and fire: March 23, 2005 [8]

The BP Texas City Refinery suffered one of the worst industrial disasters in recent U.S. history. Explosions and fires killed 15 people and injured another 180, alarmed the community, and resulted in financial losses exceeding $1.5 billion. The incident occurred during the startup of an isomerization1 (ISOM) unit when a raffinate splitter tower2 was overfilled; pressure relief devices opened, resulting in a flammable liquid geyser from a blow down stack that was not equipped with a flare. One of the main contributing factors was that an alarm meant to warn about the quantity of liquid in the unit was disabled. The release of flammables led to an explosion and fire.

The Buncefield Incident: December 11, 2005 [3]

Tank 912 at the Hertfordshire Oil Storage Limited (HOSL) part of the Buncefield oil storage depot was filling with petrol. The tank had two forms of level control: a gauge that enabled the employees to monitor the filling operation; and an independent high-level switch (IHLS) which was meant to close down operations automatically if the tank was overfilled. The first gauge stuck and the IHLS was inoperable there was therefore no means to alert the control room staff that the tank was filling to dangerous levels. No alarm was configured to flag the failure of level indicator and switch. Eventually large quantities of petrol overflowed from the top of the tank. A vapour cloud formed which ignited causing a massive explosion and a fire that lasted five days. Widespread damage occurred to neighboring properties. About 2,000 people were evacuated from their homes in a 1/2-mile radius from the site, 43 people were injured, but, miraculously, no one was killed. The overall cost of the Buncefield incident was estimated to be close to $1.5 billion.

Deepwater Horizon oil spill: April 20, 2010 [7]

The spill stemmed from a sea-floor oil gusher that resulted from the 20 April 2010 explosion of Deepwater Horizon, which drilled on the BP-operated Macondo Prospect. The explosion killed 11 men working on the platform and injured 17 others. One of the contributing factors mentioned in the investigation reports is bypassing of gas alarms and automatic shutdown systems that could prevent an explosion, and lack of training of personnel on when and how to shutdown engines and disconnect the mobile offshore drilling unit from the well to avoid a gas explosion and mitigate the damage from an explosion and fire.

1.3 Research objectives and scope of this work

The objective of this research work is to identify and study the engineering aspects of alarm management lifecycle [4] that are important for maintaining an efficient alarm system. The main focus has been on monitoring and assessment stage and the detailed design stage of the alarm management lifecycle that will be discussed in more detail in the Chapter 2. The objectives have been developed during several sessions of interaction between members of the Industrial Research Chair program in Computer Process Control in industry and academia.

In short, the objectives can be itemized as follows:

- Develop simple but effective tools for advanced analysis of routinely collected industrial alarm data
- Illustrate and compare the performance of commonly used univariate alarm design techniques on a unified framework [5]
- Investigate the use of multivariate techniques for alarm generation using Principal Components Analysis (PCA)

The work in this thesis does not discuss many other important aspects of alarm management such as alarm philosophy adoption, alarm rationalization procedure and human machine interface design.

1.4 Organization of the thesis

The remainder of this thesis is organized as follows:

- Chapter 2: Introduction to alarm management. This chapter provides a brief review of alarm management in process industries by providing a background to use of alarms in the control room and how alarm systems evolved over time. Definitions for the terminology used on this topic are also provided. A history of alarm management is outlined with pointers to some industry standards. Alarm management lifecycle suggested by the latest standard (ISA 18.02) is discussed in more detail with emphasis on monitoring and assessment and detailed design stages that are of particular interest in this work.

- Chapter 3: Graphical Tools for Routine Assessment of Industrial Alarm Systems. In this chapter, A convenient methodology is proposed to identify nuisance alarms. Alarm events which are usually stored as long strings of text on the historical alarm database are mathematically represented as binary sequences. Two Graphical tools that are instrumental in identifying nuisance alarms are proposed and illustrated with two industrial datasets.

- Chapter 4: Quantification of alarm chatter based on run length distributions. This chapter deals with chattering alarms and provides a means to quantify alarm chatter through run-length distributions. Prominent features of the proposed chatter index and its variant are demonstrated using industrial datasets.

- Chapter 5: On the use of delay timers and latches for efficient alarm design. This chapter discusses delay timers and latches that are often used in the process industry to reduce alarm chatter and to minimize nuisance alarms especially on the digital variables. Effect of varying the size of on-delay, off-delay timers and latches on the accuracy of detection is discussed from a theoretical view point by modeling them using Markov chains. Use of Return to Normal (RTN) information in addition to alarm information in designing delay timers is also discussed with application to a real industrial case study.

- Chapter 6: Application of Multivariate Statistics for Efficient Alarm Generation. This chapter discusses advantages of monitoring the PCA based T^2 and Q statistic over individual process variables. Monitoring these higher level statistics will not only reduce the false alarm and missed alarm rates but also reduces the detection latency which is one of the main drawbacks of monitoring a filtered variable. Two simulation examples and a simple industrial case study are shown to illustrate the utility of the proposed method.

- Chapter 7: Concluding remarks. The main conclusions, contributions of this thesis and some recommendations on future work in this area of research are discussed in this chapter.

Bibliography

[1] M. L. Bransby and J. Jenkinson, *The management of alarm systems: A review of best practice in the procurement, design and management of alarm systems in the chemical and power industries*, Tech. Report CRR 166, Health and Safety Executive, 1998.

[2] CCPS/AIChE, *Guidelines for engineering design for process safety*, Wiley, New York: Center for Chemical Process Safety/American Institute of Chemical Engineers, 1993.

[3] Health and Safety Executive 100021025, *The buncefield investigation: Third progress report*,

http://www.buncefieldinvestigation.gov.uk/reports/volume1.pdf (December 2008).

[4] ISA, *Management of alarm systems for the process industries*, Tech. Report ANSI/ISA-18.2-2009, International Society of Automation, ISA, 67 Alexander Drive, P.O. Box 12277, Research Triangle Park, North Carolina 27709, 2009.

[5] I. Izadi, S. L. Shah, D.S. Shook, S. R. Kondaveeti, and T. Chen, *A framework for optimal design of alarm systems*, In proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, 2009, pp. 651–656.

[6] I. Nimmo, *Adequately address abnormal situation management*, Chemical Engineering progress **91** (1995), no. 9, 36–45.

[7] National Commission on the BP Deepwater Horizon Oil Spill and Offshore Drilling, *Deep water: The gulf oil disaster and the future of offshore drilling*, `http://www.oilspillcommission.gov/sites/default/files/documents/DEEPWATER_ReporttothePresident_FINAL.pdf` (2011).

[8] United States Chemical Safety and Hazard Investigation Board, *Investigation report, refinery explosion and fire*, http://www.csb.gov/assets/document/CSBFinalReportBP.pdf (March 2007).

2

Introduction to alarm management

2.1 Overview

This chapter provides a brief review of alarm management in process industries by providing a background to the use of alarms in the control room and how alarm systems have evolved over time. Definitions for the terminology[1], used on this topic are also provided. A history of alarm management is provided with references to some commonly followed industry standards. Alarm management lifecycle suggested by the latest standard (ISA 18.02) is discussed in more detail with emphasis on monitoring and assessment and detailed design stages that are of particular interest in this work.

[1]A few sections of this chapter have been published as a journal paper: Graphical tools for routine assessment of industrial alarm systems, S.R. Kondaveeti, I. Izadi, S.L. Shah, T. Black and T. Chen, Computers & Chemical Engineering, 2012; 46(0), 39-47

2.2 Background

Definitions of some of the terms frequently used in this thesis are provided in this section.

2.2.1 Alarm

An alarm is an audible and/or visible means of indicating to the operator an equipment malfunction, process deviation, or abnormal condition requiring a response [4]. For every alarm that is configured, written alarm response procedures are typically available at the operator's disposal. Details about the potential causes, consequences and recommended corrective actions along with the response time are ideally available to the operator. In cases where multiple alarms are annunciated within a short interval of time, the action to be taken is up to the operator's discretion and depends very much on the operators understanding of the process.

2.2.2 Alarm design considerations

The most common practice for alarm generation is comparing a variable with a threshold. This variable can be, and most of the times is, an actual raw or a filtered process variable. Alternatively, the variable can come, for example from a multivariate monitoring technique. For example, the squared prediction error in a data-driven PCA model or the residual in a model-based method [8]. The threshold, also known as trip point, control limit or alarm limit, is usually selected based either on the normal operation conditions or the equipment safety considerations.

In the process industry, deadbands and delaytimers are often used to minimize nuisance alarms.

Table 2.1: Dead band recommendations based on signal type

Signal Type	Deadband as % of operating range
Flow Rate	5%
Level	5%
Pressure	2%
Temperature	1%

Alarm deadband

Alarm deadband is defined as the change in signal from the alarm limit necessary to clear the alarm. Use of alarm deadbands do not induce any detection latency. The value of the deadband is usually indicated in terms of the percentage of operating range. Industry standards ([4] & [2]) suggest a starting point based on the signal type as shown in Table 2.1

On-delay & Off-delay timers

The attributes on-delay and off-delay (i.e., filter timer and debounce timer) can be used to eliminate nuisance alarms. The on-delay is used to avoid unnecessary alarms when a signal temporarily overshoots its alarm limit, thus preventing the alarm from being triggered until the signal remains in the alarm state continuously for a specified length of time. The off-delay is used to reduce chattering alarms by locking in the alarm indication for a certain holding period after it has cleared. Use of on delay timers does induce detection latency. Industry standards ([4] & [2]) suggest a starting point based on the signal type as shown in Table 2.2. Proper engineering judgment should be employed when setting 'on' and 'off' delays in order to minimize nuisance alarms while maintaining process vigilance and plant or personnel safety.

Table 2.2: On-delay & off-delay time recommendations based on signal type

Signal Type	**Delay Time (On or Off)**
Flow Rate	15 Seconds
Level	60 Seconds
Pressure	15 Seconds
Temperature	60 Seconds

2.2.3 Types of alarms, their attributes and states

Types of alarms

Both standards: EEMUA [2] and ISA [4], identify several common methods used for detecting alarms. They are as follows:

1. Absolute alarms: An alarm generated when a constant alarm limit is exceeded.
2. Deviation alarms:An alarm generated when the difference between two analog values exceeds a limit (e.g., deviation between primary and redundant instruments or a deviation between process variable and setpoint).
3. Rate of change alarms: An alarm generated when the change in process variable per unit time, (dPV/dt), exceeds a defined limit.
4. Discrepancy alarms: An alarm generated by error between the comparison of an expected plant or device state to its actual state (e.g., when a motor fails to start after it is commanded to the on state).
5. Calculated alarms: An alarm generated from a calculated value based on one or more process measurements instead of a direct process measurement.

6. Recipe-driven alarms: An alarm with limits that depend on the recipe that is currently being executed. This type of alarm is applicable more in batch processes.

7. Bit-pattern alarms: An alarm that is generated when a pattern of digital signals matches a predetermined pattern.

8. Controller output alarms: Alarms based on control system calculated outputs such as valve opening and drive frequency

9. Systems diagnostic alarms: Alarms from the SIS that indicate dangerous faults, depending on considerations (e.g., the operator response).

10. Instrument diagnostic alarms: instrument malfunction or diagnostic alarms.

11. Adjustable alarms: An alarm for which the setpoint can be changed manually by the operator.

12. Adaptive alarms: Is basically a subsection of 'rate of change' or 'deviation' alarms where the alarm limit varies with operating region.

13. Re-alarming alarms: Alarms that are automatically re-annunciated under predefined conditions. For example, if a standing alarm is re-annunciated when the process variable increased by 5% above the original alarm limit.

14. Statistical alarms: An alarm generated based on statistical processing of a process variable or variables.

15. First-out alarms: An alarm determined (i.e., by first-out logic) to be the first, in a multiple-alarm scenario

16. Bad measurement alarms: An alarm generated when the signal for a process measurement is outside the expected range (e.g., 3.8mA for a 4-20mA signal).

Alarm attributes

An alarm message that appears on the operator console has a variety of attributes such as *time stamp*, *tag name*, *alarm identifier*, *tag description*, *plant name*, *area*, *priority*, *alarm setpoint*, *trip value* and so on. Most of the information in the alarm message is used by the operator to identify the root cause of the abnormal event. 'Time stamp' is the time of occurrence of the alarm. 'Tag name' is usually the variable name allotted to the instrument measuring any physical property such as temperature and pressure. 'Alarm identifier' has details about how the limit violation has occurred, for example the low threshold (PVLO) or the high threshold (PVHI) may have been exceeded. The convention for alarm identifier may vary from system to system. For instance, to signify a low process variable, either 'PVLO' or 'PVLOW' is usually used. Depending on the response time available to the operator, each alarm is assigned a priority (Low, High and Emergency are the commonly used priorities). 'Trip value' is the value of the process variable at the instance of alarm occurrence and it may be different from the alarm setpoint. The first three attributes (time stamp, tag name and alarm identifier) are usually sufficient to uniquely identify an alarm event.

Unique alarm

Every alarm that appears on the operator console has a purpose to alert the operator about an abnormal event. An alarm occurring at a particular time instant will be represented by its tag name followed by its identifier (tag-name.id) to include all the necessary information. An alarm with a particular set of tagname and identifier is henceforth defined as a *unique alarm* or simply an alarm. Every unique alarm requires a specific set of operator action(s).

Alarm states

An alarm configured on the DCS can exit in several states depending on the underlying process variable and also the operator action. Most common states are as follows

1. Normal state: An alarm is said to be in normal state when the underlying process variable is operating within normal specifications, the alarm is cleared and any previous alarms have been acknowledged.
2. Unacknowledged state: An alarm is said to be in unacknowledged state when the underlying process variable is violating the alarm limit due to off-target, upset or shutdown process conditions. the alarm is cleared and any previous alarms have been acknowledged. The alarm may be silenced in the unacknowledged alarm state.
3. Acknowledged state: The acknowledged alarm state is reached when an alarm has not been cleared, but an operator has received the alarm and acknowledged the alarm condition.
4. Return To Normal (RTN) Unacknowledged state: The return to normal unacknowledged alarm state is reached when the process returns within the alarm limits and the alarm clears automatically (sometimes called auto-reset) before an operator has acknowledged the alarm condition.
5. Suppressed state: An alarm is said to be in suppressed state when its potential to annunciate wholly or partially is disabled. Typically, there are two types of suppressions, one is when the CRO suppresses the alarm because it is believed to be a nuisance and the other type of suppression happens when the control system automatically suppresses the alarm because it is state based (For example, a low flow alarm is suppressed

by design when the upsteam pump is not running). Sometimes, state based alarms are implemented as disabled alarms, the only difference being that disabled alarms do not appear in the alarm suppressions list. Suppressed and disabled states are also preferred while the plant is not in normal operation (such as startup, shutdown, upset, throughput change, quality change and so on)

6. Out-of-Service: The out-of-service alarm state is used to manually suppress alarms (e.g., use control system functionality to remove alarm from service) when they are removed from service, typically for maintenance. An alarm in the out-of-service state is under the control of maintenance personnel.

2.2.4 Alarm system

The collection of hardware and software that detects an alarm state, communicates the indication of that state to the operator, and records changes in the alarm state constitutes an alarm system. There are two aspects to an alarm system. The first aspect, the engineering aspect involves design of the alarm generating algorithm, which can be anything between a simple limit checking logic on a raw process variable to something as complicated as using machine learning tools for fault classification. A review of such techniques for fault detection and diagnosis has been presented in [8]. The second aspect of the alarm system is unambiguous annunciation of a fault in a convenient manner so that the operator can take timely action towards rectification of the fault. Human Machine Interface (HMI) design is an integral part of the second aspect and has been discussed in much detail in [7].

2.3 History of alarm management

Alarm management is all about the processes and practices for determining, documenting, designing, operating, monitoring, and maintaining alarm systems [4].

Alarm management has existed as long as there have been alarm systems, but it has become more important with the implementation of the distributed control systems (DCS). Prior to the DCS, process monitoring was performed based on a wall mounted panel or board. Due to sparse instrumentation, the number of measurements being monitored was limited and only few of the variables had wall mounted alarms configured on them. Due to space constraints and also the hardware cost associated, adding new alarms to the board had to be justified with due diligence. Figure 2.1 shows a typical control room of this type (taken from `http://www.plantops.umich.edu/utilities/CentralPowerPlant/ph/old_control_room.php`).

However, since the advent of DCS in the late seventies, control and monitoring practices in process industry have changed drastically. The DCS introduced software alarms: alarms that are created or changed by configuring a setting in a computer, rather than requiring a hardwired signal to a panel. These systems also facilitated multiple alarm configuration on a single process variable (Hi, Lo, HiHi, LoLo and so on). As a result, more alarms could be configured at no extra cost and little extra effort. This resulted in increased practice of sloppy alarm design on most of the variables leading to many more alarmed variables than necessary including nuisance alarms. A typical modern day control room is shown in figure 2.2 which is taken from (taken from `http://www.kk.org/thetechnium/archives/2008/04/control_rooms_from_temples_to.php`).

Typically, in a process plant, most of the alarms are configured during the

Boiler Operator Jeff Craigie sits in the Boiler Room and monitors flows, temperatures and pressures of the boilers and feed-water system. Photo by Ryan Solomon

Figure 2.1: An old control room - Central power plant in University of Michigan.
source by:`http://www.plantops.umich.edu/utilities/CentralPowerPlant/ph/old_control_room.php`

Figure 2.2: A new control room - In the Czech Republic, the MERO Middle European Raw Oil control center
source by:http://www.kk.org/thetechnium/archives/2008/04/control_rooms_from_temples_to.php

Table 2.3: A comparison of alarm statistics across various industries with EEMUA benchmark as published by [6] and is a results of an industrial survey conducted by Matrikon Inc.

	EEMUA	**Oil and Gas**	**Petrochemical**	**Power**
Average alarms per hour	≤ 6	36	54	48
Average standing alarms	9	50	100	65
Peak alarms per hour	60	1320	1080	2100
Distribution % (low/med/high)	80/15/5	25/40/35	25/40/35	25/40/35

plant design stage or very early into the plant operation. Once the plant is operating at steady state, a change in alarm configuration is expensive in terms of the person-hours it takes. Poor rationalization and inefficient alarm design are major contributors for the occurrence of nuisance alarms. *Nuisance alarms* are alarms that do not tell the operator anything he/she does not already know, or which do not require operator action. Surveys indicate that on an average, during routine operation, many process industries currently have a much higher alarm annunciation rate compared to what the standards suggest as maximum manageable for efficient operation ([1] & [6]). Table 2.3 compares the alarm statistics over various industries with EEMUA benchmark statistics [5].

Nuisance alarms are major contributors to alarm floods. *Alarm flooding* is a condition where alarms appear on the control panels at a rate faster than the operator can comprehend or respond to. Alarm flooding overloads and prevents the operator from determining the root cause of the process upset and therefore limits the scope for effective and quick emergency response.

2.4 Benefits of alarm improvement

Inefficient alarm systems have been cited as one of the key shortcomings in the investigation reports following many major incidents [1]. Organizations such as Health and Safety Executive (HSE) and the Abnormal Situation Management (ASM) consortium have highlighted the need for designing, implementing and maintaining efficient alarm systems. Several handbooks (such as [6] & [3]) are available as ready references for guidelines on maintaining an efficient alarm system.

2.5 Standards, guides and references

United Kingdom based non-departmental public body, Health and Safety Executive (HSE) sponsored a contract research report [1] to review the best practices in procurement, design and management of alarm systems in the chemical and power industries. The report was published in 1998 and is probably the first report to highlight the need for improvements in alarm management. The report includes case studies and surveys from several sites and also incorporates some existing literature on research and development work aimed at improving the processing, display and interpretation of alarm information. The European design guide, EEMUA (Engineering Equipment and Materials Users Association) 191 has been widely followed in North America and Britain since it first publication in 1999. The design guide was revised in 2007. NAMUR first published its guide NA 102 in Germany. International Society of Automation (ISA) came up with the first complete version of its standard (ISA 18.02) in 2009.

2.6 Alarm management lifecycle

As alarm management is a continual process, a lifecycle approach is suggested by the standards. Fig. 2.3 shows the stages involved in the alarm management lifecycle which is adopted from the standard published by the International Society of Automation (ISA) [4].

2.6.1 Brief description of the stages

Alarm *philosophy* is a document that establishes the basic definitions, principles and processes that are required to design, implement and maintain an alarm system. The *Identification* stage is the collection of potential alarms that are in accordance with the principles outlined in the alarm philosophy. *Rationalization* is a documented thought process to review potential alarms using the principles in the alarm philosophy document. Alarm priority determination is also a part of alarm rationalization.

The *Detailed Design* stage includes the determination of alarm set-points and also any advanced alarming techniques that could be used to make the alarm system efficient. *Implementation* includes not only the physical and logical installation of an alarm but also operator training.

Operation is the stage where the alarm system is active. In the *Maintenance* stage, the alarm system is not active and is tested for its performance against designed standards. In *Monitoring and Assessment*, the performance of the alarm system is compared with the goals outlined in the alarm philosophy. The results of this stage might trigger maintenance or propose changes to the existing alarm system. In *Management of Change*, the changes are approved and documented. These changes follow all stages from identification to implementation. The *Audit* stage involves conducting reviews that could reveal

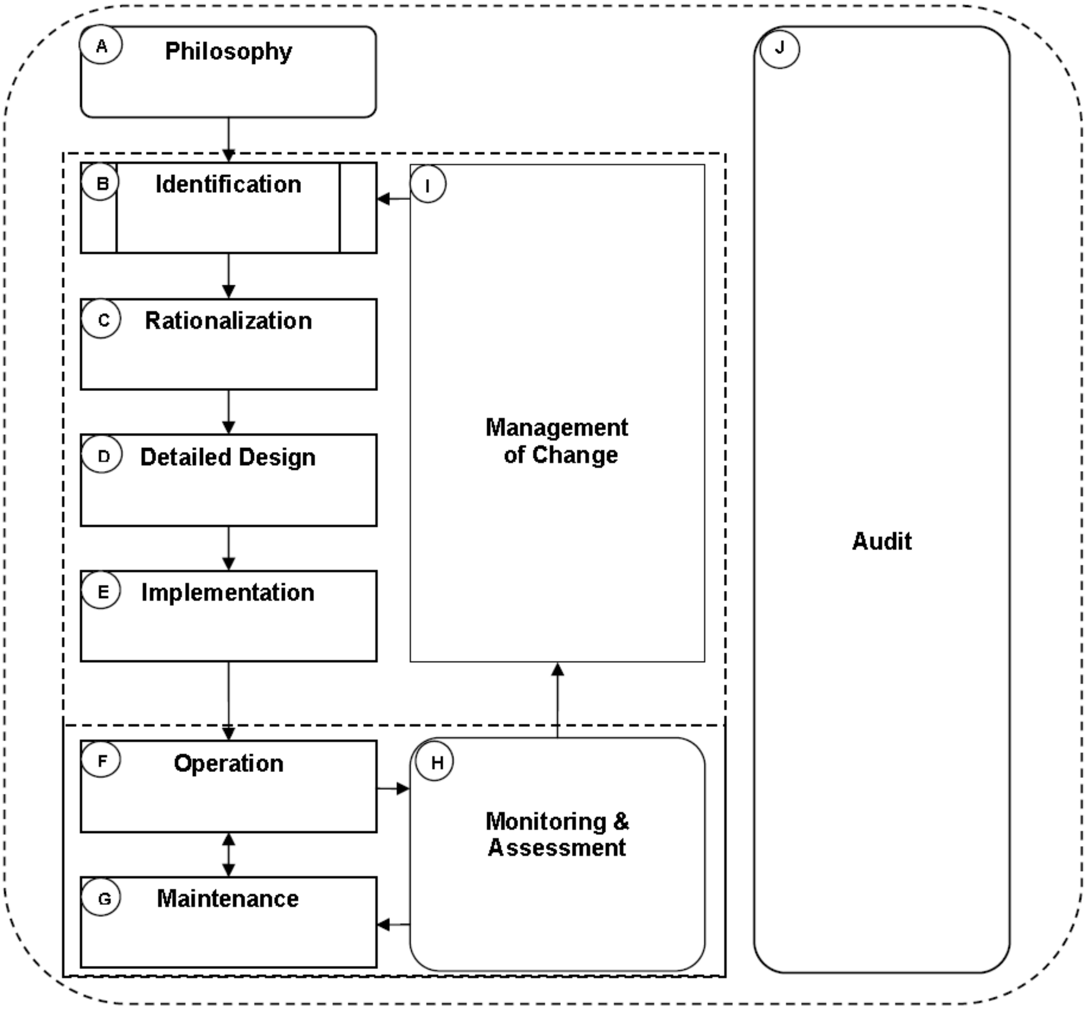


Figure 2.3: Alarm Management Lifecycle [4]

room for improvement which is not apparent from routine monitoring.

2.6.2 Importance of monitoring and assessment stage

For already existing alarm systems, the *Monitoring and Assessment* stage is a good starting points for making any improvements. Alarm system management is a continuous process, and without periodic monitoring and assessment, the performance of the alarm system tends to degrade. The main purpose of the alarm system assessment is to verify that design, implementation, rationalization, operation and maintenance are satisfactory. The problems identified in this stage can be resolved in several others stages of the lifecycle (Design and Rationalization stages for example). A challenging task in the alarm system assessment step is the identification of nuisance alarms such as *chattering alarms* and *related alarms*.

- An alarm is said to be *chattering* if it repeatedly transitions between the alarm state and Return To Normal (RTN) state within a short span of time. As the name suggests RTN is a state when the alarm has cleared. In both EEMUA and ISA standards ([4] & [2]), rules of thumb are suggested to identify chattering alarms. For example, in [4], a threshold of 3 alarms per minute is used to identify worst chattering alarms.

- *Related alarms* are a collection of two or more alarms that almost always occur simultaneously or in tandem without a specific order within a short span of time. *Consequential alarms* are a special case of related alarms where one particular alarm always appears a short duration after another alarm [6]. Redundant alarms are another special case of related alarms that always occur simultaneously or with a specific time delay. Related alarms are usually triggered due to the same root cause and result from poor alarm rationalization. If not identified and acted upon appropriately, related alarms can drastically increase the alarm count during process upsets.

2.7 Concluding remarks

Alarm systems have evolved significantly over the past few decades in terms of the design, implementation and the way human operators interact with them. Most importantly, alarms on the distributed control system are implemented as software configuration which facilitates the use of advanced alarm generating logic and also simplifies the management of change process. The ISA standard deemed alarm management to be a continuous process and recommended to follow a lifecycle approach. The monitoring and assessment and detail design stages present challenges for identifying nuisance alarms and appropriately rectifying alarm design issues.

Bibliography

[1] M. L. Bransby and J. Jenkinson, *The management of alarm systems: A review of best practice in the procurement, design and management of alarm systems in the chemical and power industries*, Tech. Report CRR 166, Health and Safety Executive, 1998.

[2] EEMUA, *Alarm systems: A guide to design, management and procurement*, 2 ed., EEMUA Publication No. 191 Engineering Equipment and Materials Users Association, London, 2007.

[3] B. R. Hollifield and E. Habibi, *Alarm management: Seven effective methods for optimum performance*, ISA, Research traingle park, NC, 2007.

[4] ISA, *Management of alarm systems for the process industries*, Tech. Report ANSI/ISA-18.2-2009, International Society of Automation, ISA, 67 Alexander Drive, P.O. Box 12277, Research Triangle Park, North Carolina 27709, 2009.

[5] I. Izadi, S. L. Shah, D.S. Shook, and T. Chen, *An introduction to alarm analysis and design*, In proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, 2009, pp. 645–650.

[6] D. H. Rothenberg, *Alarm management for process control*, Momentum Press, New Jersey, August 2009.

[7] N. A. Stanton, *Human factors in alarm design*, Taylor & Francis, Inc., Bristol, PA, USA, 1994.

[8] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri, *A review of process fault detection and diagnosis part i: Quantitative model-based methods*, Computers & Chemical Engineering **27** (2003), no. 3, 293–311.

3

Graphical Tools for Routine Assessment of Industrial Alarm Systems

3.1 Overview

In this chapter[1,2], alarm data is represented using binary sequences and subsequently, two novel alarm data visualization tools are presented: 1) The High Density Alarm Plot (HDAP) charts top alarms over a given time period and 2) Alarm Similarity Color Map (ASCM) highlights related and redundant alarms in a convenient manner. The proposed graphical tools are instrumental in performance assessment of industrial alarm systems in terms of effectively

[1]A shorter version of this chapter has been published in proceedings of the 11^{th} IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design, and Evaluation of Human-Machine Systems, August 31 - September 3, 2010, Valenciennes, France

[2]Sections of this chapter have been published as a journal: Graphical tools for routine assessment of industrial alarm systems, S.R. Kondaveeti, I. Izadi, S.L. Shah, T. Black and T. Chen, Computers & Chemical Engineering, 2012; 46(0), 39-47

identifying nuisance alarms such as chattering and related alarms based on routinely collected alarm event data. The special features and advantages of the proposed graphical tools are illustrated by successful application to two large scale industrial case studies, each involving over half a million observations for the top fifty alarm tags.

3.2 Introduction

In control room terminology, an *alarm* is a notification that a fault or an abnormal event has occurred and an operator must take action. The means of notification can be either audible (horn) or visible (flashing lights on the operator's screen). Some highly managed alarms use both means of notification simultaneously. Each alarm message that is annunciated to the operator is also stored as a text message with various fields in an event database. Alarms fall under the second and third out of the eight independent layers of protection according to safety protection layer philosophy [2].

Alarm systems in the process industry play a key role in informing indications of abnormal process conditions or equipment malfunctions to the operators. There are two aspects to an alarm system. The first aspect, the engineering aspect involves design of the alarm generating algorithm, which can be anything between a simple limit checking logic on a raw process variable to something as complicated as using machine learning tools for fault classification. A review of such techniques for fault detection and diagnosis has been presented in [13]. The second aspect of the alarm system is unambiguous annunciation of a fault in a convenient manner so that the operator can take timely action towards rectification of the fault. Human Machine Interface (HMI) design is an integral part of the second aspect and has been discussed in much detail in [11]. In this work, graphical tools that are aimed at assessing the performance of the engineering aspects of alarm systems are proposed.

Most commercially available software for alarm system assessment provide a convenient way to link to alarm databases and query the data that is stored. These tools are equipped with filters that can be used to format and archive useful information in an easily presentable format. Some of these software applications automatically generate alarm assessment reports periodically. These reports usually compare the performance of the alarm system with the alarm activation benchmark statistics suggested by standards ([5] & [3]). The statistics generated include items such as most frequent alarms list, alarm priority distribution, average alarm rate, peak alarm rate, standing alarms list, number of alarm floods and so on. Some of the available software applications allow users to make additional queries to generate trivial statistics using a variety of filters that can be used to identify nuisance alarms like related alarms and chattering alarms. These data mining exercises, however, are not standardized and typically consumes significant period of time to perform the analysis.

The purpose of the work in this chapter is to illustrate simple but effective tools for performance assessment of alarm system using routinely collected alarm data. A convenient methodology is proposed in this work to identify nuisance alarms. Alarm events which are usually stored as long strings of text on the historical alarm database are mathematically represented as binary sequences. Based on the binary sequence representation, a similarity index that measures the extent of correlation between two alarms is proposed. Using the developed tools, historical alarm data of an alarm system can be charted in an easily interpretable form. Nuisance alarms such as chattering and related alarms can be visually identified using these graphical representations. The results of the analysis provide valuable feedback to reduce nuisance alarms and hence decrease operator overload especially during process upsets.

3.3 Proposed methodology

3.3.1 Down sampling to seconds

The initiation of an alarm can occur at various levels in a process plant starting from a low level smart instrument to the highest level DCS. Because of possible communication lags in the control system architecture, the time of initiation of a fault can be different from the time stamp that is attributed to the corresponding alarm message. These communication gaps have to be taken into account along with the material time lags when detecting related alarms. The precision of the time stamp for each alarm message varies from system to system and to analyze the data as a discrete event system, it is necessary to down sample the time stamps to a reasonable precision. In this work, alarm data is sampled at every second although there are no standard guidelines for selecting this value. If an alarm is annunciated more than once within a second, only one annunciation is considered and the rest are ignored.

3.3.2 Binary sequence representation of alarms

The idea is to represent the alarms in terms of zeros and ones. Where zero (0) represents no alarm or no information and one (1) represents an alarm annunciation. Each unique alarm, tagname.id is represented by a sequence of 0's and 1's sampled every second over a given period of time. Most part of the binary sequence is filled with 0's except for time instants when an alarm is presented to the operator. Alarm data for a unique alarm, LI300A.PVHI for a period of 12 minutes is shown in Table 3.1. The corresponding binary sequence representation for one second sampling is shown in Table 3.2.

Table 3.1: Sample alarm data for LI300A.PVHI

Alarm Time Stamp	Time Count
4/24/2010 12:00:47	1
4/24/2010 12:01:25	39
4/24/2010 12:01:39	53
4/24/2010 12:05:07	261
4/24/2010 12:06:07	321
4/24/2010 12:07:09	383
4/24/2010 12:11:43	657
4/24/2010 12:12:46	720

Table 3.2: Binary Sequence Representation of LI300A.PVHI

Element Number	Value	Element Number	Value
1	1	262-320	0
2	0	321	1
3-38	0	322-382	0
39	1	383	1
40-52	0	384-656	0
53	1	657	1
54-260	0	658-719	0
261	1	720	1

Table 3.2 shows that the length of the binary sequence is 720 for the unique alarm LI300A.PVHI over a 12 minute period. The sequence length increases as we consider larger time periods for analysis. Considering computational space constraints, these sequences can be equivalently represented by capturing just the indices where 1's occur in the sequence. A mathematical representation such as the one defined in Table 3.2 facilitates calculation of indices that indicate various characteristics of a unique alarm.

3.3.3 Graphical tools

Data visualization is the graphical display of abstract information for two purposes: for process analytics (also called data analysis) and communication [12]. Data visualization is important in analyzing and comprehending large volumes of data. It is useful in detecting patterns, clusters and outliers that are not obvious using non-graphical forms of representation. The human cognitive process should be considered while coming up with the best graphical representations. In developing graphical tools in this work, identification of nuisance alarms was the main goal.

Alarm messages are generally perceived as long strings of text with information hidden in it. The analysis of such raw data corresponding to a time range is not easy to conduct and comprehend. The next two sections demonstrate how the binary sequence representation facilitates appropriate plotting of alarm data for easy interpretation. Two plots and their specialities are presented using illustrative examples of real industrial alarm data obtained from an oil sands extraction plant. The plant takes in excavated raw oil sand which is sent into a crusher using several conveyer belts and is eventually mixed with caustic and hot water. Finally the slurry is sent into a separation cell. In this plant, several thousands of alarms are configured in its alarm system based on a variety of measured variables such as temperatures, flow rates, densities and levels.

3.4 High Density Alarm Plot

The High Density Alarm Plot (HDAP) is a useful tool for visualizing large amounts of alarm data of a plant over a selected time range. It gives an overview or big picture, of the alarm system without getting into details of

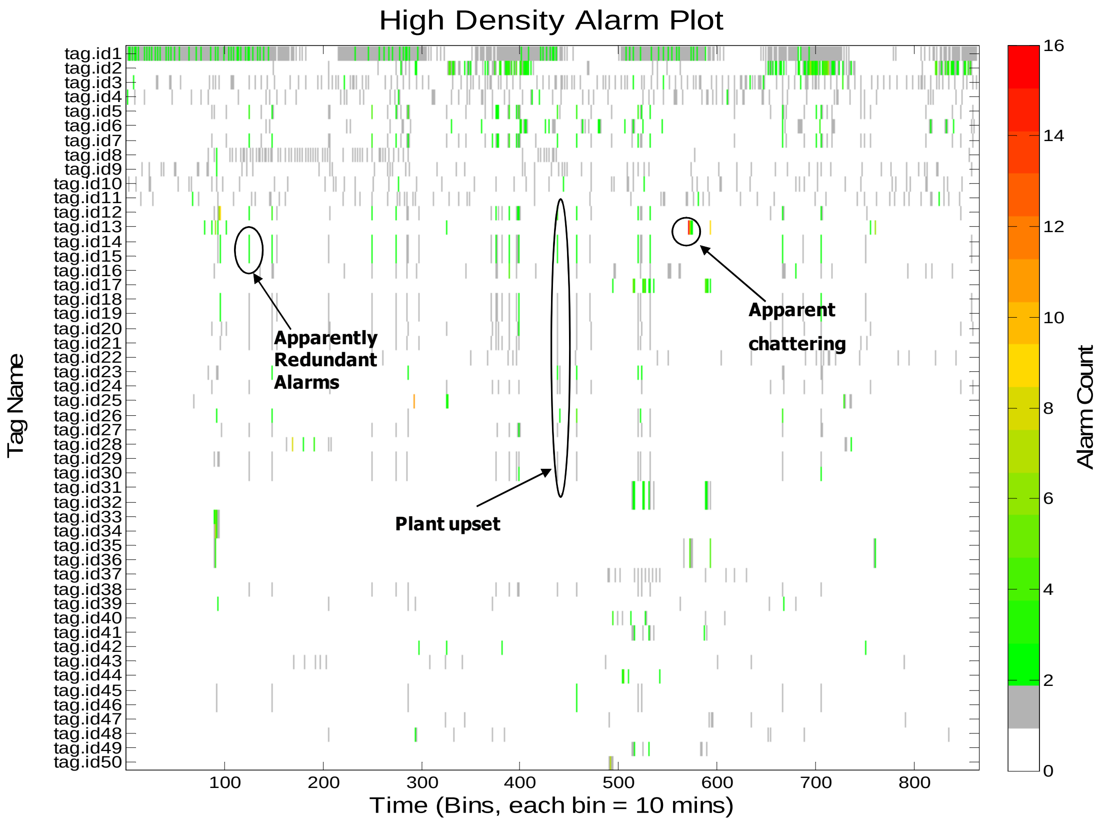


Figure 3.1: High Density Alarm Plot for the top 50 unique alarms over a one week period comprising over half a million observations showing nuisance alarms

each alarm message. Fig. 3.1 shows the HDAP of a real industrial alarm data set for top 50 alarms (commonly known as bad actors) based on the alarm count over a period of 6 days. The tag names and identifiers are masked due to confidentiality. HDAP basically shows the occurrences of top alarms (vertical axis) in the time domain (horizontal axis). Corresponding to each alarm, the alarm count in every ten minute interval (bin) is color coded. For this case study, the number of sample points considered in the binary sequence representation are $50(alarms) * 6(days) * 24(hours) * 60(minutes) * 60(seconds) = 25.92$ million.

Using HDAP, it is possible to visually identify periods of plant upset (instances where a lot of alarms are annunciated with in a short span of time), chattering alarms and related alarms. Properties and uses of the HDAP are discussed in more detail in the following subsections.

3.4.1 Each row corresponds to an alarm

Each row is a temporal representation of an alarm over the selected time range. This feature is useful in visualizing the time of occurrence of various alarms which facilitates identification of periods of plant upset. For example, the alarms as represented by, tag33.id and tag34.id appear only for a short period of time during this 6 day period (around the 100^{th} bin). Also, there are periods of plant upset which involve a subset of alarms (as represented by tag.id12, tag.id14, tag.id15, tag.id18, tag.id19, tag.id21 tag.id29 and tag.id30) occurring at the same time.

3.4.2 Color coding of alarm count in time intervals

For every alarm, the alarm count in each 10 minute interval is calculated and color coded for the same time range. The length of the interval can be customized. In this work, a 10 minute period is chosen to emphasize the alarm count recommendations in the engineering standards. According to EEMUA [3] and ISA [5] standards, during steady state operation of a plant, the operator should not receive more than one alarm over a 10 minute interval for efficient operation. By selecting an appropriate color scheme, the alarm count can be visualized with ease. From Fig. 3.1, it can be seen that the alarms as represented by, tag.id12, tag.id13, tag.id25 and tag.id28 show apparent chattering (The alarm count is more than 7 alarms in some bins).

3.4.3 Rank ordering of alarms

The alarms are ordered in such a way that the alarm count decreases as we move down. For example, over the given time period of 6 days, the alarm count on the alarm as represented by tag.id1 is greater than the alarm count on tag.id2 and so on. This kind of ordering is very useful in identifying redundant alarms because they tend to have a similar alarm count and thus appear together in the HDAP. For example, the alarms as represented by, tag.id14 and tag.id15 always appear together. In this case study, tag.id14 and tag.id15 are both density alarms generated on the same variable measured by a density probe. They have different tag names as one of the alarms as represented by, tag.id15 is sent to the DCS through a Programmable Logic Controller (PLC) program where as tag.id14 was sent directly to the DCS. After the analysis, the redundant alarm, tag.id15 was removed from the alarm system to avoid the redundancy. Likewise, other pairs of apparently redundant alarms are the ones as represented by tag.id18 and tag.id19, tag.id31 and tag.id32, tag.id35 and tag.id36 and tag.id45 and tag.id46.

3.5 Alarm Similarity Color Map (ASCM)

A procedure for carrying out event correlation analysis has been proposed in [9]. The method starts with calculating the maximum number of overlaps for various lags between a pair of events that are represented in a binary domain. The binary domain representation is based on event occurrence(s) in disjoint intervals of time. The authors suggest an interval length of at least five minutes. The authors make an assumption that the co-occurrences of two events follow a homogenous Poison process and calculate the probability that the number of overlaps is less than the maximum overlaps calculated for various lags. In calculating the co-occurrence rate parameter, the authors further assume that the two events are independent of each other. The method approximately estimates the relationship between three or more events (both alarms and operator actions) by means of similarities between event pairs.

In this work, a new similarity measure that takes advantage of the nature of the alarm data is proposed. Unlike the work presented in [9], no assumption is made in terms of approximating the behavior of alarm occurrences.

A similarity measure between two unique alarms can be defined based on their proximity of occurrence in the time domain. A simple solution to this problem is to calculate the fraction of instances when both the unique alarms have annunciated at the same time. In the binary sequence representation, this simply boils down to matching 1's between the two alarms. Once the similarity index is calculated between each and every pair of unique alarms as represented by, the obtained similarity matrix can be appropriately rearranged to highlight groups of related unique alarms.

Plotting the ASCM involves four steps. The first step is the padding of each unique alarm binary sequence with extra 1's to enrich the data. Second step is the calculation of a similarity index between the padded binary sequences corresponding to each and every pair of unique alarms. Third step is the rank ordering of unique alarms according to their degree of similarity with other unique alarms. Fourth and the final step is color coding of the re-ordered similarity matrix.

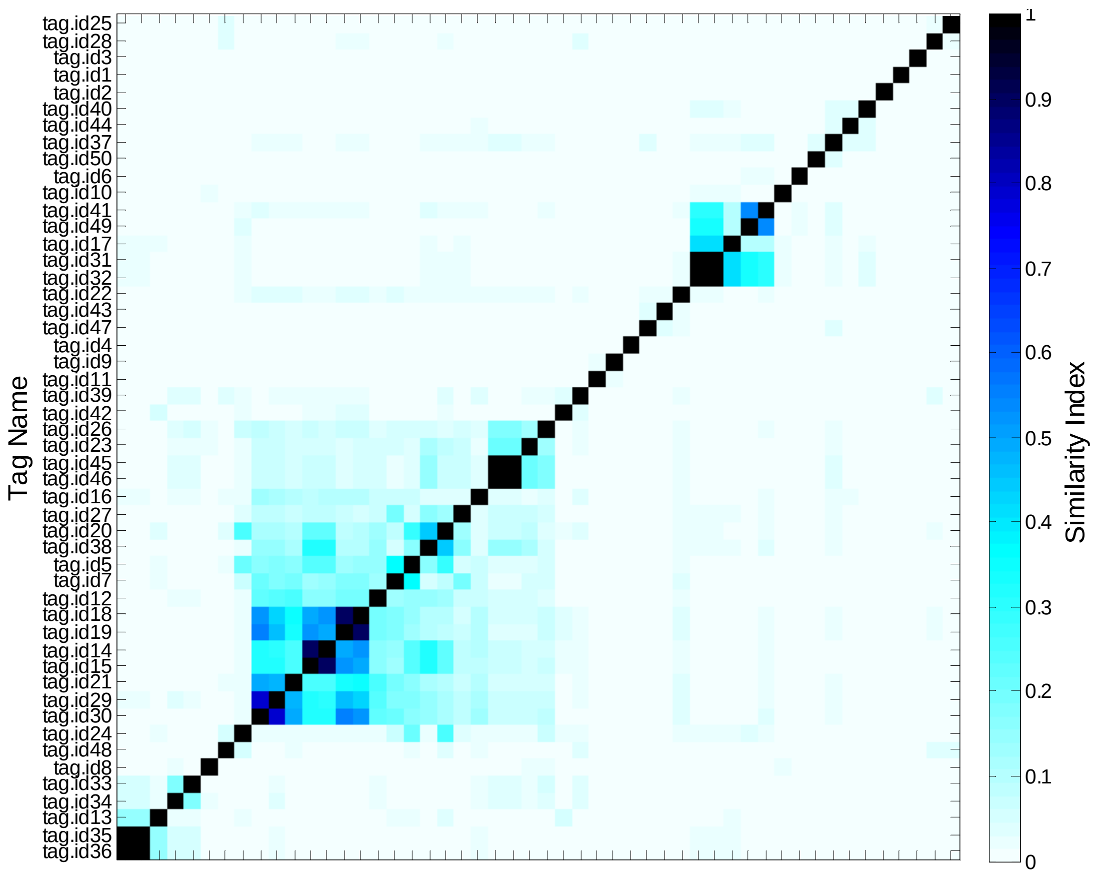


Figure 3.2: Alarm Similarity Color Map for the top 50 unique alarms showing related and redundant alarm clusters

3.5.1 Padding each binary sequence with extra 1's

To account for the possible communication lags and varying time delays between unique alarms, each binary sequence can be padded with extra 1's to enrich the existing alarm occurrences. In this work, for each occurrence of an alarm in a binary sequence, five 1's are added on either side of the actual occurrence, making a total of eleven 1's corresponding to each alarm occurrence. In other words, the span of influence of each alarm occurrence is increased to 11 seconds in the time domain. The padding length is a design parameter and should be adjusted with appropriate reasoning. This value should be in the range of the time taken by the operator to see and understand the alarm and differentiate it with other events. Once the operator understands the meaning of that alarm, he/she could take appropriate action to mitigate the process abnormality. If the padding length is increased drastically, it can be shown that all the alarms turn out to be redundant. It is worth noting that Return To Normal (RTN) state information is not used here to define the span of influence. However if RTN state information is used to define the span of influence, the interpretation of such similarity index will be different from the one defined in this work.

3.5.2 Alarm similarity index

A similarity index between two unique alarms in this case is expected to measure the strength of proximity of their occurrences. In the binary sequence representation of a unique alarm, the 0's have little meaning because they indicate that no message has been sent to the operator at those time instants for that unique alarm. Furthermore, the idea is to come up with a list of alarms that annunciate approximately at the same time and not in potential alarms that are dormant at the same time. Hence the samples at which both the unique alarms are not annunciated are discarded.

Informally, similarity measures are functions that quantify the extent to which objects resemble one another [8]. Considering the properties of the binary sequences, Jaccard similarity index [8] becomes an obvious choice for calculating the similarity between two unique alarms. The Jaccard similarity index, $S_{jac}(X,Y)$ between two padded alarm sequences, X and Y can be calculated by the following expression :

$$S_{jac}(X,Y) = \max_{l \in L}\left(\frac{a(l)}{a(l)+b(l)+c(l)}\right) \tag{3.1}$$

where $a(l)$ is the number of matches $(x_i = 1, y_{i+l} = 1)$
$b(l)$ is the number of mismatches $(x_i = 1, y_{i+l} = 0)$
$c(l)$ is the number of mismatches $(x_i = 0, y_{i+l} = 1)$

$$\forall\ i \in \begin{cases} [1\text{-}l,\text{N}] & l \leq 0 \\ [1,\text{N-}l] & l > 0 \end{cases}$$

l is the length of time lag between the sequences

X $(x_1, x_2....x_N)$ and Y $(y_1, y_2....y_N)$ are the padded binary sequence representations of the corresponding unique alarms. N is the length of any binary sequence which is same for all the unique alarms. L is the set of lags considered. In this work, $L = \{-240, .., 0, ..., 240\}$ which is a 4 minutes lag or lead.
The time delay, (d)between a pair of unique alarms is equal to the lag at which the similarity attains a maximum value. if $d < 0$, then X is likely to occur after Y occurs and if $d > 0$ then X is likely to occur before Y occurs.

Variations to the Jaccard similarity factor would allow different weights assigned for matches and mismatches. Positivity, symmetry, maximality, continuity and normality are the properties of the Jaccard similarity factor [8]. The Jaccard similarity factor lies between 0 and 1 and a high value of $S_{jac}(X, Y)$ indicates that the two alarms are closely related. For the top 50 unique alarms (bad actors), a 50 by 50 symmetric similarity matrix is obtained. The computational complexity in calculating the similarity matrix increases with the increase in the number of bad actors (n) and also with the number of lags considered (n_l) as

$$\text{computational complexity} = \mathcal{O}(n_l * n^2) \tag{3.2}$$

3.5.3 Re-ordering of the similarity matrix

For effective visualization of the similarity matrix, the rows and columns are rearranged according to the clustering order in hierarchial clustering combined with a heuristic for internal reordering. For hierarchial clustering, the distance measure between two unique alarms as represented by, X and Y is defined as $1 - S_{jac}(X, Y)$ and for comparing clusters with multiple alarms, the average linkage is adopted. The rearrangement may vary with the type of linkage (single linkage, complete linkage and so on) being used.

Clustering is not the same operation as linear ordering as hierarchial clustering only determines a tree structure. To obtain a linear order from the tree structure, either a heuristic or an optimization scheme has to be adopted. In this work, a heuristic based on alarm count is adopted so that unique alarms with smaller alarm count tend to appear on the same side of the final linear order. To compute the optimal leaf ordering, one that maximizes the similarity of adjacent objects was thought to be impractical [4] for cases where there are a large number of objects. A comparison between various leaf ordering heuristics can be carried out on the similarity matrix using existing visualization tools

Table 3.3: Clustering example that shows newly formed clusters and its constituents at each stage

New Cluster	Cluster 1	Cluster 2
8	2	5
9	3	4
10	6	8
11	1	7
12	9	11
13	10	12

such as the ones described in [14] and [1].

The internal tree reordering methodology adopted in this work is illustrated using an example comprising of 7 objects numbered 1 to 7. Assume a clustering order as shown in Figure 3.3 or the same is shown in Table 3.3. At each stage of the clustering algorithm, the newly formed cluster is numbered. Note that the cluster number in Cluster 1 is smaller than that in Cluster 2. If such an ordering is not adopted, the number of combinations of the final rearrangement would be $2^{(n-1)}$ where n is the total number of objects.

Once the clusters are obtained, a linear ordering is determined by decoding the clustering order as shown in Table 3.4. At each stage, the cluster with maximum number is decomposed into its constituents until only the basic objects (in this example, 1 to 7) are retained.

3.5.4 Color coding

The correlation matrix with rearranged rows and columns is color coded as shown in Fig. 3.2. It is to be noted that the tag names on the horizontal axis are omitted deliberately because they follow the same order as on the vertical axis. The apparently related alarms spotted earlier in the HDAP can

Table 3.4: Clustering example that shows the adopted leaf ordering heuristic. At each stage, the newest cluster is decomposed into its constituents

<table>
<tr><td colspan="7">13</td></tr>
<tr><td colspan="3">10</td><td colspan="4">12</td></tr>
<tr><td colspan="3">10</td><td colspan="2">9</td><td colspan="2">11</td></tr>
<tr><td colspan="3">10</td><td colspan="2">9</td><td>1</td><td>7</td></tr>
<tr><td>6</td><td colspan="2">8</td><td colspan="2">9</td><td>1</td><td>7</td></tr>
<tr><td>6</td><td colspan="2">8</td><td>3</td><td>4</td><td>1</td><td>7</td></tr>
<tr><td>6</td><td>2</td><td>5</td><td>3</td><td>4</td><td>1</td><td>7</td></tr>
</table>

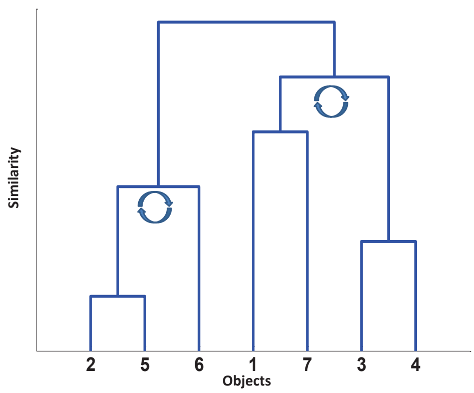


Figure 3.3: Dendrogram for the clustering example in Table 3.3 showing nodes that require flipping according to the adopted heuristic

be found to be related using the ASCM. Also, due to re-ordering, the alarms contributing to the plant upset form clusters in the ASCM. For example, in Fig. 3.2, the unique alarms as represented by tag.id14, tag.id15, tag.id18, tag.id19, tag.id21, tag.id29 and tag.id30 are part of a bigger cluster in the ASCM.

3.6 Second case study

In this section, alarm data for over a 10 day period from a crude oil refinery unit is analyzed using the tools discussed earlier.

3.6.1 HDAP for the second case study

HDAP for this case is shown in Fig.3.4. A modified legend is used to represent the color scheme in this case study. Grey color is used if there is just one alarm in a ten minute interval. Green color is used if the alarm count falls in between 2 and 5. Golden yellow color represents alarm counts that are between 6 and 10 alarms. For alarm counts of over 10, red color is used. Generally, flooding periods are defined as 10 minute intervals in which total alarm count exceeds 10. Thus, periods of alarm floods can be identified easily here. The apparently chattering alarms are the ones represented by tag.id1, tag.id2, tag.id3 and tag.id5. Apparently redundant alarms are the ones represented by tag.id25 and tag.1d26, tag.id37 and tag.id38, tag.id44 and tag.id45 and tag.id48 and tag.id49. There are two instances of plant upset. The first one occurs at around 240^{th} bin and the second one happens around the 640^{th} bin.

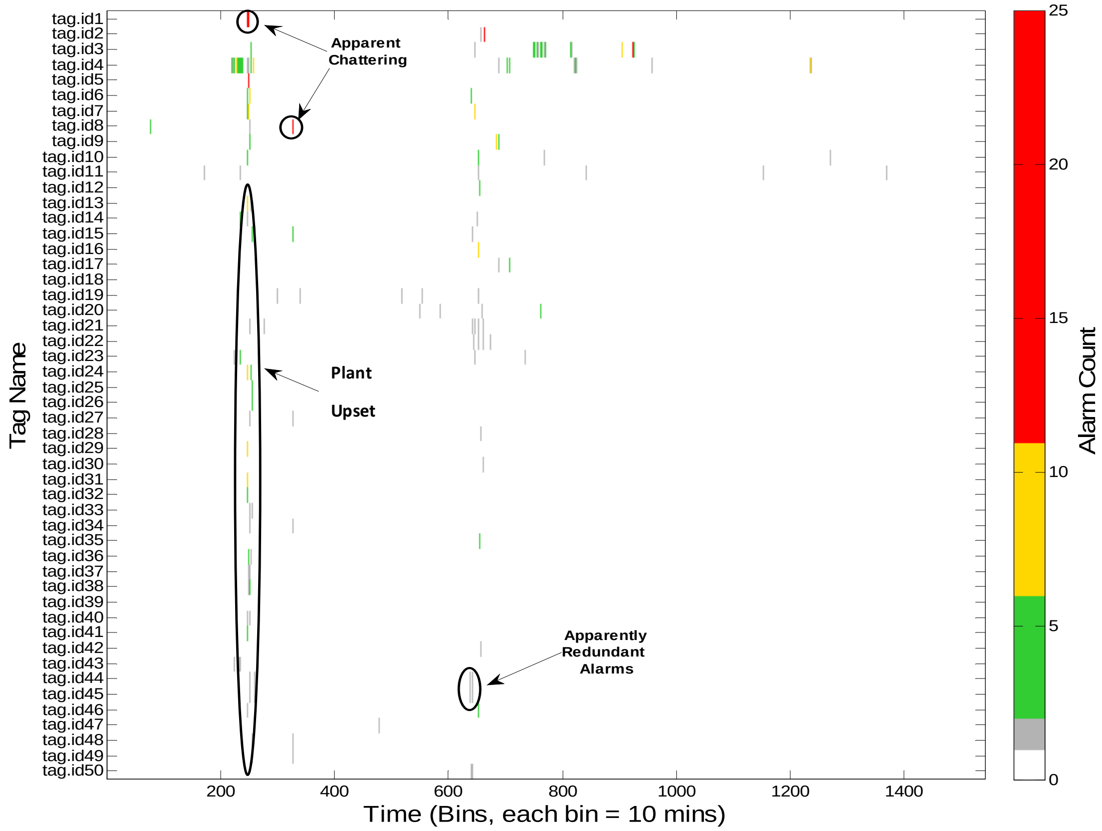


Figure 3.4: High Density Alarm Plot (Second case study) for top 50 unique alarms for a period of over 10 days showing nuisance alarms

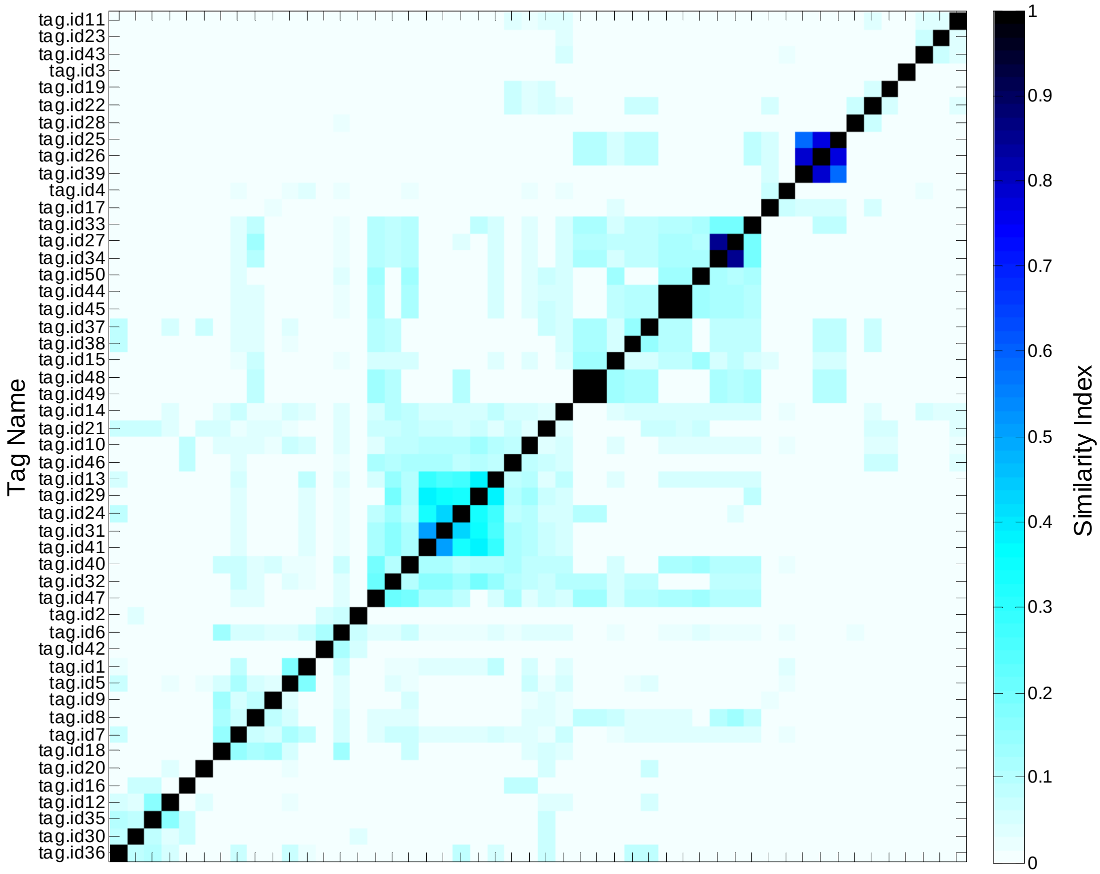


Figure 3.5: Alarm Similarity Color Map (second case study) for top 50 unique alarms for a period of over 10 days showing related and redundant alarm clusters

3.6.2 ASCM for the second case study

ASCM in Fig.3.5 clearly shows the redundant alarms which have a similarity factor of 1 between them. The unique alarms as represented by, tag.id44 and tag.id45 and tag.id48 and tag.id49 are redundant alarms. However, tag.id25 and tag.id26 show a lower value of the similarity factor and hence are not redundant although they appear to be redundant in the HDAP. The unique alarms as represented by tag.id25 and tag.id26 correspond to Hi and HiHi level alarms respectively in a hot separator drum. It just shows that the alarm limits for these two alarms are very close and the rate of change of the underlying process variable from Hi limit to HiHi limit is very fast. The Hi alarm limit is changed to that of the past HiHi alarm limit and the HiHi alarm is removed to reduce nuisance alarms.

Three clusters of related alarms can be seen from the ASCM. The first one involves the unique alarms as represented by, tag.id25, tag.id26 and tag.id39. The second cluster involves the unique alarms as represented by, tag.id33, tag.id27, tag.id34, tag.id50, tag.id44 and tag.id45. The last cluster involves unique alarms as represented by, tag.id13, tag.id29, tag.id24, tag.id31 and tag.id41. All the unique alarms in each of these three clusters are confirmed to be related (belongs to the same sub unit) through process knowledge.

3.7 Discussion

The utility of the proposed graphical tools have been demonstrated using two industrial data sets. Each alarm in these case studies has over half a million observations in its binary sequence representation. Both case studies have highlighted the practicality of the developed tools in terms of their ability to identify nuisance alarms with ease. The analysis using HDAP is sensitive to factors such as the analysis time period that is considered, color scheme and bin size. In this work, a bin size of 10 minutes is chosen to emphasize the standard recommendations on acceptable and maximum manageable alarm annunciation rate ([3] & [5]). The color scheme can be varied to emphasize the chattering behavior of individual alarms. For example, in the second case study, all the instances where there are more than 10 alarms in a 10 minute bin are colored red to emphasize unacceptable chattering behavior. The time period must be selected in such a way that the individual bins are clearly visible. The ASCM depicts rearranged alarms to show clusters that have similarity in time of occurrence. The way ASCM is plotted depends on factors such as the number of top alarms that are considered, analysis time period, padding length, type of linkage in building the clusters and the type of leaf ordering method chosen.

Once the chattering and related alarms are identified, appropriate measures have to be taken to reduce their occurrence. Alarm rationalization needs to be carried out for these bad actors and advanced signal processing techniques or multivariate alarming techniques [7] can be employed to minimize them. In [6], the use of various filters, dead bands and time delays in eliminating nuisance alarms is discussed in a unified framework. Enhanced alarming methods can also be used to reduce the nuisance alarms without changing the basic alarming algorithm. For example, knowledge based techniques that involve pattern recognition can be used to raise a multivariate alarm that indicates a predefined faulty condition [10].

Both HDAP and ASCM have been used as weekly alarm system assessment tools on the plant discussed in the first case study over a period of four months. The tools proposed in this work were successful in readily identifying chattering alarms and related alarms. Several changes have been made to the alarm system to minimize nuisance alarms. Techniques such as filtering of underlying process data, deadbands and delay timers were used to reduce chattering alarms. Redundant alarms were identified and removed. At the end of the four month period, the weekly alarm count was reduced by over 90%. With these new alarm settings, the plant has been operating closer to the EEMUA recommensations on maximum manageable alarm annunciation rate [3].

3.8 Concluding remarks

This perspective tutorial highlights some of the challenges in the performance assessment stage of the alarm management lifecycle. Many industries shy away from carrying out this exercise due to the complexity involved in analyzing alarm event data. Binary sequence representation of alarm data proposed in this work facilitates advanced analysis of alarm events. Two graphical tools that are specially designed to efficiently identify nuisance alarms are presented and their characteristics are discussed using two industrial datasets. The three dimensional HDAP encapsulates the information from alarm data for a given period of time. HDAP not only shows the progression of top alarms with time but also highlights apparently redundant and chattering alarms. A similarity measure is proposed for this application and justified through its physical interpretation. ASCM shows the similarity measure between rearranged unique alarms in a color coded matrix format and is useful for identifying groups of related alarms providing insights into process interactions. These two graphical representations of the alarm data provide quick and valuable feedback to make improvements to the alarm system in several other steps in the alarm management lifecycle and contribute to reduction in nuisance alarms.

Bibliography

[1] G. Caraux and S. Pinloche, *Permutmatrix: a graphical environment to arrange gene expression profiles in optimal linear order*, Bioinformatics **21** (2005), no. 7, 1280–1281.

[2] CCPS/AIChE, *Guidelines for engineering design for process safety*, Wiley, New York: Center for Chemical Process Safety/American Institute of Chemical Engineers, 1993.

[3] EEMUA, *Alarm systems: A guide to design, management and procurement*, 2 ed., EEMUA Publication No. 191 Engineering Equipment and Materials Users Association, London, 2007.

[4] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein, *Cluster analysis and display of genome-wide expression patterns*, Proc. Natl. Acad. Sci. USA **95** (1998), 14863–14868.

[5] ISA, *Management of alarm systems for the process industries*, Tech. Report ANSI/ISA-18.2-2009, International Society of Automation, ISA, 67 Alexander Drive, P.O. Box 12277, Research Triangle Park, North Carolina 27709, 2009.

[6] I. Izadi, S. L. Shah, D.S. Shook, S. R. Kondaveeti, and T. Chen, *A framework for optimal design of alarm systems*, In proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, 2009, pp. 651–656.

[7] S. R. Kondaveeti, S. L. Shah, and I. Izadi, *Application of multivariate statistics for efficient alarm generation*, In proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, 2009, pp. 657–662.

[8] M-J. Lesot, M. Rifqi, and H. Benhadda, *Similarity measures for binary and numerical data: a survey*, Int. J. Knowledge Engineering and Soft Data Paradigms **1** (2009), no. 1, 63–84.

[9] J. Nishiguchi and T. Takai, *Ipl2 and 3 performance improvement method for process safety using event correlation analysis*, Computers & Chemical Engineering **34** (2010), no. 12, 2007–2013.

[10] D. H. Rothenberg, *Alarm management for process control*, Momentum Press, New Jersey, August 2009.

[11] N. A. Stanton, *Human factors in alarm design*, Taylor & Francis, Inc., Bristol, PA, USA, 1994.

[12] F. Stephen, *Encyclopedia of human-computer interaction*, ch. Data Visualization for Human Perception, 2012.

[13] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri, *A review of process fault detection and diagnosis part i: Quantitative model-based methods*, Computers & Chemical Engineering **27** (2003), no. 3, 293–311.

[14] H.-M. Wu, Y.-J. Tien, and C.-H Chen, *Gap: A graphical environment for matrix visualization and cluster analysis*, Computational Statistics and Data Analysis **54** (2010), no. 3, 767–778.

4

Quantification of alarm chatter based on run length distributions

4.1 Overview

This chapter deals with chattering alarms and provides a means to quantify alarm chatter through run-length distributions[1,2]. Due to improper design of alarm generating logic or lack of tuning, alarms are announced more frequently than what is typically sufficient to alert the operator, a condition commonly known as 'alarm chatter'. The concept of run length is introduced in the alarm management context and an index is proposed to quantify the degree of alarm chatter based on run length distributions obtained from historical

[1]A shorter version of this chapter has been published in Proceedings of the 49^{th} IEEE Conference on Decision and Control, December 15-17, 2010, Atlanta, Georgia, USA.

[2]Sections of this chapter have been accepted for publication in Chemical Engineering Research and Design

alarm data. Prominent features of the proposed chatter index and its variant are demonstrated using industrial datasets.

4.2 Introduction

The purpose of an alarm system is to alert the control room operator when the process shifts towards unsafe or low quality production. Now-a-days, due to the ease in implementing alarms, the volume of process and system variables that have alarms configured on them has risen exponentially. In a typical plant, most of these alarms are configured during the design and commissioning phase when there is limited knowledge of the nature of the variable being measured and monitored. The two main aspects of alarm design are the selection of alarm generating algorithm and tuning. The alarm generating algorithm or fault detection algorithm can be as simple as a difference between a raw process variable and a fixed alarm limit to as complicated as using machine learning tools for fault classification and are reviewed by [17]. Tuning involves use of simple techniques such as deadbands (also known as hysteresis where a different value from alarm limit is used to clear an alarm), on-delay and off-delay timers and so on. For example, on analog measurements, some control systems have deadband (also known as alarm hysteresis) of 0.5% of instrument range as default tuning parameter for the alarms configured on it. Depending on characteristics like the measurement noise in the signal and type of variable being monitored, a higher value of deadband or another type of tuning such as delay timers [10] may be required for efficient alarm annunciation.

In the presence of inefficient alarm design, the rate at which alarms are presented to the control room operator during abnormal events tends to be much higher than what he/she can comprehend and respond to. Most of the alarms during these alarm floods are a nuisance to the operator as they limit the operator's ability to identify the root cause variables or critical alarms.

In the recent past, there has been a significant interest in the field of alarm management in the process industries. There are several incidents following which the investigation reports have pointed at ineffective alarm systems as one of the major drawbacks [4]. Several standards ([7] and [5] to name a few) have been published with pointers to effectively managing an alarm system. [16] has provided a summary of the problems and best practices for managing an efficient alarm system.

There are several stages involved in the *alarm management life cycle* as described in [7]. To make improvements to an already existing alarm system, the *monitoring and assessment stage* is a good entry point into the life cycle. The problems identified in this stage can be rectified in several other stages in the life cycle such as the rationalization and design stage. Identification of nuisance alarms is one of the main objectives in this stage. Chattering alarms are the most common form of nuisance alarms as they fail to provide sufficient time for the operator to respond to each and every occurrence of that alarm. Essentially, chattering alarms conflict with the basic philosophy of each alarm being actionable. Eliminating chattering alarms would improve the quality of alarm data. Good quality alarm data is a prerequisite for advanced alarm correlation analysis like the ones performed by [11], [19] and [2] or the dynamic risk analysis performed by [15] to improve process safety and product quality.

In the academic or engineering practice literature, there are neither standard procedures for identifying chattering alarms nor are there any measures to quantify alarm chatter. The process of rationalizing each alarm tag configured on the alarm system and implementing appropriate design is very time consuming. A feasible approach would be to identify chattering alarms using routinely collected alarm data. Once the chattering alarms are identified along with the amount and nature of chattering, one of the standard design changes can be implemented to reduce the amount of chatter [8]. Moreover, such an index for alarm chatter would help in optimal design of a suitable filter using only the alarm data. Reducing alarm chatter using adaptive dead bands based on time series modeling of the process data has been proposed in [6]. However, this approach requires identification of chattering alarms as a first step and then collection of high frequency process data for modeling and design of adaptive deadbands.

This chapter is organized as follows. Section 4.3 introduces the problem and causes of alarm chattering. Section 4.4 introduces the concept of runs and run lengths and briefly discusses their application in the alarm monitoring context. In section 4.5, a means to quantify alarm chatter based on run length distributions is defined and illustrated with appropriate industrial case studies in section 4.6. Section 4.7 discusses the improvements achieved in the chatter index due to design changes made on two industrial alarm case studies.

4.3 What is a chattering alarm?

A *chattering alarm* is a unique alarm that is activated and cleared excessively within a short span of time (similar definitions are presented in [5], [7] and [16]). As a rule of thumb, an alarm that repeats three or more times in one minute is often used as a first pass identification of the worst chattering alarms [7]. It is evident that chattering is very vaguely defined and there are no standard guidelines to calculate the degree of chatter on a unique alarm. As a key performance indicator, there is no acceptable quantity of chattering alarms. Therefore, all the chattering alarms should be eliminated as part of a good alarm management process.

In [16], chattering alarms are defined to be generated only by digital signals whereas *repeating alarms* which are very similar to chattering alarms are normally caused by analog signals. In this work, no such distinction is made and both of them are referred to as chattering alarms. There are several causes for an alarm to chatter. For alarms configured on analog signals with inefficient design, chattering occurs when the process is operating close to alarm limits. Due to the presence of process and measurement noise, the analog signal tends to cross the alarm limit frequently. Improper alarm design such as inappropriate use of delay timers and latches is often the main cause for alarm chatter on most digital signals.

4.4 Runs and run lengths

In probability and statistics, a run of a certain type of element is defined as an uninterrupted sequence of one or more identical elements that are preceded and followed by other types of elements or no elements at all. Run length can be defined as the number of elements in a run [12].

4.4.1 Brief history of the use of runs and run lengths

Runs and run-lengths are useful in many fields, including statistical process control, reliability, bio-informatics and finance, for compression and analysis of several forms of data. The interpretation of run lengths is based on the application context. For example, in the statistical process control terminology, *Average Run Length* (ARL) is defined as the average time for which a process remains within some specified control limits. ARL is very useful in evaluating the performance of various process control charts (Shewhart, CUSUM, EWMA and their variants) [14]. In the Wald-Wolfowitz test (also known as the *runs* *test*) for randomness, negative and positive runs are defined based on whether the elements are above or below a limit [18]. Run lengths are similarly defined for the error signal (difference between Set Point (SP)and Controlled Variable (CV)) for evaluating the performance of process controllers [13]. In computer science and information theory, run length encoding is extensively used as a form of data compression.

Statistics based on run lengths provide a reasonable criterion and constitute an evidence for the underlying process. Analysis of the run lengths depend on the nature of the application. For example, the *runs test* for randomness is a different kind of analysis compared to ARL in statistical quality control. The analysis methodology depends on how runs are defined for a particular application [3].

4.4.2 Run lengths in the alarm context

As mentioned in [9], the most important parts of an alarm message are the time stamp (up to seconds precision), tag name (usually has information about the instrument number, plant name and variable type) and alarm identifier (PVLO, PVHI, TRIP, etc.). All these three fields are required to uniquely identify an alarm. It is assumed that each unique alarm (tagname.identifier) requires a unique operator action. In [9], it has been shown that industrial alarm data can be mathematically represented using binary sequences. In this binary time series representation, a value of **1** indicates that an alarm is annunciated to the operator at that time instant whereas **0** indicates no alarm is annunciated. This way of binary sequence representation would capture only the instants when an alarm is sent to the operator and not when it is standing over a period of time. Hence the analysis performed on this type of data is more operators centric and need not include behavior of the process itself.

Definition

A *run* in the alarm monitoring context is defined intuitively as the sequence of a 1 followed by uninterrupted 0's before another 1 is encountered in the binary sequence representation of a unique alarm. The length of a sequence is called the *run length*. Thus a run length can be perceived as the time difference in seconds between two consecutive unique alarms on the same tag. These two alarms may be due to a single abnormal event or two different abnormal events. During this period, no assumption is made as to whether the operator takes an action to mitigate that abnormal event or not. If an alarm appears, clears and reappears within an interval of one second, the run length is assumed to be 1 second. This limitation is due to the one second sampling for alarm data which is assumed to be quite reasonable for industrial alarm systems. However, this assumption is violated by controllers that generate events (messages) with higher execution frequency. If the alarm activates, clears and reactivates within the period of least count of the run-length, the runlegth for that alarm can be approximated to be equal to the corresponding least count.

Illustrative example on alarm run lengths

The second column in Table 4.1 shows the time instants at which a fictitious level alarm as represented by LI300B.PVHI is announced. The third column in the same table shows the time count in seconds from the start of the first alarm. A time trend for this unique alarm is shown in figure 4.1. The fourth column shows the run lengths for this unique alarm. It can be seen that the minimum run length is 2 seconds for this example and it occurs for the 7th alarm. It means that the process had returned to normal and re-exceeded the

Table 4.1: Run Length for a fictitious alarm as represented by LI300B.PVHI based on historical data

S.No:	Alarm Time Stamp	Time Count	Time difference (Run Length, r)
1	4/24/2010 12:00:01	1	3
2	4/24/2010 12:00:04	4	3
3	4/24/2010 12:00:07	7	5
4	4/24/2010 12:00:12	12	7
5	4/24/2010 12:00:19	19	7
6	4/24/2010 12:00:26	26	7
7	4/24/2010 12:00:33	33	2
8	4/24/2010 12:00:35	35	5
9	4/24/2010 12:00:40	40	7
10	4/24/2010 12:00:47	47	15
11	4/24/2010 12:01:02	62	-

alarm limits within those 2 seconds. This duration is too short for an operator to take appropriate action.

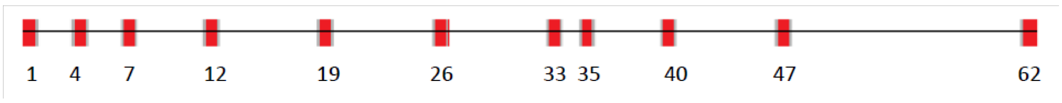


Figure 4.1: Time trend showing alarm annunciations and the respective time count for the alarm represented by LI300B.PVHI

4.4.3 Run Length Distribution

A *run length distribution* (RLD) can be built by summing up and grouping the number of times various run-lengths appear. It is basically a histogram of the run lengths. Figure 4.2 shows the RLD for the alarm represented by LI300B.PVHI. The vertical axis represents the frequency or the alarm count (n_r) and the horizontal axis is the run length (r). RLD based on a large amount

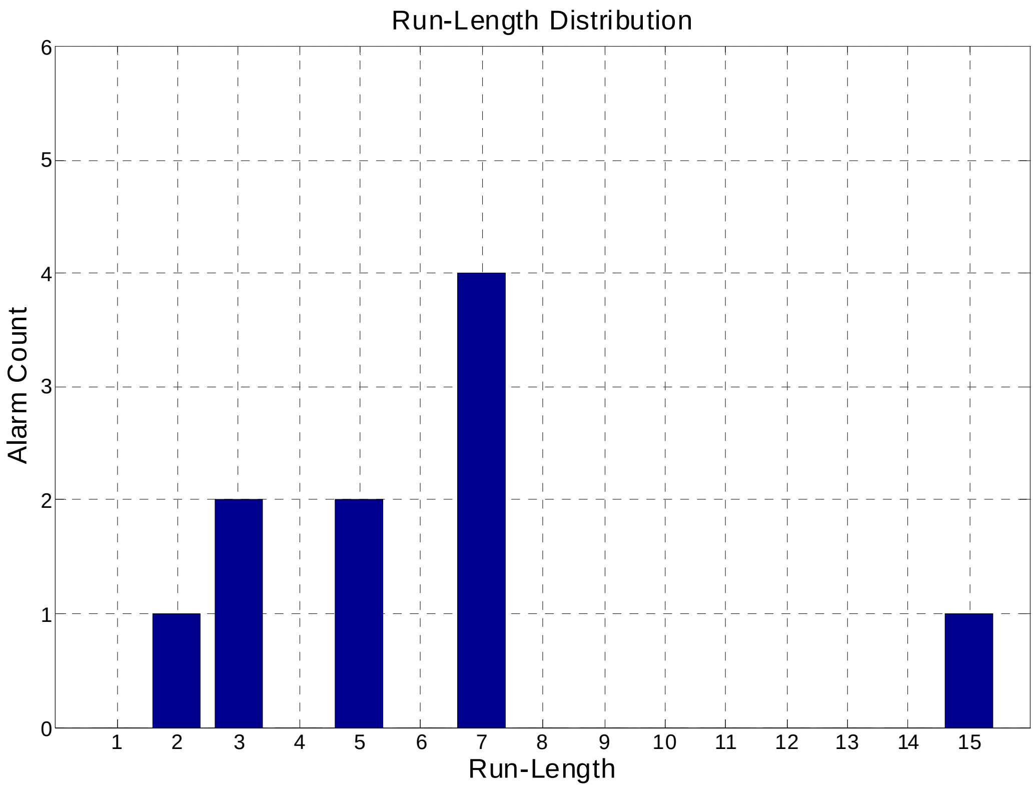


Figure 4.2: Run Length Distribution for the fictitious alarm represented as by LI300B.PVHI

of alarm data will reveal reliable statistics about the behavior of the alarm. For example, for a unique alarm that resets once every 10 seconds during an abnormal event, the RLD will display a peak at a run length, $r = 10$ seconds.

4.5 Chatter index based on the run length distribution

4.5.1 Differences between chattering and non-chattering alarms

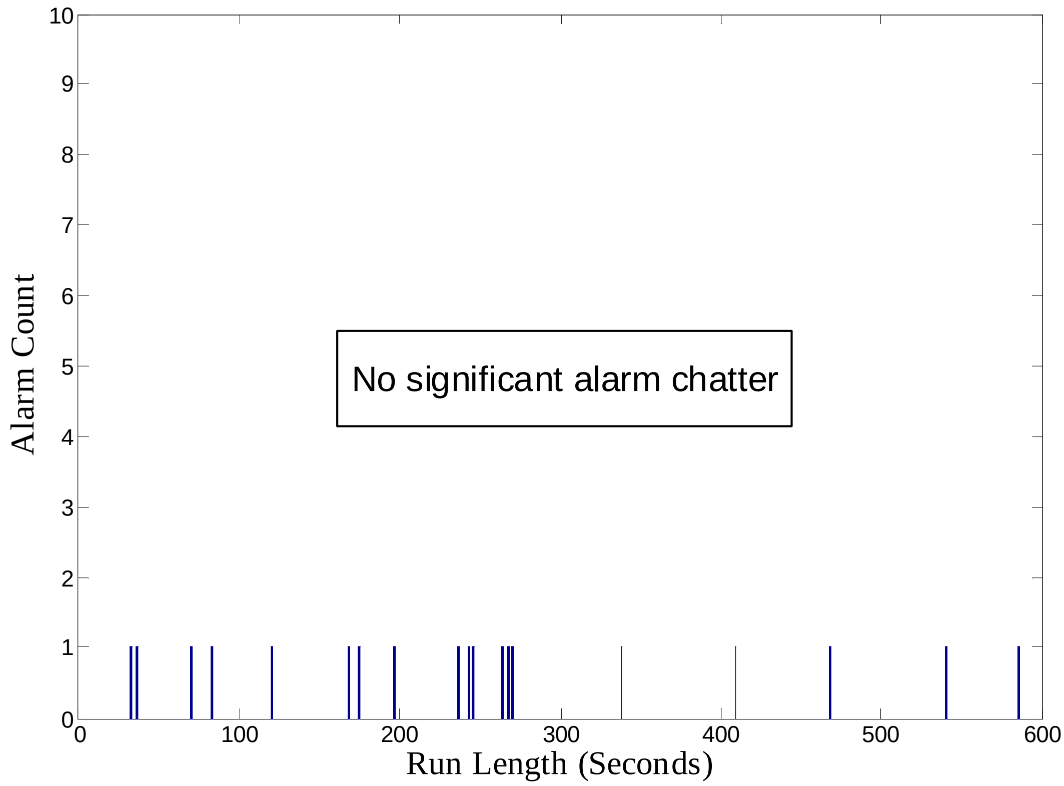


Figure 4.3: Run length distribution for a non-chattering tag

Figures 4.3 and 4.4 show the RLDs of two unique alarms based on real industrial alarm data over a period of one week. From figure 4.3, it is clear that there are not many alarms with short run lengths. The minimum run length observed is 33 seconds and the distribution is fairly uniform with just one alarm count for each existing run length.

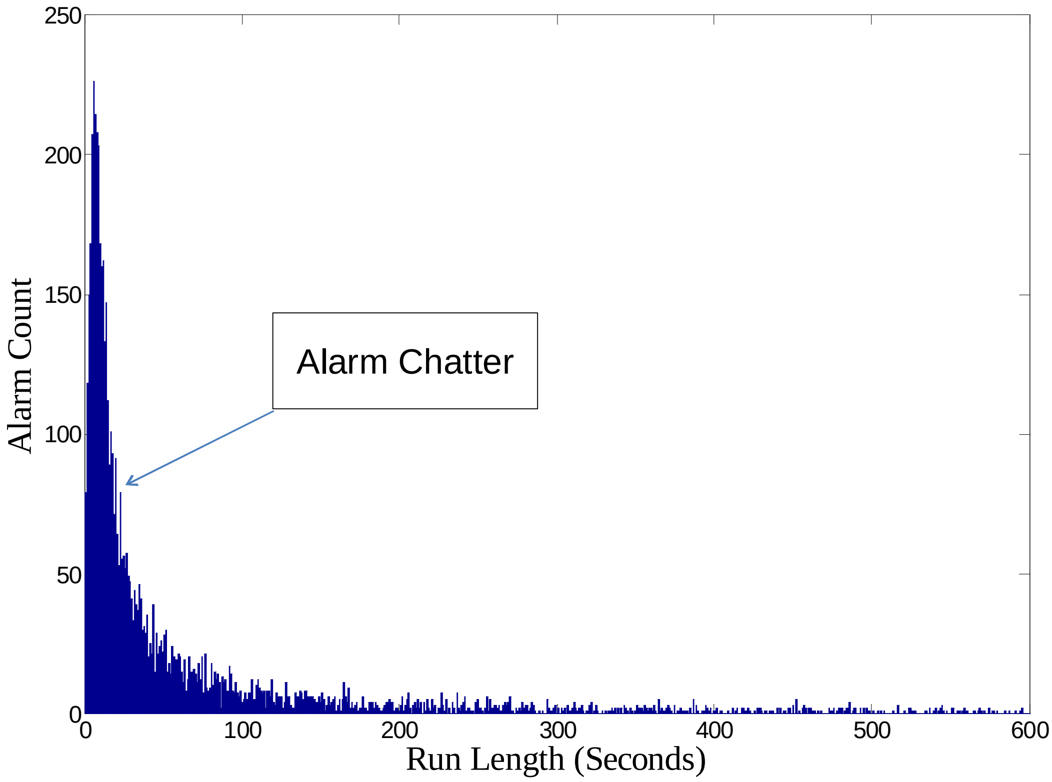


Figure 4.4: Run length distribution for a chattering tag

Figure 4.4 shows a highly skewed RLD for another unique alarm. Significant alarm counts exist for run lengths as short as 1 second. This is clearly a heavily chattering alarm. The idea here is to measure the extent of alarm chatter based on these differences in the RLDs.

4.5.2 An index to measure alarm chatter

To calculate a chatter index based on the RLD, it is important to have sufficient data that can represent the behavior of the alarm tag. There are no standard guidelines on the amount of data needed for this analysis. The confidence in the calculation increases with the volume of data available. Once RLD for an unique alarm is obtained, it can then be normalized to obtain the Discrete Probability Function (DPF). It can be shown that a DPF can be obtained from the RLD by normalizing it with a factor $\sum_{r\in\mathbb{N}} n_r$ which is one less than the total number of alarms on the unique alarm during the considered time period. This is because the last alarm does not have a run length.

$$P_r = \frac{n_r}{\sum_{r\in\mathbb{N}} n_r}, \ \forall \ \ r \in \mathbb{N} \tag{4.1}$$

where P_r represents the probability and n_r represents the alarm count for any run length r.

The chatter index is then defined by choosing an appropriate weighting function that emphasizes alarm counts with short run lengths. For this purpose, the DPF is weighted with a function whose value decreases with increasing run length. Once the weighting function is chosen, a chatter index can be calculated as

$$\text{Chatter index} = \sum_{r\in\mathbb{N}} P_r w_r \tag{4.2}$$

4.5.3 Chatter index based on inverse weighting of the DPF

Run length is the time in seconds between two consecutive alarms on the same tag. The inverse of the run length would be the instantaneous frequency of occurrence of the alarm. If the inverse of the run length is used as the weighting function, the proposed Chatter Index (Ψ) can be written as:

$$\Psi = \sum_{r \in \mathbb{N}} P_r \frac{1}{r} \tag{4.3}$$

A.2 shows how Ψ is calculated using an example of a fictitious alarm. In spite of the fact that Ψ does not uniquely determine the corresponding DPF, Ψ is useful as a good measure to capture the skewness in a RLD towards shorter alarm run lengths.

4.5.4 Properties of the proposed chatter index, Ψ

Listed below are the properties of the proposed chatter index.

Theoretical bounds on Ψ

As shown in A.1, Ψ can take values between and including 0 and 1. The higher the chatter index, the higher the amount of alarm chatter. It is easy to deduce that an alarm tag can have $\Psi = 1$ only when there are alarms appearing every second without interruption (i.e $P_r = 1$ for $r = 1$). And Ψ can take a value 0 when there are less than 2 alarms on the tag during the same period.

Physical interpretation of Ψ

The chattering metric, Ψ of a unique alarm can be perceived as the mean frequency of annunciation of that alarm assuming that the abnormal event prevails for an indefinite period of time. Units of Ψ are alarms/second.

Ψ is independent

No tuning parameter is required to calculate Ψ. Once we have the alarm data corresponding to a unique alarm over a certain time period, calculation of Ψ is straightforward.

A rule of thumb for the cut off on Ψ

Although there are no standard procedures to identify chattering alarms, in [7], it has been mentioned that a frequency of 3 or more alarms per minute can be used as a rule of thumb to identify the worst chattering alarms. Thus a reasonable cutoff on Ψ to identify worst chattering alarms is $\Psi_{cutoff} = \frac{3}{60} =$ **0.05** alarms/second.

4.5.5 Scope for a modified chatter index

In defining Ψ, it is assumed that the abnormal event prevails for an indefinite period of time. However, in the following section, it will be shown that the assumption is reasonable due to the fact that large run lengths contribute insignificantly towards the calculation of Ψ.

If we were to know that for a specific abnormal event, nuisance alarms (in the form of chatter) following an actual alarm will not last for more than a specified duration (say τ seconds), all the run lengths greater than τ can be ignored. DPF can then be modified according to a truncated RLD (truncated to τ seconds). The normalizing factor will then be $\sum_{r=1}^{\tau} n_r$. The DPF can be defined as

$$P_{r,\tau} = \begin{cases} \dfrac{n_r}{\sum_{r=1}^{r=\tau} n_r}, & \forall \;\; r \in \{1, 2, 3...\tau\} \\ 0, & \forall \;\; r \in \{\tau+1, \tau+2, \tau+3...\infty\} \end{cases} \tag{4.4}$$

The modified chatter index, Ψ_τ can be written as

$$\Psi_\tau = \sum_{r \in \mathbb{N}} P_{r,\tau} \frac{1}{r} \tag{4.5}$$

For the fictitious example considered in A.2, if we were to know that the alarms corresponding to a single abnormal event will not be separated by more than 10 seconds, $\Psi_{10} = \dfrac{(60-1)*3*7}{(60-1)*3*7} * \dfrac{1}{10} = 0.1$, which is strictly equal to the frequency of occurrence (1 alarm in 10 seconds). Further, for real industrial alarm data, the best chatter index can be obtained by calculating Ψ_τ over a reasonable range of values of τ and picking the best one (similar to picking the top best factors from a scree plot in Principal Components Analysis). It can be shown that Ψ_τ has the same bounds as Ψ. Additionally, these chatter indices, Ψ and Ψ_τ, can be multiplied by a factor of 60 to represent the frequency of alarm occurrence per minute instead of a second.

4.6 Industrial case study

In this section, alarm data from an oil sands extraction plant is analyzed for chattering alarms. For convenience, only four unique alarms of interest are shown in this work.

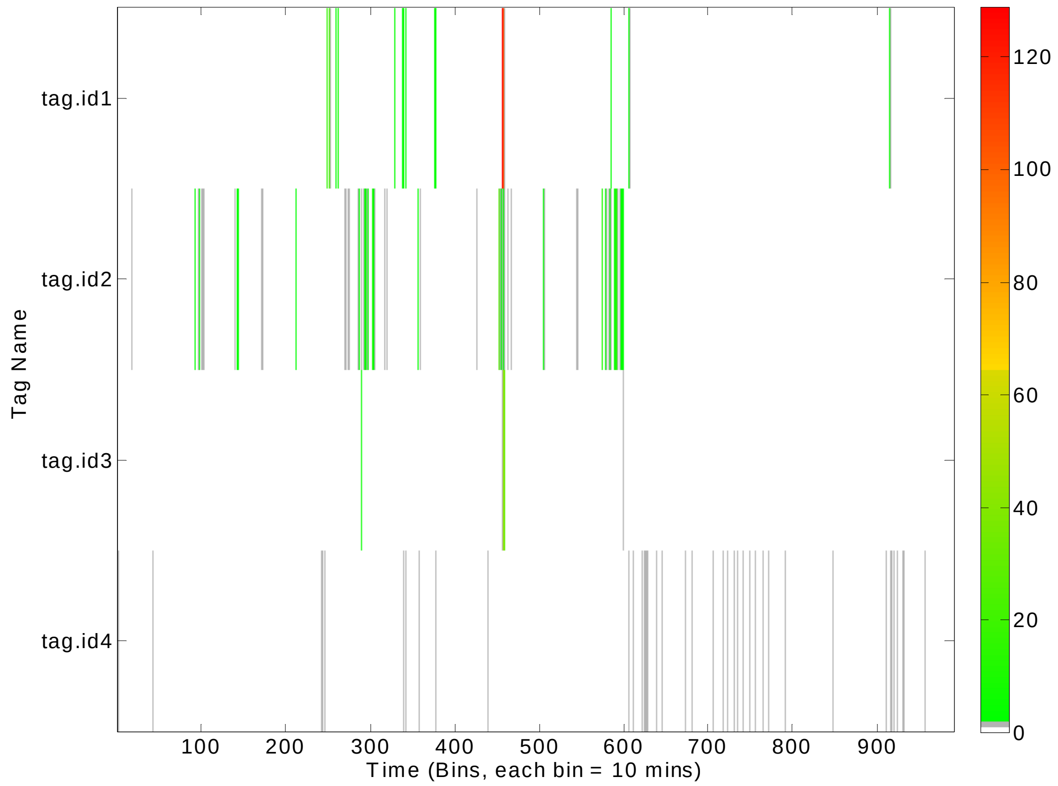


Figure 4.5: High Density Alarm Plot for 4 Alarm Tags

4.6.1 High Density Alarm Plot

The High Density Alarm Plot (HDAP) is useful for visualizing large amounts of alarm data of a plant over a selected time range [9]. For every unique alarm that represents a row in the HDAP, the alarm count in each 10 minute interval is calculated and color coded. Using HDAP, it is possible to visually identify chattering alarms, related alarms and periods of plant instability. HDAP for just four tags of interest of a plant over a period of one week is presented in figure 4.5. The tag names and identifiers are masked due to confidentiality.

The unique alarm as represented by tag.id1 shows significant chatter at around 450^{th} bin. There are over 120 alarms over a 10 minute interval during that period. During the same time period, there are about 60 alarms in a 10 min interval on the alarm as represented by tag.id3. The unique alarm as represented by tag.id2 has relatively less chattering but the number of alarms raised during the considered time period is quite high. The number of alarms on the unique alarms represented by tag.id1, tag.id2, tag.id3 and tag.id4 during the one week period are 332, 190, 91 and 59 respectively.

4.6.2 RLDs

Figure 4.6 shows the truncated RLD for the unique alarm as represented by tag.id1. The RLD is skewed towards shorter run lengths and a large number of alarms (about 80) have run lengths as short as 1 second. Therefore the unique alarm as represented by tag.id1 is expected to have a higher chatter index compared to the others.

For the unique alarm as represented by tag.id2, we can see from figure 4.7 that there is just one alarm with the shortest run length of 5 seconds. However, there are a good number of alarms with run lengths shorter than 100 seconds.

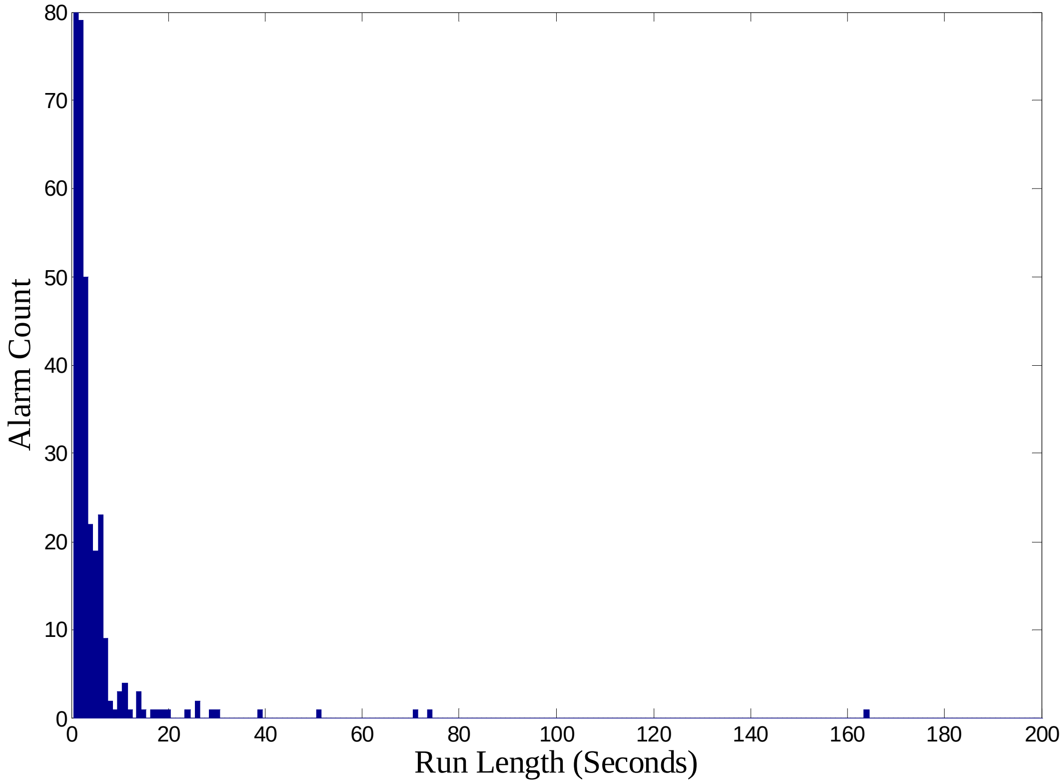


Figure 4.6: Run length distribution for the unique alarm as represented by tag.id1

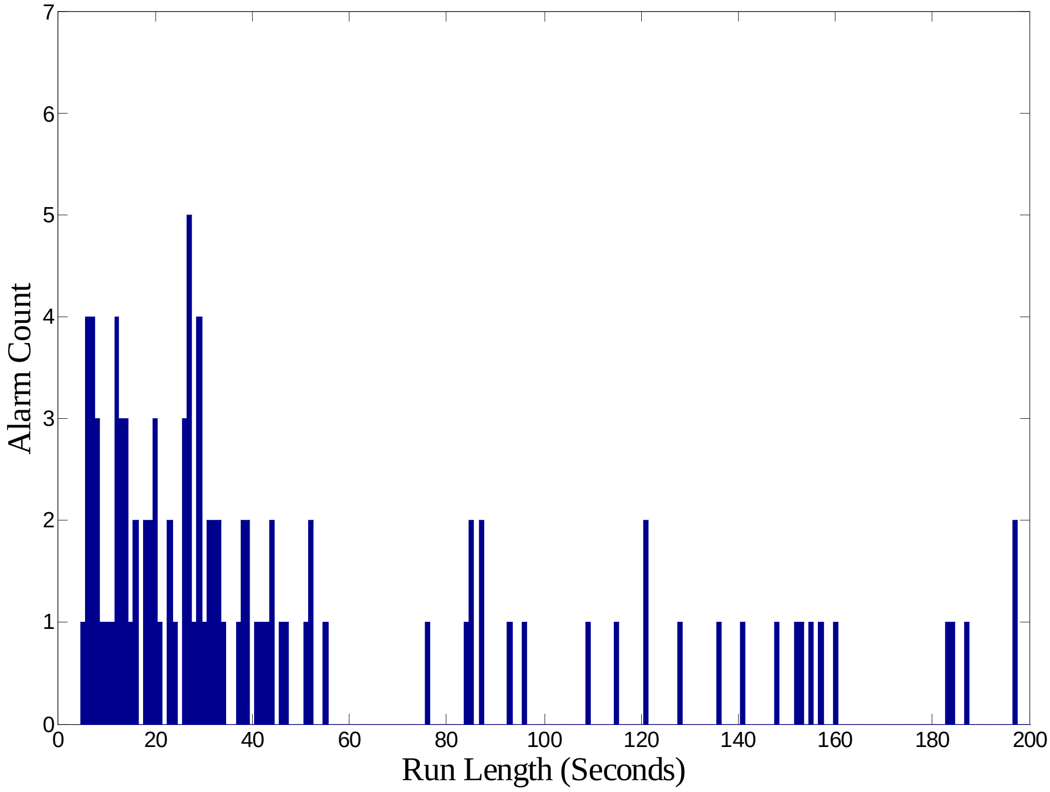


Figure 4.7: Run length distribution for the alarm as represented by tag.id1 tag.id2

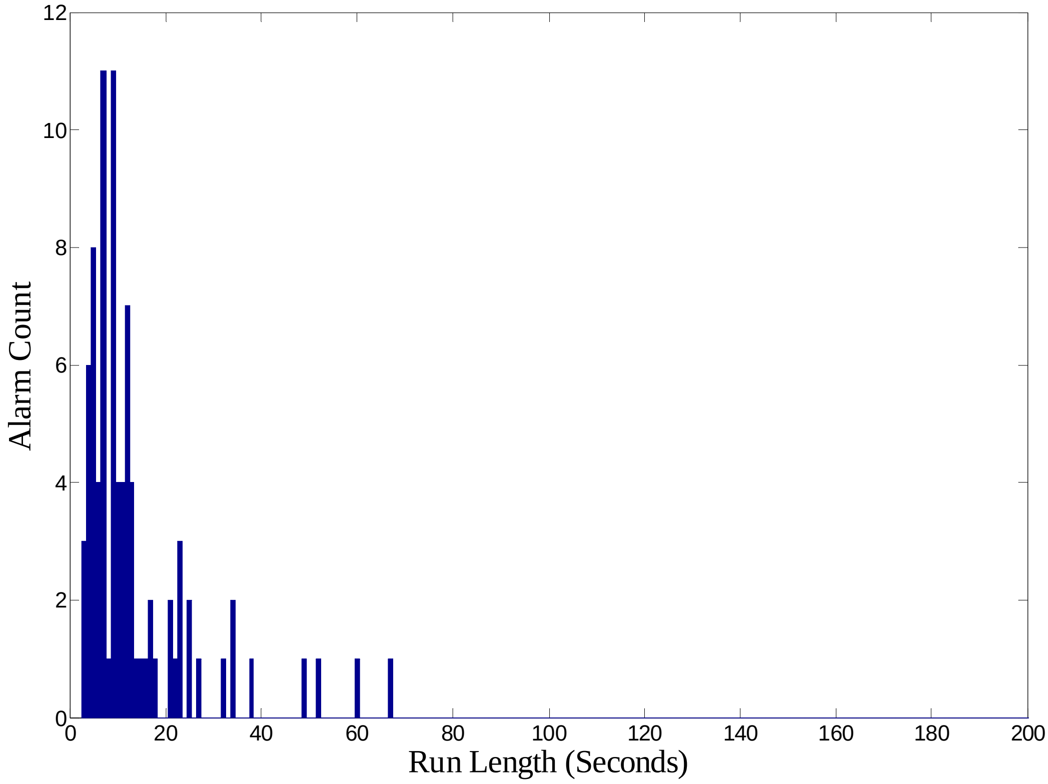


Figure 4.8: Run length distribution for the unique alarm as represented by tag.id3

Figure 4.8 shows the RLD for the unique alarm as represented by tag.id3. It is evident that most of the alarms have very short run lengths ranging from 3 seconds to about 70 seconds. Thus the unique alarm as represented by, tag.id3 is expected to have a higher chatter index compared to the unique alarm as represented by tag.id2.

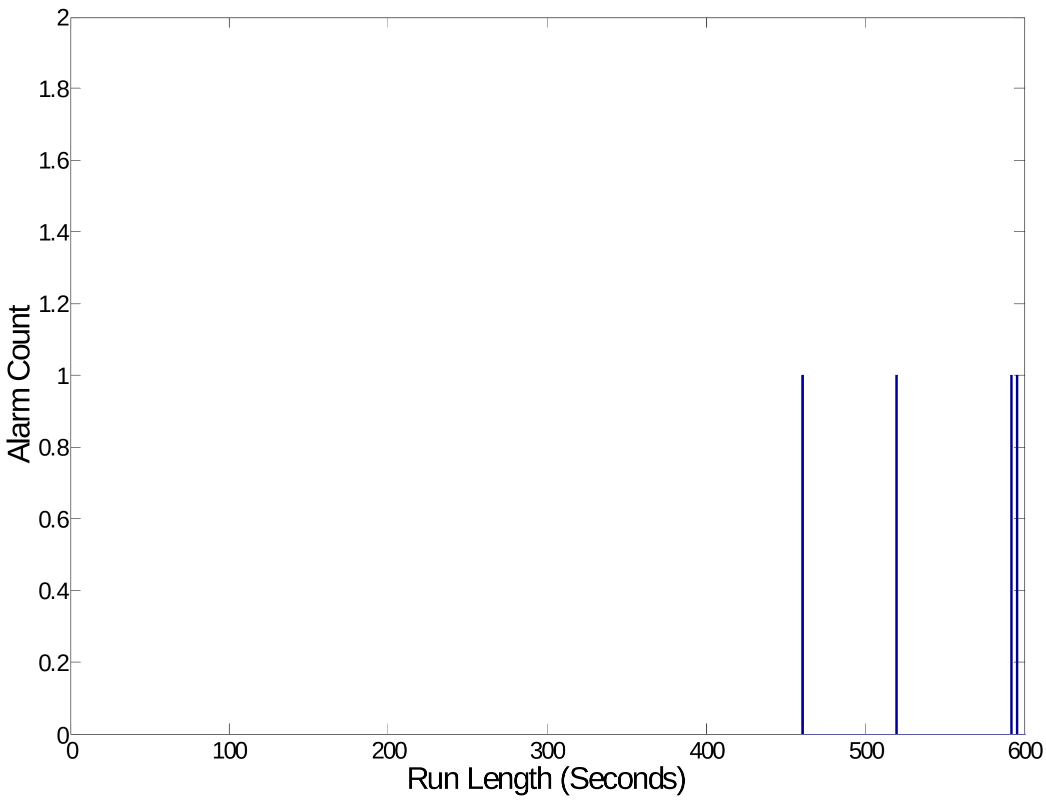


Figure 4.9: Run length distribution for the unique alarm as represented by tag.id4

RLD for the unique alarm as represented by tag.id4 in figure 4.9 shows that there are hardly any alarms with short run lengths. There is just one alarm with the shortest run length of 461 seconds. The unique alarm as represented by tag.id4, is expected to have a negligible chatter index.

4.6.3 Chatter indices Ψ and Ψ_τ

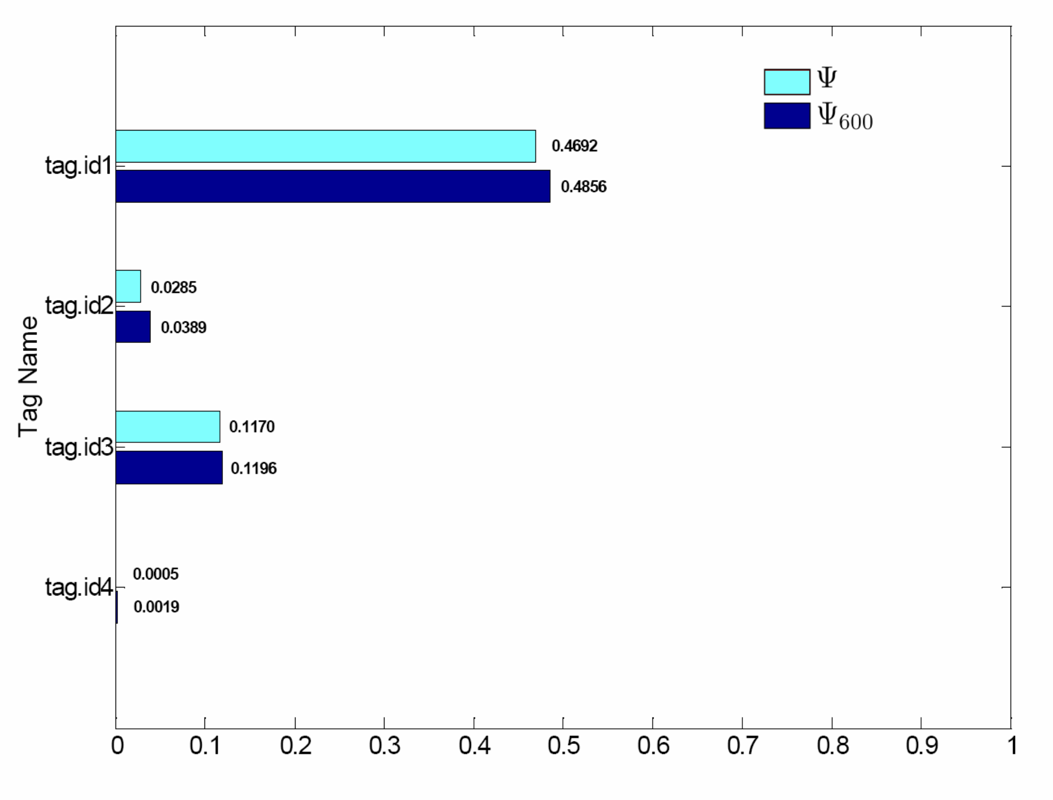


Figure 4.10: Comparision of chatter indices Ψ and Ψ_τ for the four tags

Figure 4.10 shows the bar chart of the chatter indices, Ψ and $\Psi_{\tau=600}$ for all the four unique alarms under consideration. Two striking observations based on this figure are given below.

Both Ψ and Ψ_{600} show a similar trend

A small value for $\tau = 600$ is chosen to calculate the chatter index. In calculating Ψ_{600} all the alarms with run lengths longer than 10 minutes (600 seconds) are ignored. It means that two consecutive alarms on the same tag that are separated by more than 10 minutes are assumed to represent two different abnormal events. It is interesting to note that longer run lengths contribute insignificantly to the chatter index on an absolute scale. $\Psi_{\tau=600}$ has a slightly larger value compared to Ψ mainly due to the fact that $P_{r,600} \geq P_r \ \forall\ r$. See A.3 for a detailed derivation.

Magnitudes of both Ψ and Ψ_{600} agree with visual observations

As expected, the unique alarms as represented by tag.id1 and tag.id3 showed a higher chatter index (in fact greater than $\Psi_{cutoff} = 0.05$) compared to the unique alarm as represented by tag.id2. The unique alarm as represented by tag.id4, has an insignificant value for both Ψ and Ψ_{600}. It is to be noted that the unique alarm as represented by tag.id3 has a higher chatter index compared to the unique alarm as represented by tag.id2 even though the overall alarm count is higher for the unique alarm as represented by tag.id2.

4.7 Improvement in chatter index after redesign of alarms

This section presents a case study where improvement in the alarm run length distribution and hence in the chatter index is observed because of appropriate design changes on each of the two real industrial unique alarms. The first unique alarm is a flow tag labeled as tag.id5 and the second one is a density tag labeled as tag.id6. Alarm data was collected for these two unique alarms for a period of over one month both before and after the alarm design changes were implemented.

Table 4.2: Chatter index before and after alarm design changes are implemented (computed using one month worth of alarm data)

		Before		**After**		
S.No	Tag Name	Alarm Count	Ψ	Alarm Count	Ψ	Percentage reduction in Ψ
1	tag.id5	218	0.1322	26	0.0009	99.3
2	tag.id6	1248	0.0848	44	0.0021	97.6

4.7.1 Flow tag - tag.id5

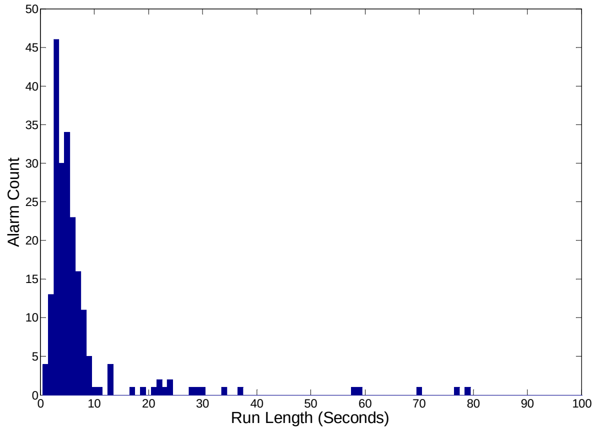
The original alarm design had a deadband of 5%. After reviewing the process data, it was found that there was a significant noise in the underlying process signal and appropriate filtering would help reduce chattering. The following changes were implemented : Moving average filter of length 5 was implemented to filter out the noise; with this change in addition to the existing deadband was kept at 5 %.

4.7.2 Density tag - tag.id6

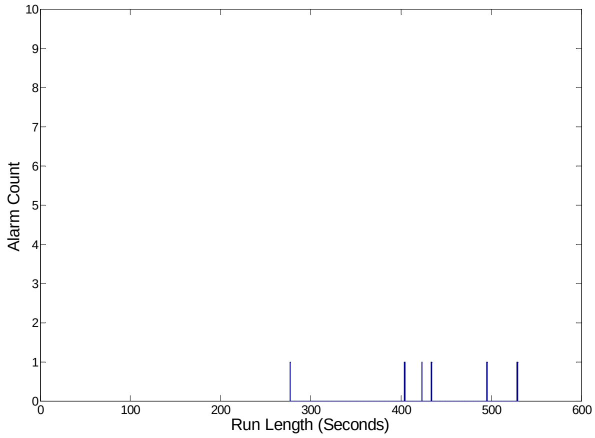
The original design had a deadband of 1%. After reviewing the data, it was concluded that a larger deadband would help reduce chattering. The following changes were implemented: Existing deadband of 1 % was increased to 5 %.

4.7.3 Results and Discussion

Figures 4.11(a) & 4.11(b) show the alarm run length distributions of the unique alarm as represented by tag.id5, both before and after the alarm design changes are implemented. It is evident that there are not as many alarms with short run lengths after the changes are implemented. The shortest run length observed

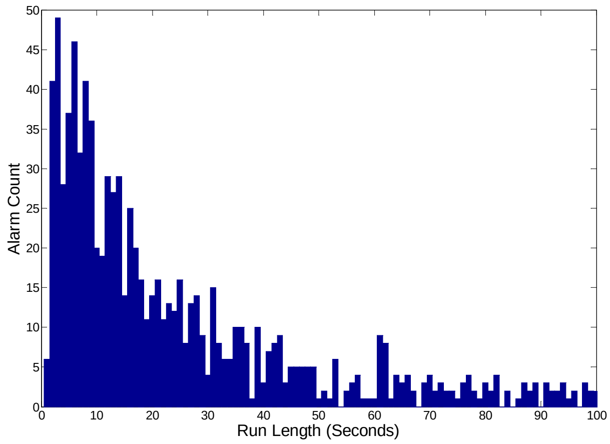


(a) Before

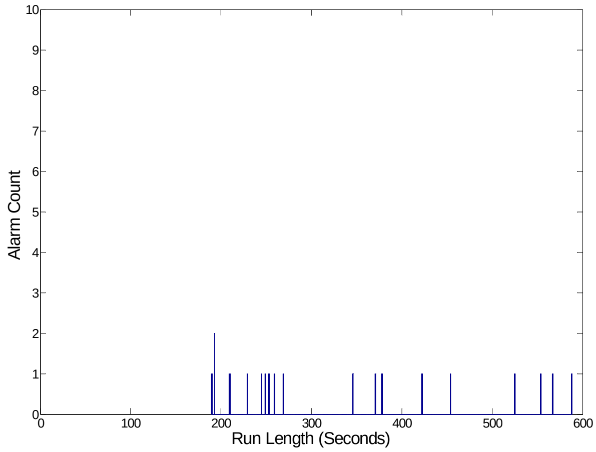


(b) After

Figure 4.11: Run length distribution for the unique alarm as represented by tag.id5 before and after the design changes are implemented



(a) Before



(b) After

Figure 4.12: Run length distribution for the unique alarm as represented by tag.id6

in 4.11(b) is close to 300 seconds. A similar result was achieved for the unique alarm as represented by tag.id6. The shortest run length observed in 4.12(b) is close to 200 seconds. Table 4.2 shows the values of Ψ for both the tags before and after the alarm design changes were implemented. It can be seen that there is significant reduction (over 90% reduction in Ψ for each tag) in chattering due to alarm design changes. The design changes implemented in this case study are based on experience and does not take into consideration the effects of detection delay [1] induced due to the changes.

4.8 Concluding remarks

Alarm system performance assessment is a crucial step in the alarm management life cycle. In this step, identification of nuisance alarms due to bad design and improper configuration is an important activity. Chattering alarms are the most common form of nuisance alarms and there are no standard procedures to identify them. In this work, a chatter index, Ψ is proposed based on RLDs using only the alarm data which is easily available. A variant of Ψ, Ψ_τ, with flexible assumptions is also proposed. It has been shown that for reasonable range of τ, there is no significant difference in the values of Ψ and Ψ_τ.

Ψ can be calculated automatically given the alarm data for a unique alarm over a period of time and hence reduces the effort required for identifying top chattering alarms as part of routine assessment of alarm system. A limit on Ψ has been calculated to identify the worst chattering alarms based on a rule of thumb ($\Psi_{cutoff} = 0.05$). The chatter indices proposed, Ψ or Ψ_τ, can be used in optimal design of a suitable filter in order to reduce chattering.

Bibliography

[1] N. A. Adnan, I. Izadi, and T. Chen, *On expected detection delays for alarm systems with deadbands and delay-timers*, Journal of Process Control **21** (2011), no. 9, 1318 – 1331.

[2] K. Ahmed, *Similarity analysis of industrial alarm flood data*, Master's thesis, University of Alberta, 2011.

[3] N. Balakrishnan and M. V. Koutras, *Runs and scans with applications*, 1 ed., Wiley-Interscience, New York, 2002.

[4] M. L. Bransby and J. Jenkinson, *The management of alarm systems: A review of best practice in the procurement, design and management of alarm systems in the chemical and power industries*, Tech. Report CRR 166, Health and Safety Executive, 1998.

[5] EEMUA, *Alarm systems: A guide to design, management and procurement*, 2 ed., EEMUA Publication No. 191 Engineering Equipment and Materials Users Association, London, 2007.

[6] A. J. Hugo, *Estimation of alarm deadbands*, In proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes (2009), 663–667.

[7] ISA, *Management of alarm systems for the process industries*, Tech. Report ANSI/ISA-18.2-2009, International Society of Automation, ISA, 67 Alexander Drive, P.O. Box 12277, Research Triangle Park, North Carolina 27709, 2009.

[8] I. Izadi, S. L. Shah, D.S. Shook, S. R. Kondaveeti, and T. Chen, *A framework for optimal design of alarm systems*, In proceedings of the 7th

IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, 2009, pp. 651–656.

[9] S. R. Kondaveeti, I. Izadi, S. L. Shah, and T. Black, *Graphical representation of industrial alarm data*, In proceedings of the 11th IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design and Evaluation of Human-Machine Systems, 2010.

[10] S. R. Kondaveeti, I. Izadi, S. L. Shah, and T. Chen, *On the use of delay timers and latches for efficient alarm design*, In proceedings of the 19th Mediterranean Conference on Control Automation (MED), 2011, pp. 970 –975.

[11] S.R. Kondaveeti, I. Izadi, S.L. Shah, T. Black, and T. Chen, *Graphical tools for routine assessment of industrial alarm systems*, Computers & Chemical Engineering **46** (2012), no. 0, 39 – 47.

[12] S. Kotz and N. L. Johnson, *Encyclopedia of statistical sciences*, Wiley, New York, 1988.

[13] Q. Li, J. R. Whiteley, and R. R. Rhinehart, *An automated performance monitor for process controllers*, Control Engineering Practice **12** (2004), no. 5, 537 – 553.

[14] D. C. Montgomery, *Introduction to statistical quality control*, Wiley, NY, 2001.

[15] A. Pariyani, W. D. Seider, U. G. Oktem, and M. Soroush, *Dynamic risk analysis using alarm databases to improve process safety and product quality: Part i - data compaction*, AIChE Journal **58** (2012), no. 3, 812 – 825.

[16] D. H. Rothenberg, *Alarm management for process control*, Momentum Press, New Jersey, August 2009.

[17] V. Venkatasubramanian, R. Rengaswamy, S. N. Kavuri, and K. Yin, *A review of process fault detection and diagnosis: Part iii: Process history based methods*, Computers & Chemical Engineering **27** (2003), no. 3, 327 – 346.

[18] A. Wald and J. Wolfowitz, *On a test whether two samples are from the same population*, Annals of Mathematical Statistics **11** (1940), no. 2, 147–162.

[19] F. Yang, S. L. Shah, D. Xiao, and T. Chen, *Improved correlation analysis and visualization of industrial alarm data*, ISA Transactions **51** (2012), no. 4, 499 – 506.

5

On the use of delay timers and latches for efficient alarm design

5.1 Overview

This chapter discusses delay timers and latches are often used in the process industry to reduce alarm chatter and to minimize nuisance alarms especially on the digital variables[1]. Effect of varying the size of on-delay, off-delay timers and latches on the accuracy of detection is discussed from a theoretical view point by modeling them using Markov chains. Use of Return to Normal (RTN) information in addition to alarm information in designing delay timers is also discussed with application to a real industrial case study.

[1]A condensed version of this chapter has been published in Proceedings of the 19^{th} Mediterranean Conference on Control and Automation June 20-23 2011, Corfu, Greece.

5.2 Introduction

In the process industries, plant operators should be able to rely on process alarms to provide essential information for running their units safely not only during normal process operation but also during process startups/shutdowns/upsets. However, in practice, due to improper management of the alarm systems, most process plants face decreased plant reliability, increased off-spec products and increased environmental excursions. Investigation reports following some major incidents such as BP Texas city refinery explosion and fire in March 2005 [14] and the Buncefield incident in December 2005 [5] have pointed at inefficient alarm systems as one of the major shortcomings of these incidents.

Proper management of an alarm system is crucial to achieve timely detection and diagnosis of abnormal operating conditions. A well managed alarm system must be defined and meticulously configured to be in compliant with standards such as EEMUA [2] and ISA [7]. A comprehensive introduction to the problem of alarm management for process control and some of the solutions are discussed in [13]. Description of various stages involved in an efficient alarm management life cycle is available in [6] and also emphasized in the ISA standard [7]. Two of the main stages where engineering aspects of alarm management play a key role are the *monitoring and assessment stage* and the *detailed design stage*. Problems identified in the monitoring and assessment stage can be rectified in several other stages of the life cycle. Identification of nuisance alarms such as chattering and related alarms is an important task in the monitoring and assessment stage. The work presented in [10] shows an automated way of identifying these nuisance alarms.

In the contract survey report [1] submitted to the United kingdom based Health and Safety Executive, it is mentioned that the single most common cause of nuisance alarms was repeating alarms or chattering alarms. These are alarms setting and resetting repeatedly in a short span of time. More information on chattering alarms and how alarm chatter can be quantified is discussed in detail in [11]. The work presented in [11] provides a convenient means to identify the list of chattering alarms from a given set of historical alarm data.

After a list of chattering alarms is obtained, appropriate changes need to be implemented on the alarm generating logic to reduce the occurrence of these alarms. In [9], a framework for optimal design of alarm systems is presented. As mentioned in [8] and [9], some of the simple techniques to reduce the occurrence of nuisance alarms require filtering of process data, adding delay timers and using alarm dead bands.

In this chapter, the applicability of delay timers is discussed in more detail by modeling them using Markov chains and analyzing the performance of delay timers of various lengths in the ROC framework as presented in [9]. A new technique called alarm latch which is very similar in functionality to an off-delay timer is introduced and analyzed in the ROC framework. Finally, a design procedure to select the type and length of a delay timer using historical alarm data and return to normal information is presented using real industrial data.

5.3 Delay timers and latches

5.3.1 Alarm delay timers

The attributes, on-delay and off-delay (also known as filter timer and debounce timer in the electrical engineering terminology), can be used to eliminate nuisance alarms (See [7] and [1] for more details). The on-delay timer is used to avoid nuisance alarms when a signal temporarily overshoots its limit, thus preventing the alarm from being triggered until the signal remains in the alarm state continuously for a specified length of time (equivalently a specified number of samples). Thus the use of on-delay timers induces detection delay. The off-delay timer is used to reduce chattering alarms by locking in the alarm indication for a certain holding period after it has cleared thus avoiding immediate resetting of alarms. Both techniques can be implemented simultaneously. The length of the on-delay timer/off-delay timer in terms of the number of samples is the main design parameter. Based on the type of signal being monitored, rules of thumb for selecting the length of on-delay timers are available in [2] and [7]. In this work, a more systematic approach is followed for designing the length of the delay timer.

5.3.2 Alarm latches

An alarm on-latch is very similar to the off-delay timer in the sense that both of them are used to delay the clearing of an alarm. The alarm on-latch is used to raise an alarm as soon as the underlying process variable crosses the check limits. The status remains in alarm for at least the duration of the length of the alarm on-latch. The alarm clears at the sampling instant after this duration when the process variable has a value within the alarm limit. An alarm off-latch is analogous to an alarm on-delay timer and can be defined in a similar way. Hence alarm off-latches are not discussed anymore in this work. Henceforth, for convenience, an alarm on-latch is referred to as an alarm latch or simply a latch.

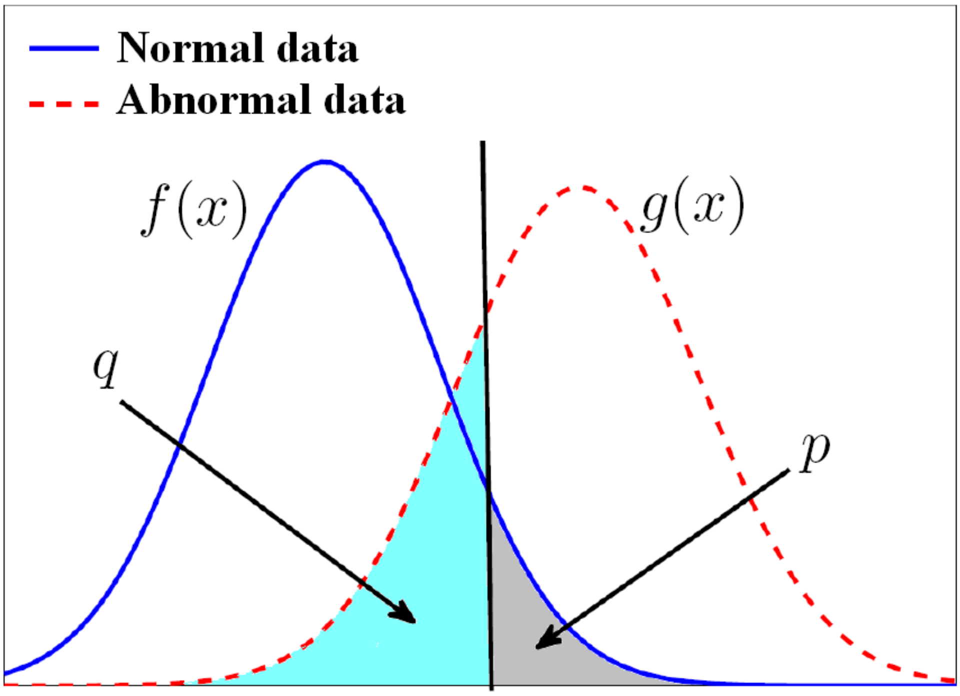


Figure 5.1: Distribution of normal data (solid), distribution of abnormal data (dashed) and a trip point (vertical line) as shown in [9]. p denotes the probability of false alarm and q denotes the probability of a missed alarm.

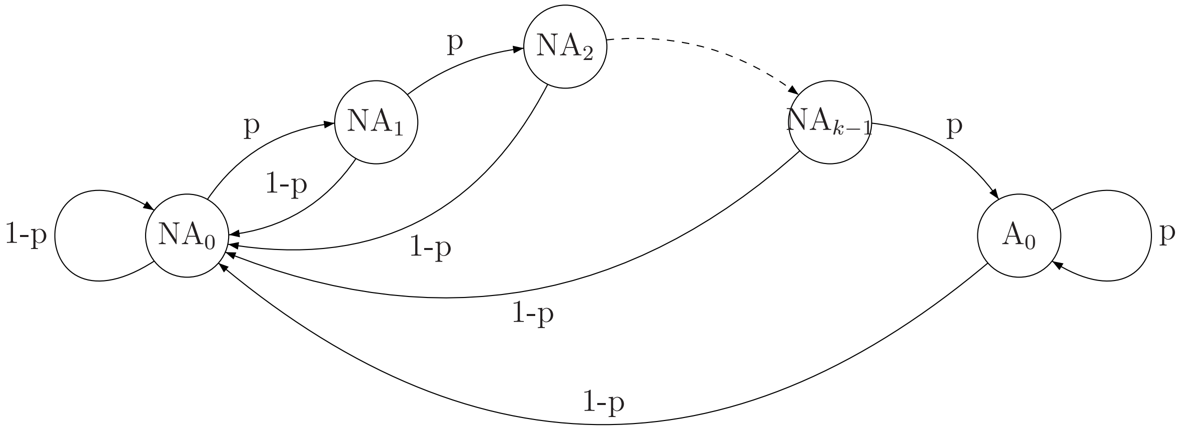


Figure 5.2: Markov chain representation of an on-delay timer of length k (transition probabilities during normal operation are shown along the signal paths)

5.4 Markov chains

Markov chains are the simplest mathematical models for random phenomenon evolving in time [12]. A Markov chain is a probabilistic model applying to systems that exhibit a special type of dependence, that is, where the state of the system on the n+1th observation depends only on the state of the system on the nth observation [15]. This simply means that the future changes in the system depend only on the current state and not on the way the system reached the current state. Markov chains are applied in a number of ways to a variety of fields such as physics, chemistry, biology, economics and finance.

In [3], a Markov chain is defined as a stochastic process, $\{X_t\}, t = 0, 1, 2, ...$ where X_t takes values in the finite set $S = \{1, 2, ...N\}$, and is such that

$$Pr(X_n = i_n | X_0 = i_0, ..., X_{n-1} = i_{n-1}) = Pr(X_n = i_n | X_{n-1} = i_{n-1}) \quad (5.1)$$

In the application at hand, the set S contains a finite number of elements and the state transitions occur at discrete time intervals thus it is called a discrete time Markov chain. The equation given implies that to make predictions about the future behavior of the system it suffices to consider only its present state and not its past history. The probability, $Pr(X_n = i_n|X_{n-1} = i_{n-1})$ is known as a one step transition probability. The more general transition probability, $Pr(X_n = i|X_m = j)$ satisfies the well known Chapman Kolmogorov equation given by

$$Pr(X_n = i|X_m = j) = \sum_k Pr(X_r = k|X_m = j)Pr(X_n = i|X_r = k) \tag{5.2}$$

$$\text{where } m < r < n$$

A time homogeneous Markov chain (or stationary Markov chain) is one for which the transition probabilities depend only on the difference, $n-m$, rather than on n or m. In particular, the one step transition probability, $Pr(X_n = i|X_{n-1} = j)$ can be written as simply p_{ij}. The $N \times N$ matrix, $\mathbf{P}$, with ij^{th} element p_{ij} is a stochastic matrix, i.e.

$$0 \leq p_{ij} \leq 1,\ 1 \leq i, j \leq N \tag{5.3}$$

$$\sum_{j=1}^{N} p_{ij} = 1,\ 1 \leq i \leq N \tag{5.4}$$

$\mathbf{P}$ is known as a transition matrix. The ij^{th} element of $\mathbf{P}^n$ gives the n step transition probability. For a time-homogeneous Markov chain with a finite

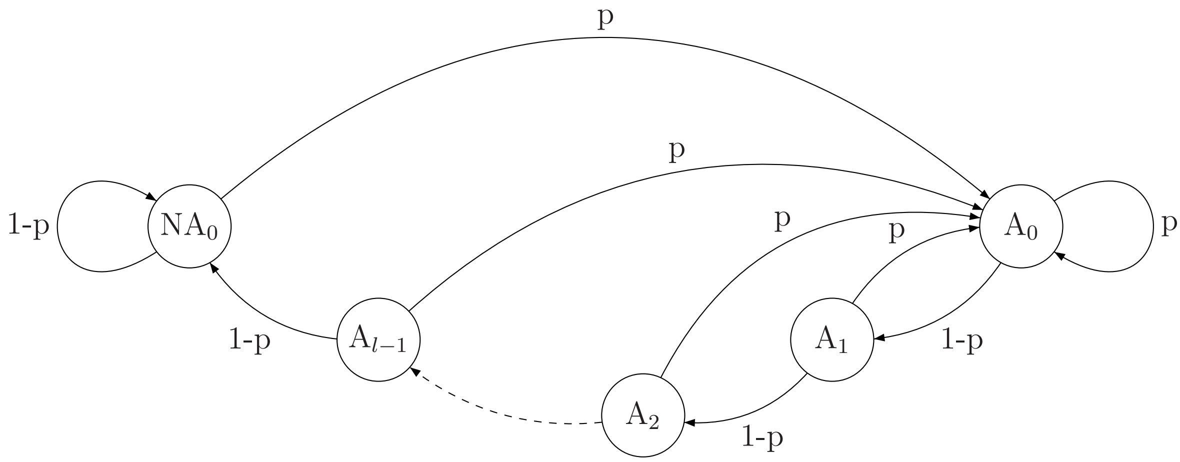


Figure 5.3: Markov chain representation of an off-delay timer of length l (transition probabilities during normal operation are shown along the signal paths)

state space, the stationary distribution π is a vector whose entries are non-negative, sum to unity and satisfies the equation

$$\pi = \pi \mathbf{P} \tag{5.5}$$

In other words, the stationary distribution π is a normalized (meaning that the sum of its entries is 1) left eigenvector of the transition matrix associated with unit eigenvalue. The i^{th} elements of the π vector represents the equilibrium probability that the state of the system is i.

5.5 Modeling delay timers and latches using Markov chains

Assume that the underlying process variable of a unique alarm follows a stationary distribution and is operating in its normal region. If a simple limit checking logic is used to generate an alarm, the overall false alarm rate would be equal to p as shown in Fig. 5.1. p is a function of only the alarm limit.

Fig. 5.2 shows the Markov process representation of an on-delay timer of length k in its normal operation. Assume that the initial state of an alarm tag is NA_0, the state of the alarm tag at the next sampling instant can be either NA_0 or NA_1 depending on whether the process variable lies within the alarm limit or violates it. As long as the state of the alarm tag is in any of NA_0,NA_1.....and NA_{k-1}, no alarm message is sent to the operator. If the process variable goes within the alarm limit at any sample , the state of the alarm tag immediately goes back to NA_0.

After the state of the alarm tag changes to A_0, an alarm message is sent to the process operator. This simply means that if the initial state of the alarm tag is NA_0, the underlying process variable must violate the alarm limit for at least k consecutive sampling instants to generate an alarm. The state transition matrix for an on-delay timer in normal operation is shown below:

$$\mathbf{P} = \begin{array}{c} \\ NA_0 \\ NA_1 \\ NA_2 \\ \vdots \\ NA_{k-1} \\ A_0 \end{array} \begin{array}{c} \begin{array}{cccccc} NA_0 & NA_1 & NA_2 & \cdots & NA_{k-1} & A_0 \end{array} \\ \begin{pmatrix} 1-p & p & 0 & \cdots & 0 & 0 \\ 1-p & 0 & p & \cdots & 0 & 0 \\ 1-p & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1-p & 0 & 0 & \cdots & 0 & p \\ 1-p & 0 & 0 & \cdots & 0 & p \end{pmatrix} \end{array}$$

Fig. 5.3 shows the Markov chain representation of an off-delay timer of length l in normal operation. For an off-delay timer, an alarm is raised as soon as the underlying process variable violates the alarm limits. For the alarm tag to return to normal, l consecutive samples must be within the alarm limits.

Fig. 5.4 shows the Markov chain representation of an alarm latch of length m. When an alarm latch is used, an alarm is raised as soon as the underlying process variable violates the alarm limits. The status of the tag remains in alarm for at least m sampling instants. Only after these m sampling instants can the status change to return to normal depending on the value of the underlying process variable. The state transition matrix for off-delay timer and alarm on latch during normal and abnormal operation can be defined according to their corresponding Markov chain representations and are not shown in this work.

5.5.1 Receiver Operating Characteristic curve

Assume that the cumulative distribution function (CDF) of the normal operating data for a random variable, X is denoted by $F(x)$ and that of the abnormal data is denoted by $G(x)$. Fig. 5.1 shows an example of both the distributions where the abnormal data ($g(x)$) has a higher mean compared to that of the normal data ($f(x)$). Thus in this case, the random variable is monitored for a high alarm limit. If it is known that the random variable is operating in the normal region but the instantaneous value is greater than the high alarm limit (area with probability p in Fig. 5.1), a false alarm is generated. Similarly, the area with probability q in Fig. 5.1 denotes a missed alarm as the actual distribution of the variable comes under abnormal operation but the instantaneous value is less than the high alarm limit. Thus, once the value of high alarm limit is fixed, the probability of false alarm (p) and the probability of missed alarm (q) can be calculated. A plot of False Alarm Rate (FAR) vs the corresponding Missed Alarm Rate (MAR) for various alarm limits is known as the Receiver Operating Characteristic (ROC) curve [4]. ROC curves are extremely useful because they can be generated using historical data (of

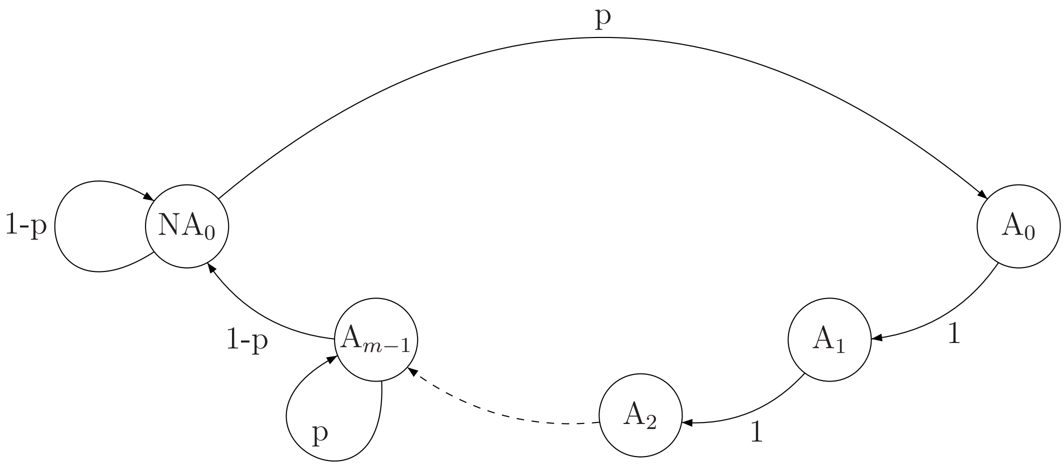


Figure 5.4: Markov chain representation of a alarm on-latch of length m (transition probabilities during normal operation are shown along the signal paths)

both normal and abnormal operation) and they are especially useful in cases where there are skewed distributions and unequal classification error costs. The utility of ROC curves in visualizing the performance of various alarm generating algorithms has been discussed in [9].

5.5.2 ROC analysis for delay timers and latches

In the presence of a delay timer or a latch to generate an alarm, the FAR and MAR must be calculated using the stationary distributions, π_{no} and π_{ano} corresponding to normal operation and abnormal operation respectively.

$$\text{FAR} = \sum \pi_{no}(A_i) \tag{5.6}$$

$$\text{where } i = \begin{cases} 0, & \text{for on-delay timer} \\ 0, 1, 2...l-1, & \text{for off-delay timer} \\ 0, 1, 2...m-1, & \text{for a latch} \end{cases}$$

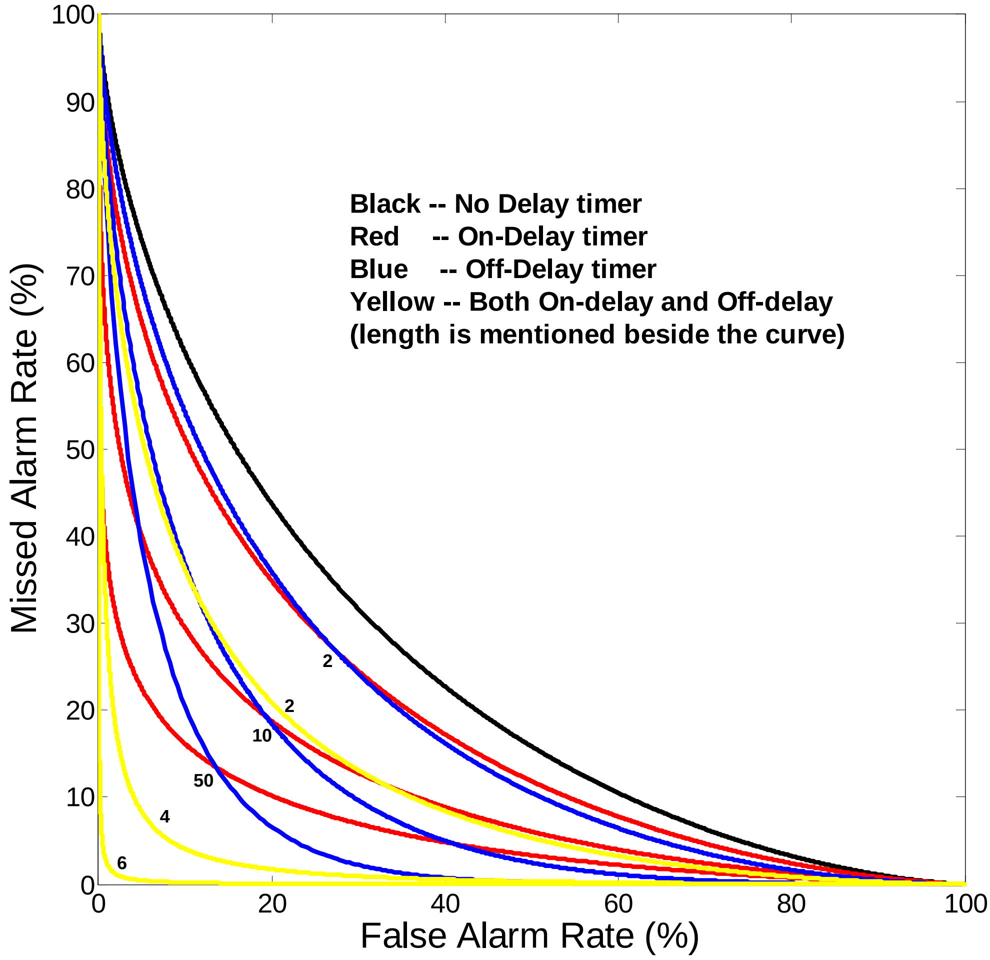


Figure 5.5: ROC curves with the use of various delay timers

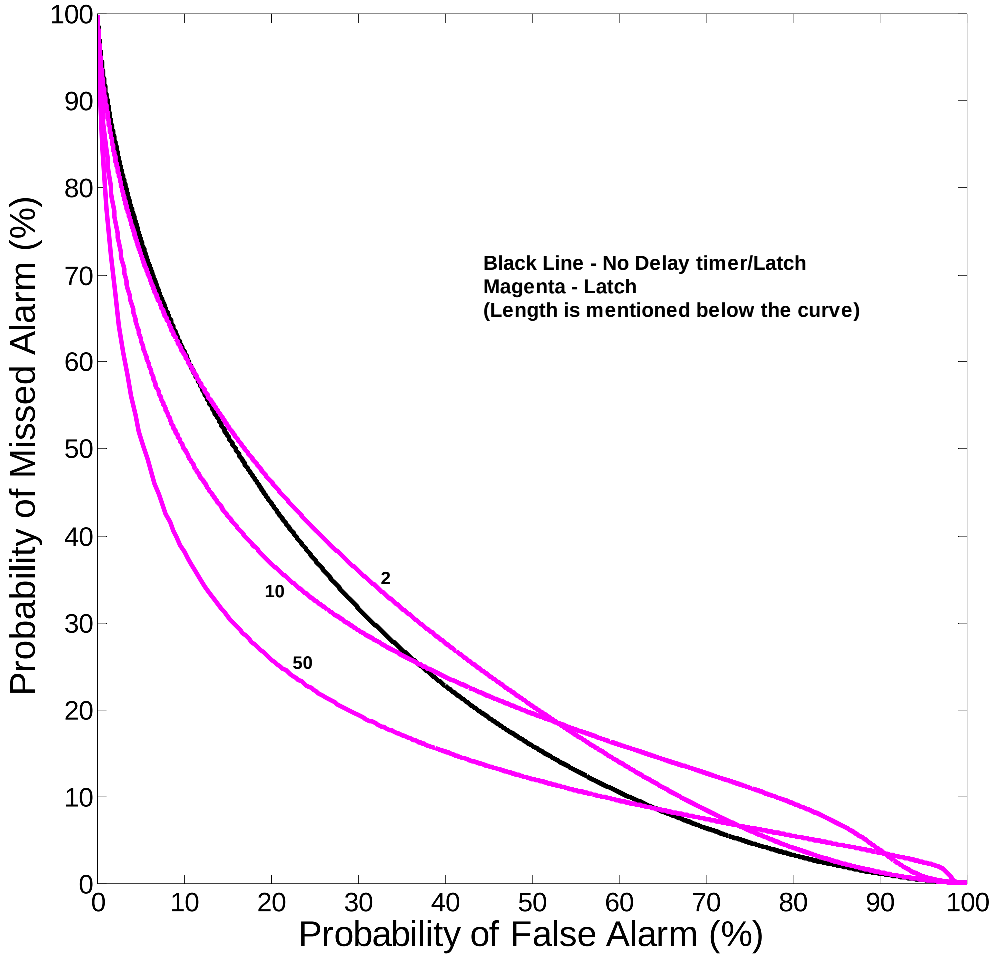


Figure 5.6: ROC curves with the use of various latches

$$\text{MAR} = \sum \pi_{ano}(NA_j)$$
$$\text{where } j = \begin{cases} 0, 1, 2...k-1, & \text{for on-delay timer} \\ 0, & \text{for off-delay timer} \\ 0, & \text{for a latch} \end{cases}$$

The combination of FAR and MAR is calculated for various values of p ranging from 0 to 1. It is to be noted that in calculating π_{ano}, the state transition probability matrix used looks very similar to that of the normal operation except for the change that all the p's are replaced by $1-q$.

5.5.3 Case study and discussion

For simplicity, assume that both normal and abnormal data for the underlying process variable, X of a unique alarm follow normal distributions. Let the distribution of X in normal operation be $\mathcal{N}(0,1)$ and that during abnormal operation be $\mathcal{N}(1,1)$.

Fig. 5.5 shows the ROC curves for delay timers of various lengths. The black line in Fig. 5.5 is the ROC curve when simple limit checking logic is used to raise and clear an alarm. In ROC curve, the best alarm limit corresponds to the FAR and MAR pair on the curve that is closest to the origin assuming both false alarms and missed alarms are equally undesirable.

It can be seen from Fig. 5.5 that as the length of the delay timer is increased, the ROC curve comes closer to the origin. The ROC curves for delay timers of only selected few lengths are displayed due to clarity in presentation. A smooth transition trend in the ROC curves is observed if all the lengths are considered. Fig. 5.5 also shows the ROC curves for various lengths of combined delay timers (both on-delay timer and off-delay timer of same length are implemented simultaneously). The Markov chain representation of such a delay timer is not shown in this work but it is basically a combination of on-delay timer and off-delay timer of the same length. An analytical expression for the FAR and MAR for this case is presented in [9].

The performance of the combined delay timer is much better in terms of accuracy compared to the performance of either an on-delay timer or an off-delay timer of the same length. The yellow lines in Fig. 5.5 that represent the ROC curves for combined delay timers are symmetric about the FAR=MAR line. However, the ROC curves for pure on-delay timer or an off-delay timer are not symmetric about the same line. This observation can be explained from the fact that on-delay timers are meant to reduce false alarms and off-delay timers are meant to reduce missed alarms.

Fig. 5.6 shows the ROC curves for latches of various lengths. Two striking observations can be drawn from this figure. The first observation is that there is no significant improvement achieved by latches of various lengths in terms of detection accuracy. The next observation is that the curve drifts towards higher false alarm rates and the curves come closer to the axes only at extreme points (FAR=0%,MAR=100% and FAR=100%, MAR =0%).

5.6 Design of delay timers using historical alarm and return to normal data in an industrial case study

Most process control systems have the capability to store alarm event data into a historian. Other events that are logged include but are not limited to Return To Normal (RTN), operator acknowledgement, operator action and alarm setting changes. In this part of the work, a design procedure to select the type and length of the delay timer based on historical alarm and RTN data is presented. In [11], a systematic procedure to identify worst chattering alarms is presented. Once the list of alarms is available, for each alarm tag, a histogram of time difference between each pair of alarm followed by a RTN is plotted. Pale blue colored bars in Fig. 5.7 shows a truncated (only time differences up to 20 seconds are shown) and normalized (expressed as percentage) histogram for a real industrial alarm tag. The pale blue line in Fig. 5.7 shows the cumulative distribution function of the histogram represented by the pale blue bars. For instance, over 50% of the alarms have alarm to RTN distance of less than or equal to 5 seconds. If we were to use an on-delay timer of length 5 seconds, all these alarms can be prevented. This method provides a design strategy to select an appropriate on-delay timer.

It is to be noted that the state transition in the Markov chain representation occurs once every sampling instant. The sampling instant is the same as the scan time of a Programmable Logic Controller (PLC) for instance. On-delay timers can be implemented by selecting appropriate time in seconds of the delay in detection that can be tolerated. The equivalent length of the Markov chain representation of the on-delay timer is obtained by dividing this time by the scan time of the controller. For example, if the scan time of the controller is 5 milliseconds, then the length of the Markov chain for a 5 seconds on-delay timer is 1000. Thus, as the scan time of the controller decreases, delay timers become more efficient.

Similarly a histogram of time difference between each pair of RTN to alarm is also plotted using red bars in Fig. 5.7. The cumulate distribution function represented by the red line shows the % reduction in alarms by using an off-delay timer. It is safe to use a pure off-delay timer as it does not induce any detection delay.

For this particular case study, the use of off-delay timer of length 5 would reduce 70% of the alarm occurrences. All 70% of these alarms can be considered as nuisance as they reappear within a short time (5 seconds in this case) of

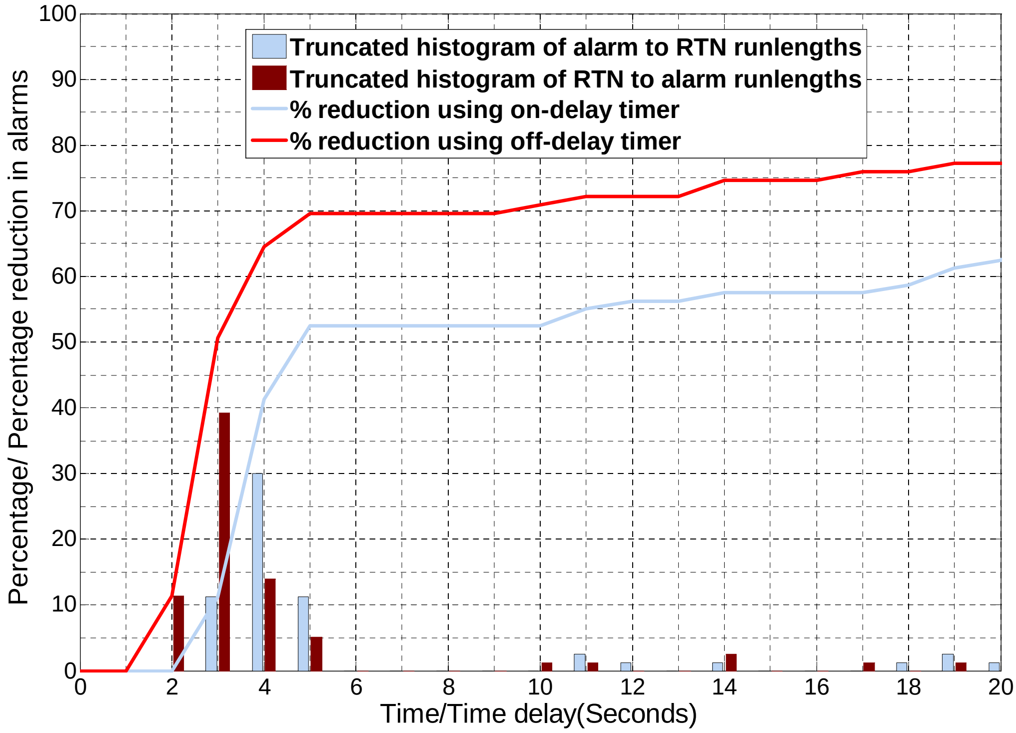


Figure 5.7: Design of delay timers using alarm and return to normal data

the occurrence of their corresponding predecessors.

5.7 Concluding remarks

This chapter discusses the performance of two commonly used techniques to reduce nuisance alarms in the process industry, namely, delay timers and latches. The performance of these techniques in terms of detection accuracy is discussed in the ROC framework by modeling them using Markov chains. It has been shown that the performance of combined delay timers is much better than the pure delay timers of the same length. Alarm latches do not perform anywhere close to delay timers when it comes to detection accuracy but they may be used to monitor critical variables without detection delay and chattering problems. Finally, a design methodology using historical alarm and RTN data to select the type and length of delay timers is illustrated using a real industrial case study.

Bibliography

[1] M. L. Bransby and J. Jenkinson, *The management of alarm systems: A review of best practice in the procurement, design and management of alarm systems in the chemical and power industries*, Tech. Report CRR 166, Health and Safety Executive, 1998.

[2] EEMUA, *Alarm systems: A guide to design, management and procurement*, 2 ed., EEMUA Publication No. 191 Engineering Equipment and Materials Users Association, London, 2007.

[3] B. S. Everitt and A. Skrondal, *The cambridge dictionary of statistics*, 4 ed., Cambridge University Press, 2010.

[4] T. Fawcett, *An introduction to roc analysis*, Pattern Recognition Letters **27** (2006), no. 8, 861–874.

[5] Health and Safety Executive 100021025, *The buncefield investigation: Third progress report*, http://www.buncefieldinvestigation.gov.uk/reports/volume1.pdf (December 2008).

[6] B. R. Hollifield and E. Habibi, *Alarm management: Seven effective methods for optimum performance*, ISA, Research traingle park, NC, 2007.

[7] ISA, *Management of alarm systems for the process industries*, Tech. Report ANSI/ISA-18.2-2009, International Society of Automation, ISA,

67 Alexander Drive, P.O. Box 12277, Research Triangle Park, North Carolina 27709, 2009.

[8] I. Izadi, S. L. Shah, D.S. Shook, and T. Chen, *An introduction to alarm analysis and design*, In proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, 2009, pp. 645–650.

[9] I. Izadi, S. L. Shah, D.S. Shook, S. R. Kondaveeti, and T. Chen, *A framework for optimal design of alarm systems*, In proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, 2009, pp. 651–656.

[10] S. R. Kondaveeti, I. Izadi, S. L. Shah, and T. Black, *Graphical representation of industrial alarm data*, In proceedings of the 11th IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design and Evaluation of Human-Machine Systems, 2010.

[11] S. R. Kondaveeti, I. Izadi, S. L. Shah, D.S. Shook, and R. Kadali, *Quantification of alarm chatter based on run length distributions*, In proceedings of the 49th IEEE Conference on Decision and Control, 2010, pp. 6809–6814.

[12] J. R. Norris, *Markov chains*, Cambridge University Press, 1997.

[13] D. H. Rothenberg, *Alarm management for process control*, Momentum Press, New Jersey, August 2009.

[14] United States Chemical Safety and Hazard Investigation Board, *Investigation report, refinery explosion and fire*, http://www.csb.gov/assets/document/CSBFinalReportBP.pdf (March 2007).

[15] A. Tamir, *Applications of markov chains in chemical engineering*, Elsevier Science, 1998.

6

Application of Multivariate Statistics for Efficient Alarm Generation

6.1 Overview

This chapter discusses advantages of monitoring the PCA based T^2 and Q statistic over individual process variables[1]. Check limits on univariate alarms for economical process operation are usually based on statistical quality control (three sigma limits, also known as Shewhart charts). While annunciating a univariate alarm on a particular variable, the information from other variables is often ignored. Modern day process plants have variables which are highly correlated. This correlation structure can be exploited in the efficient management of alarms. This work demonstrates the advantages of monitoring the

[1] A shorter version of this chapter has been published in Proceedings of the 7^{th} IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, Barcelona, Spain, June 30 - July 3, 2009.

PCA based T^2 and Q statistic over individual process variables. Monitoring these higher level statistics will not only reduce the false alarm and missed alarm rates but also reduces the detection latency which is one of the main drawbacks of monitoring a filtered variable. Two simulation examples and a simple industrial case study are shown to illustrate the utility of the proposed method.

6.2 Introduction

With increasing complexity in process industries, the study of abnormal event reduction has attracted a lot of attention in the recent past. Abnormal events in process plants lead to a variety of consequences ranging from a simple subsystem breakdown to as worse as loss of human life. On the economic front, an unplanned plant shutdown can wipe out all the benefits realized by advanced process control strategies. All abnormal events are initiated by a fault. Thus timely fault detection and diagnosis is a crucial step in reducing potential abnormal events.

6.2.1 Fault detection and process monitoring

The term *fault* is generally defined as a departure from an acceptable range of an observed variable or a calculated parameter associated with a process [3]. The criterion for delineating a fault is a subjective task and is by no means straightforward if the stochastic aspects are taken into account. Thus, the definition of a fault depends on the characteristics of the process variables, their acceptable ranges, and the accuracy of the statistic used for classification of a potential fault.

There is a lot of literature on process fault detection ranging from analytical methods to artificial intelligence and statistical approaches [8]. From a modeling perspective, there are methods that require accurate process models, semi-quantitative models, or qualitative models. At the other end of the spectrum, there are methods that do not assume any form of model information and rely only on historical process data. Owing to their ease of use, better diagnosis capability and simplicity, the signal processing and statistics based approaches are widely embraced for process monitoring.

To ensure process specifications and safety, faults in the process need to be detected, diagnosed and eradicated. These tasks are associated with process monitoring [7]. Thus the objective of any process monitoring scheme would be to ensure that the plant operators are informed of faults in plant behavior in the form of alarms before they get worse and lead to a subsystem/system failure.

6.2.2 Alarms and alarm systems

In the control room terminology, an *alarm* is a notification that a fault/abnormal event has occurred and an operator must take action. Thus, alarm monitoring which is part of process monitoring is the plant operators' duty. The purpose of an alarm is to protect personnel, equipment or the process from unsafe conditions, or to alert the operator when the process tends to drift towards low quality production.

An alarm would be annunciated if the variable being monitored crosses the check limits (High alarm limit and low alarm limit). These check limits are called alarm limits (or control limits in the statistical quality control terminology). The alarm limits depend on the type of variable being monitored (ex: raw signal, error signal or filtered signal) and in some situations there can be just one limit(High or Low) instead of both or there can be more than two limits (high-high, high, low, low-low etc). The count of alarms on a variable should ideally be the same as the number of actions available at the operators disposal. An *alarm system* is a collection of alarms configured on a set of variables which is to be monitored by the control room operator. A well designed alarm system is expected to provide precise information regarding the health of the process. Thus the control room operator relies on the alarm system for monitoring the process.

Control and monitoring practice in process industries changed drastically after the advent of Distributed Control System (DCS) in the late 70s. The DCS introduced software alarms - alarms that are created or changed by configuring a setting in a computer, rather than requiring a hard-wired signal to a panel. As a result, more alarms could be configured at no extra cost. This resulted in sloppy alarm design on most of the variables leading to ”nuisance alarms” - alarms that do not tell the operator anything he/she does not already know, or which do not require operator action. As a consequence of this oversight, alarm flooding occurs during an abnormal situation. Alarm flooding is a condition where alarms appear on the control panels at a rate faster than the operator can comprehend or respond to. Alarm flooding prevents the operator from determining the root cause of the process upset and therefore limits the scope for effective and quick emergency response.

6.3 Multivariate Statistical Process Control

The main disadvantage of univariate monitoring schemes is that for a process with many variables, the correlation between the process variables is not considered while setting the individual alarm limits. Moreover, the difficulty in univariate process monitoring increases with the complexity of the process. Multivariate quality control (MQC) methods overcome this disadvantage by

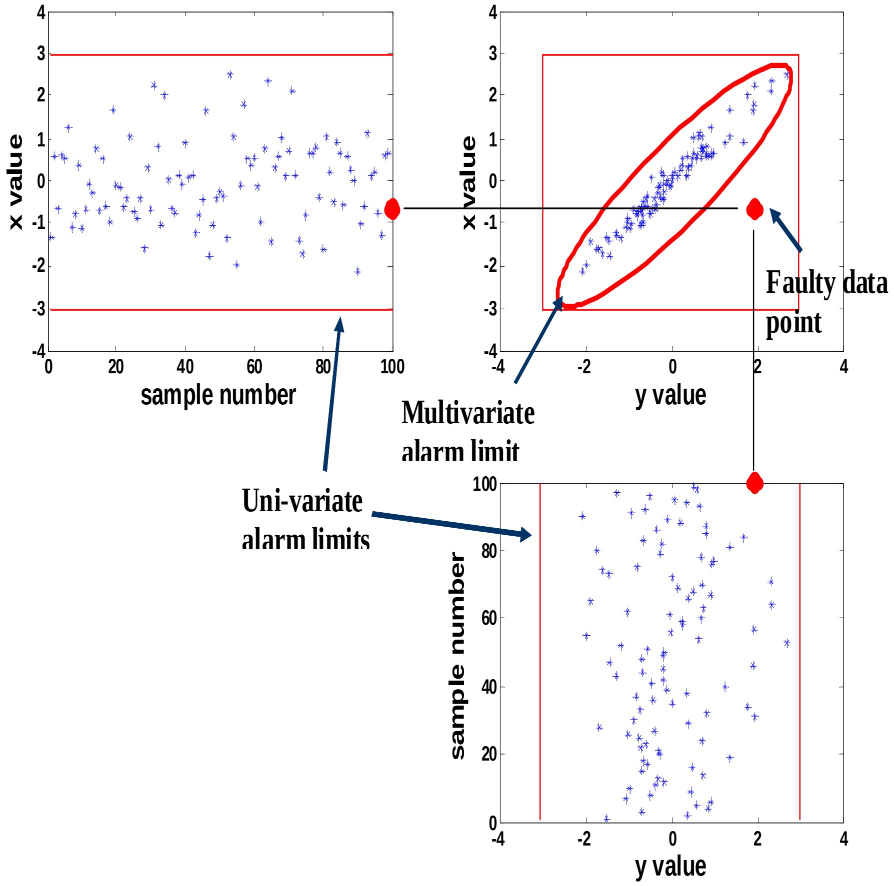


Figure 6.1: Univariate versus multivariate process monitoring

monitoring several variables simultaneously [4]. A simple way to see the advantages of MQC is to superimpose univariate control charts on top of each other and create a graph of all the points of each control chart in an area of space.

Fig. 6.1 shows three plots. Top left and the bottom right are individual scatter plots of multivariate data composed of two variables ($p = 2$), x and y. The top right plot shows the graph of x v y. The individual control limits for each variable's respective univariate chart are shown in the control rectangle.

Though there is a faulty data point in the given data, the univariate control limits are not violated. If the strong correlation between x and y is exploited, the control limit on the process could be narrowed down and one such control limit shown as an ellipse in the top right plot of Fig. 6.1 would be able to flag the fault. Assuming that a multivariate process with p variables is in control (no fault), the joint probability of a false alarm is

$$1 - (1 - \alpha)^p \approx \alpha p \tag{6.1}$$

for small univariate false alarm rate, α. The value of the joint probability of false alarm rate increases as the number of process variables increase.

6.4 PCA based Statistical Process Monitoring (SPM)

Principal Component Analysis (PCA) is a well known dimensionality reduction technique [9]. By choosing the appropriate number of Principal Components (PCs), a given data set can be projected on to two orthogonal subspaces, the principal component subspace (PCS) and the residual subspace (RS). PCS captures the normal variation of the data where as RS ideally captures only noise. Let $x \in \mathbb{R}^m$ denote a sample vector of m sensors. Collecting the normal operating data for N samples, a data matrix X can be formed with each row representing a sample and each column representing a sensor. For correlation based PCA, the matrix X is scaled to zero mean and unit variance. The matrix X can be decomposed into a score matrix T and a loadings matrix P as

$$X = \bar{T}\bar{P}^T = TP^T + \tilde{T}\tilde{P}^T = \hat{X} + \tilde{X} \tag{6.2}$$

Given a sample vector x, it can be projected on PCS and RS as

$$\hat{x} = PP^T x \tag{6.3}$$

and

$$\tilde{x} = (I - PP^T)x \tag{6.4}$$

Two fault detection indices, T^2 statistic and Q statistic are used as a measure of variability of a given sample in PCS and RS respectively. They are defined as follows:

$$\text{Q statistic} = \text{SPE} = \|\tilde{x}\|^2 = \|(I - PP^T)x\|^2 \tag{6.5}$$

and

$$T^2 = x^T P \Lambda^{-1} P^T x \tag{6.6}$$

where Λ is a diagonal matrix of eigen values corresponding to the eigen vectors in P.

A higher value for the T^2 statistic suggests that the sample is farther away from the normal operating region in the PCS. If the same sample has a small value for the Q statistic, it means that the correlation structure is not broken. Unless the fault magnitude is very high, the break in correlation does not increase the T^2 statistic significantly. The Q statistic is very sensitive to any break in correlation structure in the sample thus a low value of Q statistic indicates that there is no change in correlation structure. Thus the T^2 statistic and Q statistic play complementary roles [5]. Monitoring Q statistic is especially useful because it is sensitive to most common faults such as sensor failure, sensor bias, leaks in flows etc.

		True Class	
		Fault	No-Fault
Hypothesized class	Alarm	**T**rue **A**larms (TA)	**F**alse **A**larms (FA)
	No-alarm	**M**issed **A**larms (MA)	**T**rue “**N**o-alarms” (TN)

Figure 6.2: Confusion matrix in the alarm system terminology

6.5 Receiver Operating Characteristics (ROC) curves

The ROC curve is a graphical tool for visualizing a classifiers’ performance. ROC curves have long been used in fields like signal detection theory, medical diagnostic testing and machine learning to depict the tradeoff between hit rates and false alarm rates [2]. ROC curves are well appreciated in domains where there are skewed class distributions and unequal classification error costs. In this work, fault detection is dealt as a two class classification problem. The true classes are fault and no-fault, the corresponding hypothesized classes are alarm and no-alarm and the classifier is an alarm limit. Thus, given a sample of a signal (either raw or processed), it is compared with an appropriate threshold (classifier or alarm limit) and is mapped to one of the two classes, alarm (in case the threshold is violated) and no-alarm (otherwise). The confusion matrix is shown in Fig. 6.2.

Entries in the confusion matrix change as the discriminating threshold (alarm limit) is varied. Various measures can be defined for the classifiers performance. The most important ones are the false alarm rate and the missed alarm rate. They are defined as follows:

$$\text{False Alarm Rate (FAR)} = 100 * \frac{FA}{FA+TN}\% \tag{6.7}$$

$$\text{Missed Alarm Rate (MAR)} = 100 * \frac{MA}{MA+TA}\% \tag{6.8}$$

Both FAR and MAR are problematic to the operator and the objective is to choose alarm threshold limit to attain optimum balance (trade-off) between the two.

Traditionally, ROC curves are plotted as FAR v 100-MAR. However, in this work, for better visualization and analysis, ROC curves are represented equivalently by plotting FAR v MAR. The best alarm limit usually corresponds to the point on the ROC curve closest to the origin (FAR = 0 %, MAR = 0%) when both false alarms and missed alarms are equally undesirable. The alarm rates corresponding to the best alarm limit are hereafter called as Minimum FAR (MFAR) and Minimum MAR (MMAR). Following is a simple illustration of an ROC curve and how MFAR and MMAR can be calculated.

Fig. 6.3 shows the ROC curve plotted as the alarm limit is varied to discriminate between normal operating data (N(0,1)) and faulty data (N(2.5,2.25)) where $N(\mu, \sigma^2)$ represents normal distribution with mean μ and variance σ^2. The inset shows both the normal operating and faulty data. The design procedure involves selecting an appropriate alarm limit using the ROC curve. For example if the operator considers both false alarms and missed alarms to be equally undesirable, the best alarm limit corresponds to the point nearest to origin on the ROC curve. However, if the operator is willing to face more false alarms than missed alarms due to safety concern, an appropriate alarm

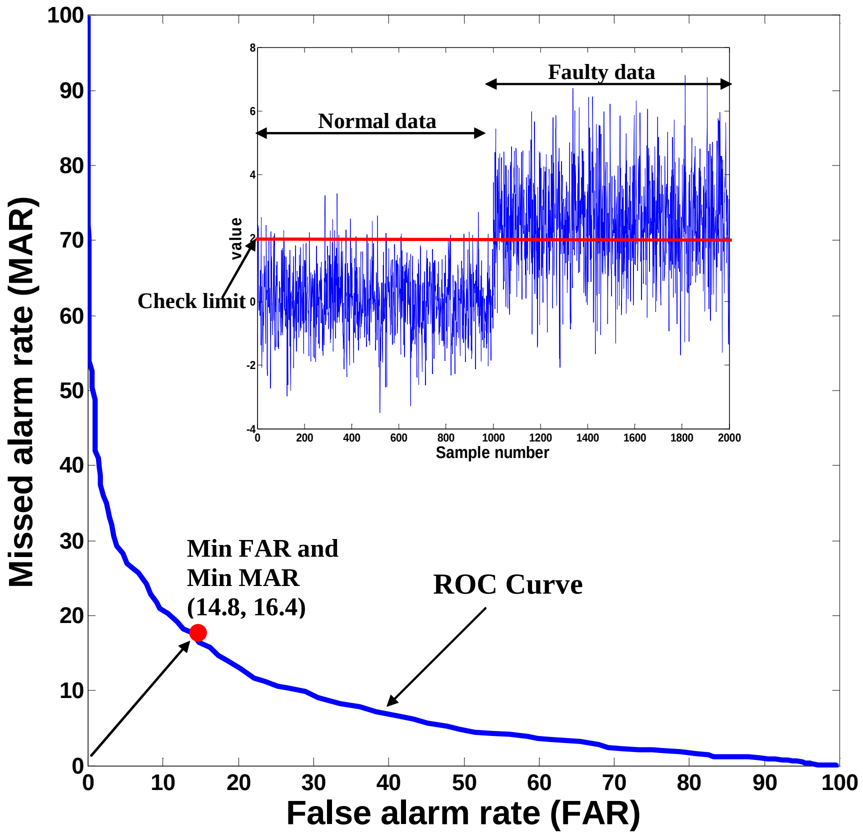


Figure 6.3: A simple ROC curve showing the selection of the best threshold

limit can be determined using the ROC curve. Here, equal importance is given to both false alarms and missed alarms and the corresponding MFAR and MMAR point is determined as shown in Fig. 6.3.

6.6 Effectiveness of multivariate techniques

6.6.1 Linear System

The following equations describe the dynamics of a linear system with 10 measured variables

$$
\begin{aligned}
x_5 &= x_1 + x_2 \\
x_6 &= x_2 + x_3 \\
x_7 &= x_3 + x_4 \\
x_8 &= x_1 + x_2 + x_4 \\
x_9 &= x_2 + x_4 \\
x_{10} &= x_1 + x_2 + x_3
\end{aligned}
$$

x_1, x_2, x_3 and x_4 are independent inputs with sufficient excitation. The excitation is added in the form of uniform random disturbance with magnitude of not more than 10% of their respective nominal values. x_5, x_6, x_7, x_8, x_9 and x_{10} are linear combinations of the inputs and are the process outputs. $X_1, X_2....X_{10}$ are process measurement vectors with added measurement noise. The measurement noise is normally distributed with zero mean and standard deviation of about 1% of the average of all the process variables at normal operation. Two types of faults are considered here. The first one is a sensor bias and the second is a measured disturbance (throughput change)

Sensor bias in the linear system

The process is simulated for 10000 sampling instants. A sensor bias of magnitude 3% is introduced in the variable x_8 from sampling instant 5001. Since there are 4 independent variables in the process, a PCA model with 4 principal components (PCs) should be able to explain most of the variance in the normal operating data.

Fig. 6.4 shows the performance of three variables of interest for monitoring. They are the Q statistic using a PCA model with 4 PCs, faulty variable X_8 and filtered faulty variable X_8^*. Here, a third order moving average filter is

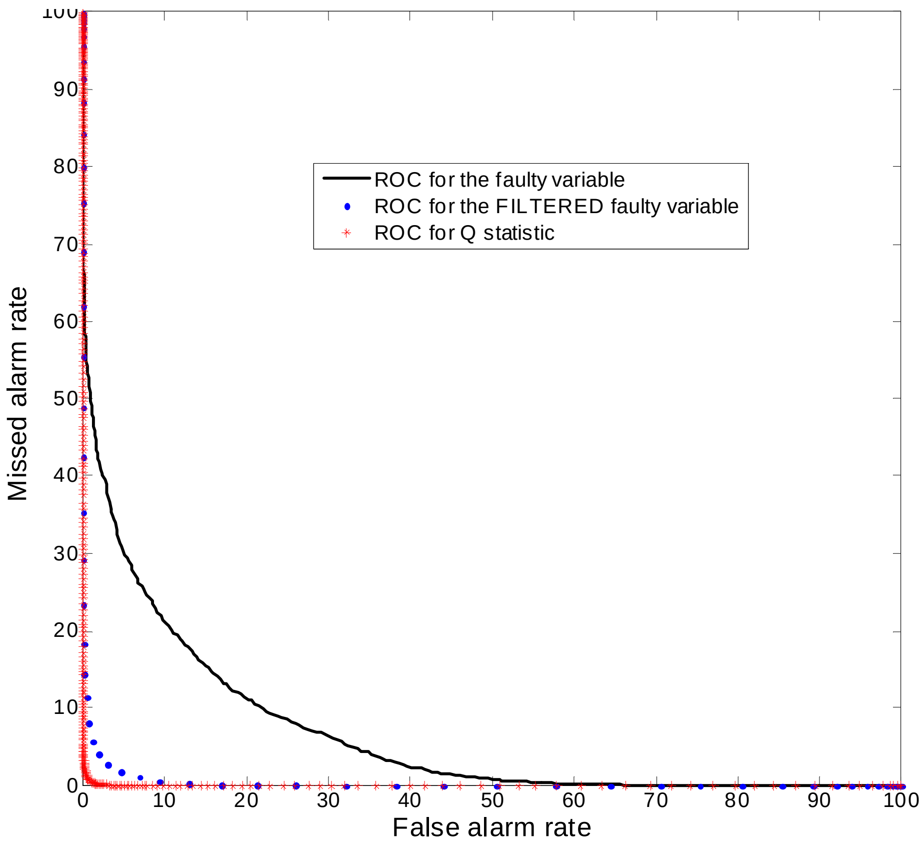


Figure 6.4: ROC curves for the linear process

used to obtain X_8^* from X_8. It is to be noted that, unlike other two variables, filtering introduces detection delay in X_8^*. The magnitude of detection delay usually depends on filter type and distributions of normal and faulty operating data. It can be seen from Fig. 6.4 that the ROC curve for Q statistic is closer to origin (FAR =0, MAR =0) than that of X_8 and X_8^*.

Fig. 6.5 shows the MFAR and MMAR values for process variables, Q statistic and T^2 statistic. The PCA is done on the normal operating data after zero centering and unit variance scaling of all the variables. For the sensor bias fault, the correlation structure is broken in the faulty samples, thus the distribution of Q statistic is expected to change as we move from normal operation data to that of faulty. The accuracy of the Q statistic depends on how well the model is represented by the chosen PCs. It is evident that the MFAR and MMAR for the Q statistic drops down significantly from the 4th PC (it is within 1% MFAR and 1% MMAR). For this linear process, a PCA model with 4 PCs should be adequate. If more PCs are considered, there is a good chance of including noise in the prediction and hence in the Q statistic. In this example, it appears that any PCA model with 4 to 8 PCs can be used to compute the Q statistic. However if the signal to noise ratio is low, it is recommended to use the exact number of latent variables (4 PCs in this case).

Throughput change in the linear system

In this case, appropriate step disturbances are given in three of the input variables (about 7% in x_1, 4% in x_3 and 3% in x_4) starting from sampling instant 5001 until 10000. As there is be no break in correlation structure, the Q statistic for the PCA model with 4 PCs do not show any change. However, the T^2 statistic with 4 PCs gives lowest values for MFAR and MMAR (see Fig. 6.6). It is to be noted that the fault magnitude is larger here compared to that in the sensor bias case.

6.6.2 Tennessee Eastman Process (TEP)

The TE process is a simulation environment of a real chemical process with masked components and dynamics. The Tennessee Eastman (TE) Plant-wide Industrial Process Control Problem was proposed in the early 90s as a challenge test problem for a number of control related topics [1]. As shown in Fig. 6.7, the TE process includes following units: a two-phase reactor, a flash separator,

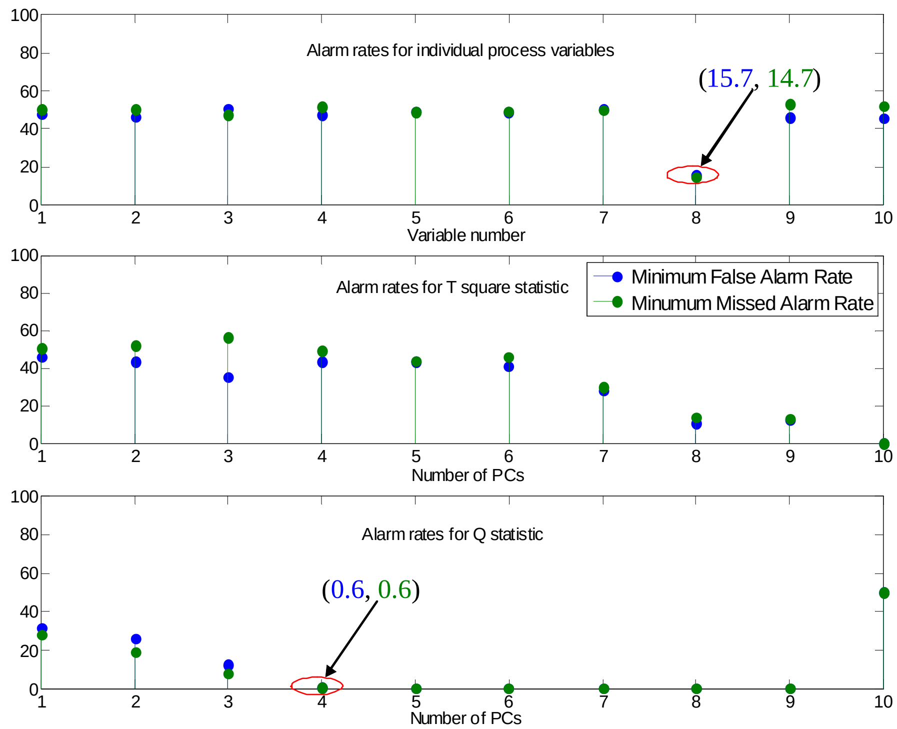


Figure 6.5: MFAR and MMAR for sensor bias case in the linear process

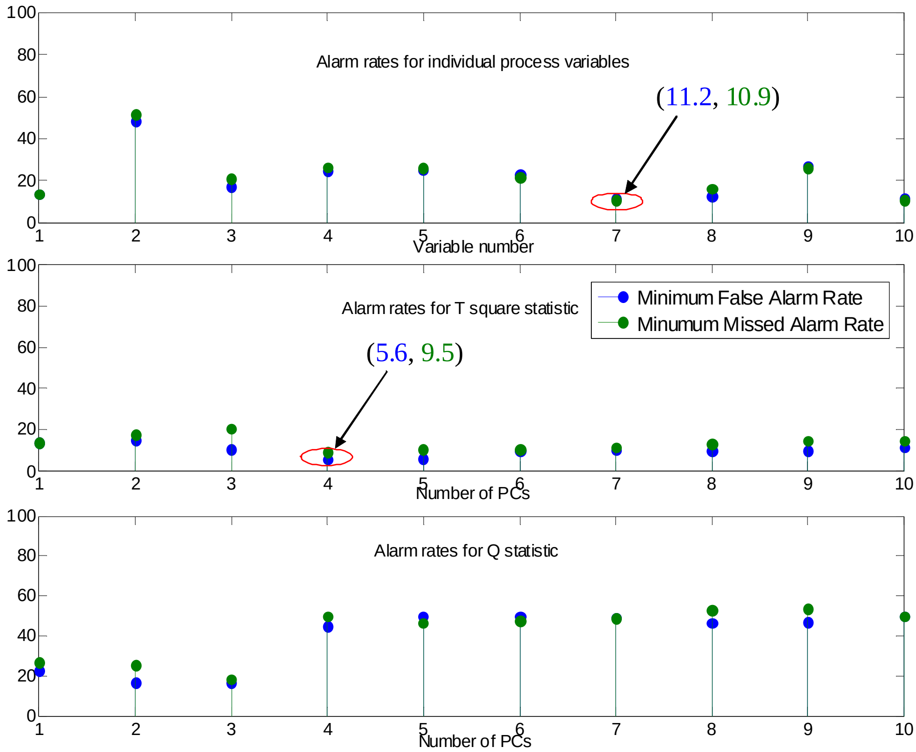


Figure 6.6: MFAR and MMAR for throughput change case in the linear process

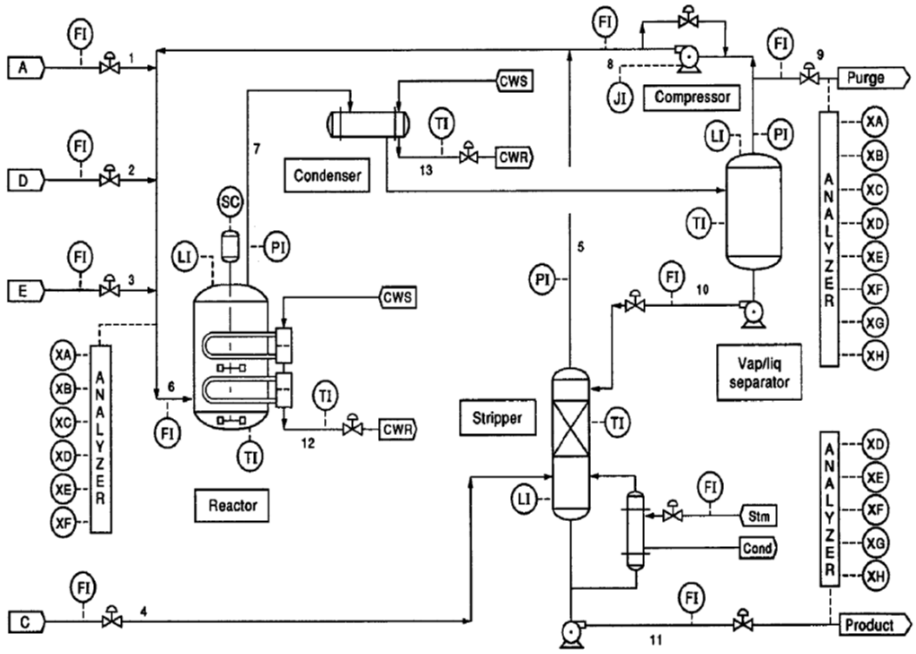


Figure 6.7: Tennessee Eastman process

reboiled stripper, recycle compressor and a condenser. There are four gaseous reactants (A, C, D, and E), two liquid products (G and H), a byproduct F and an inert B. In all there are 41 measured variables and 12 manipulated variables in the original TE process.

The open loop process as such is non-linear and highly unstable. In this work, the control structure proposed in [6] is adopted to stabilize the plant and study its response under two faulty situations. The TE simulator with control structure in place can be downloaded from `http://depts.washington.edu/control/LARRY/TE/download.html\#Multiloop`. Only 9 out of 12 manipulated variables are used in this control structure. Thus in all there are 50 (41 + 9) process variables. The faults are introduced after 36 hours of normal operation and then the process is simulated for another 36 hours. The sampling rate is 36 seconds and 7200 samples are obtained in all. Sample 1601 to sample 3600 is considered as steady state normal operating data and sample 5601 to sample 7200 is taken as steady state faulty data.

Bias in the pressure sensor

Reactor pressure (variable 7) is one of the most important variable in the TE process. According to the benchmark problem [1], a 3.4% increase in the reactor pressure from normal operating range will lead to a process shutdown. A small sensor bias of magnitude 2 kPa is introduced in variable 7 after 36 hours of normal operation. The top part in Fig. 6.8 shows the MFAR and MMAR for all the process variables. Variable 7 shows no significant change from normal operation and faulty operation. This is due to the fact that the sensor bias is compensated in this case due to the presence of various controllers in place. However, the fault is propagated and there is a significant change in separator pressure (variable 13) and stripper pressure(variable 16). These two variables are not controlled and the change in these two variables is a direct consequence of sensor bias in the reactor pressure sensor. The bottom stem plot in Fig. 6.8 shows that this sensor bias fault can be detected with lower MFAR and MMAR using anywhere between 27 and 37 PCs.

Product leak

The second fault considered is the stripper underflow (variable 17) leakage. The product flow is also a controlled variable thus we do not expect any change in its value from normal operation to the faulty operation. The top stem plot in Fig. 6.9 shows the MFAR and MMAR for all the 50 variables in the process. Variables 42 (D feed), 45 (A&C feed), 48 (Stripper underflow

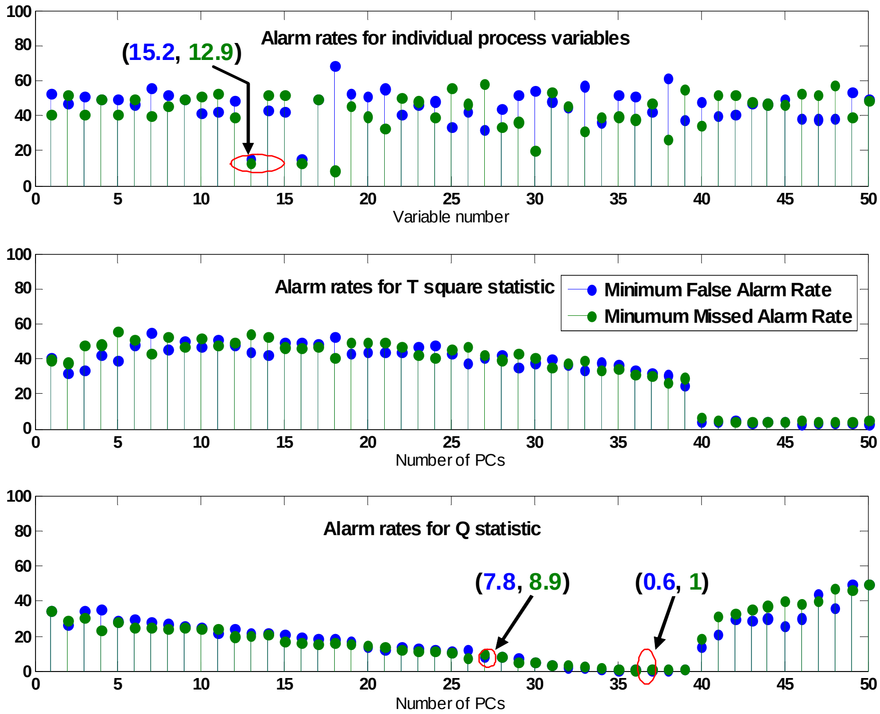


Figure 6.8: MFAR and MMAR for pressure sensor bias case in TE process

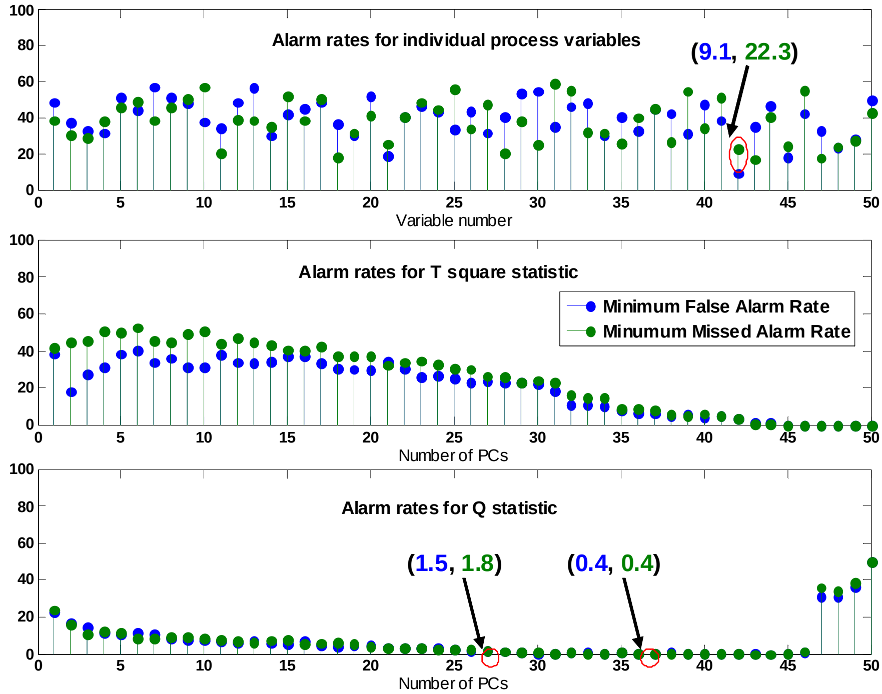


Figure 6.9: MFAR and MMAR for product leak case in TE process

valve) show significant change in their distribution. It is to be noted that all these variables are manipulated variables for the process and they increase in order to compensate for the product loss.

Here again, the bottom stem plot (for Q statistic) in Fig. 6.9 shows relatively low values for MMAR and MFAR. PCA model with around 27 PCs gives good results and as we move from 27 to 37 PCs, the Q statistic gives MFAR and MMAR values within the 1% level which is much better compared to monitoring any of the individual process variables.

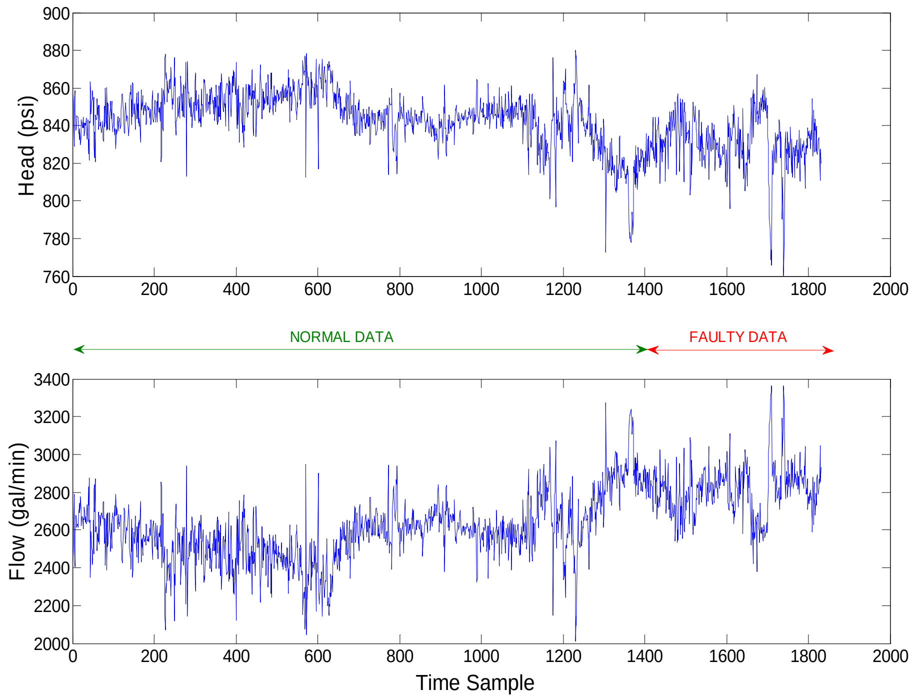


Figure 6.10: 30 hours of Pump head and flowrate data sampled once every one minute showing sections of normal and faulty operation

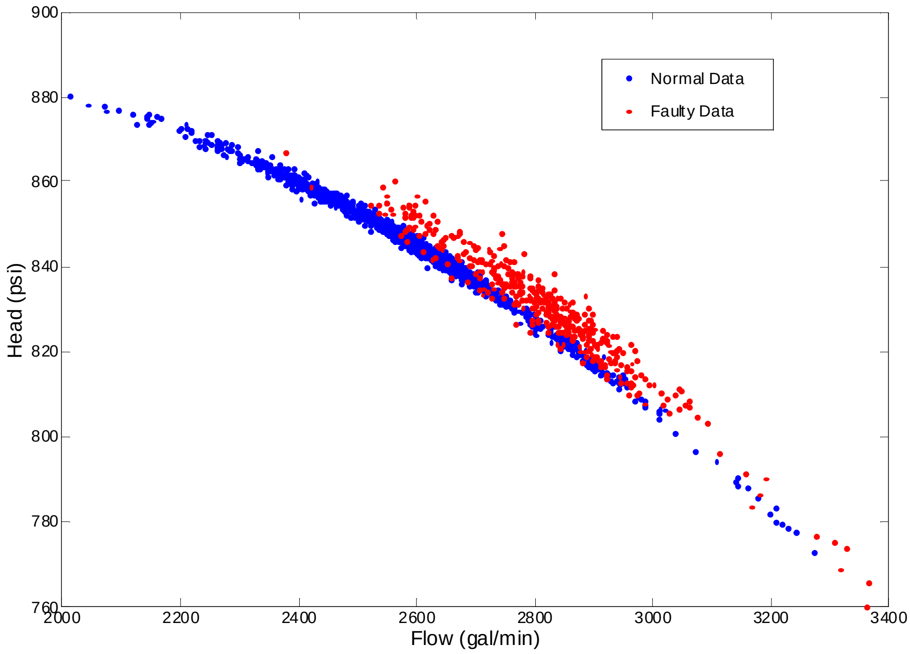


Figure 6.11: Pump head versus flowrate data showing normal and faulty data

6.6.3 Industrial case study (Pump example)

A heavy duty centrifugal pump is used to transfer sea water to an oil processing facility. Output flow rate (gal/min) and the head pressure (psi) are measured every minute to monitor the performance of the pump. Data is available for about 30 hours of operation (1831 samples) that includes normal and faulty periods of operation. The time trend of both output flow rate and pump head are shown in figure 6.10. It is also known that a fault od small magnitude occurred at about 1401^{th} sampling instant thereby throwing the pump behavior our of its typical flow curve. However, it is not straightforward to make out what instant the fault has occurred by looking at the individual trends of either the pump head or the flow rate. Figure 6.10 also shows many instances during the normal operation where the data is non-stationary.

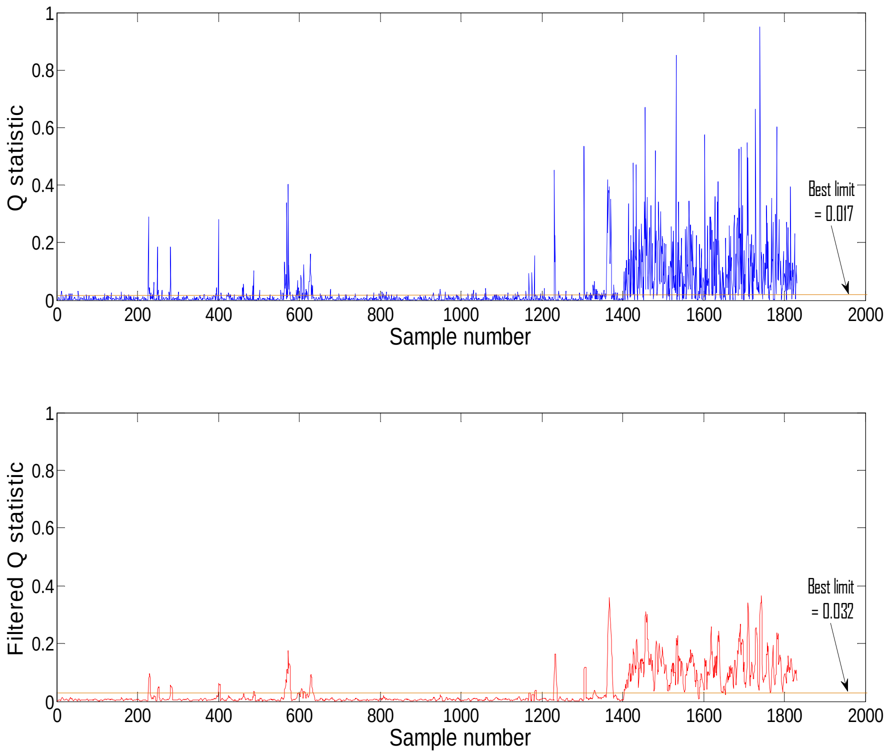


Figure 6.12: Time series plot of Q and filtered Q statistic

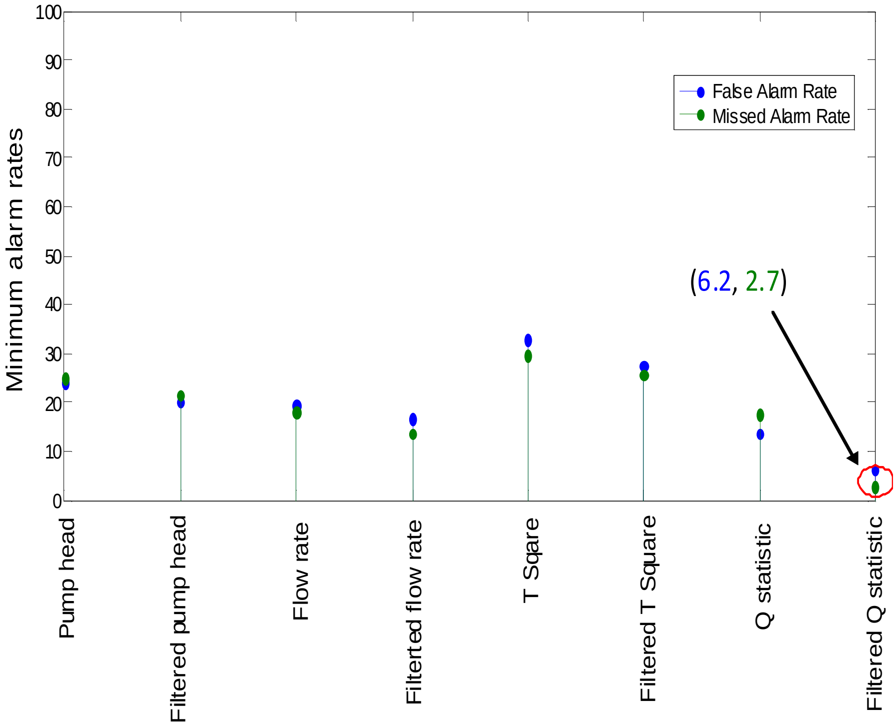


Figure 6.13: MFAR and MMAR for pump example

Figure 6.11 shows the pump head as a function of the flow rate. There is clearly a quadratic relationship between the pump head and the flowrate through the pump. Figure 6.11 also shows the faulty data starting from sampling instant 1401 that seems to deviate from the quadratic relationship. A PCA model is built using the normal data (from 1 to 1400 samples) comprising of pump head, flow rate and the square of flow rate. Since there is just one independent variable in this system, one principal component must be able to explain most of the variance. In this case, the first principal component explains 99.58% variance in the data.

Using only the first principal component, the T^2 and Q statistic are calculated. Figure 6.13 shows the MFAR and MMAR when monitoring a variety of variables such as pump head, filtered pump head, flow rate, filtered flow rate, T^2 statistic, filtered T^2 statistic, Q statistic and filtered Q statistic. A fifth order moving average filter is used on all the variables for this purpose.

In this case study, as the fault lead to the pump operating outside its curve leading to break in correlation structure between the involved process variables, the Q statistic and its filtered version showed the lowest MFAR and MMAR. Figure 6.12 shows the Q statistic and its filtered version over time.

Instead of using a PCA model, the same data in the normal operation can be modeled by fitting a quadratic equation in the least squares sense as follows

$$H(Q) = -3.7396 * 10^{-5} Q^2 + 0.1123Q + 806.1036 \tag{6.9}$$

where H is the pump head and Q is the flow rate through the pump. Using the relation, the head can be estimated ($\hat{H}$) given a flow rate. The difference between the actual head and the estimated head, also called as the prediction error (e) can be used to monitor the break in correlation between these

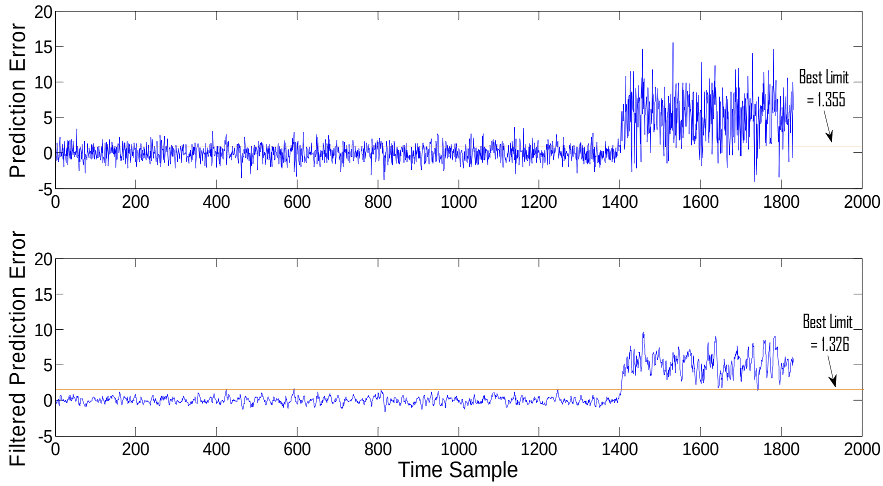


Figure 6.14: Time series plot of prediction error and filtered prediction error showing the best threshold limit

variables. For any sample k, the prediction error is written as

$$e_k = H_k - \hat{H}_k \tag{6.10}$$

Figure 6.14 shows the time series plot of prediction error and its filtered version (fifth order moving average filter). The prediction error gave a MFAR and MMAR of 8.5% and 11.3% respectively where as the filtered prediction error gave a much lower MFAR and MMAR of 0.28 % and 0.69% respectively.

During the time period of consideration, the filtered prediction error based on the fitted quadratic model gave much less MFAR and MMAR compared to what filtered version of PCA based Q statistic could achieve. This difference is mainly due to the fact that the raw data used to build the PCA model is non-stationary and as a result lead to more false alarms and missed alarms especially during the instants when the process showed the non-stationary behavior.

6.7 Concluding remarks

As complexity in process plants increase, univariate process monitoring becomes a Herculean task. It is necessary to switch to multivariate alarms which appear to take advantage of analytical redundancy in the process networks. ROC curves are useful for visualizing the performance of alarm limits and are particularly appreciated in cases with skewed class distributions and unequal classification error costs. The PCA based T^2 and Q statistic incorporate information from several variables simultaneously to flag a fault and thus are more efficient compared to univariate alarming techniques. The two simulation case studies, first one the linear process and second the TE benchmark process and a simple industrial case study involving a pump illustrated the efficiency of PCA based T^2 and Q statistic in minimizing nuisance alarms. Monitoring filtered variables is another way to reduce the false alarms and missed alarms but filtering introduces detection delay. Once a a multivariate alarm is annunciated, the contribution plots [5] can be used to pinpoint the variables responsible for the fault.

Bibliography

[1] J. J. Downs and E. F. Vogel, *A plant-wide industrial process control problem*, Computers & Chemical Engineering **17** (1993), no. 3, 245–255.

[2] T. Fawcett, *An introduction to roc analysis*, Pattern Recognition Letters **27** (2006), no. 8, 861–874.

[3] D. M. Himmelblau, *Fault detection and diagnosis in chemical and petrochemical processes*, Elsevier press, Amsterdam, 1978.

[4] J. F. MacGregor and T. Kourti, *Statistical process control of multivariate processes*, Control Engineering Practice **3** (1995), no. 3, 403–414.

[5] S. J. Qin, *Statistical process monitoring: Basics and beyond*, Journal of Chemometrics **17** (2003), no. 8-9, 480–502.

[6] N. L. Ricker, *Decentralized control of the tennessee eastman challenge process*, Journal of Process Control **6** (1996), no. 4, 205–221.

[7] E. L. Russel, L. H. Chiang, and R. D. Braatz, *Data-driven techiniques for fault detection and diagnosis in chemical processes*, Springer-Verlag, London, 2000.

[8] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri, *A review of process fault detection and diagnosis part i: Quantitative model-based methods*, Computers & Chemical Engineering **27** (2003), no. 3, 293–311.

[9] S. Wold, K. Esbensen, and P. Geladi, *Principal component analysis*, Chemometrics and Intelligent Laboratory Systems **2** (1987), no. 1-3, 37–52.

7 Concluding remarks

The objectives of the work reported in this thesis are to develop tools for advanced analysis and redesign of industrial alarm systems. In particular the work has focused on developing simple and effective tools for assessment of alarm systems based on routinely collected alarm event data. For univariate alarm design, two commonly used techniques, delay timers and latches are compared for performance on the ROC framework. Utility of Principal Components Analysis based multivariate techniques is illustrated using simulated examples on the same framework.

7.1 Main contributions of this thesis

Chapter 2 provides a tutorial introduction to alarm management including definition of terms used in context. Engineering aspects of alarm management are discussed in its life cycle approach. The importance of good alarm system design and maintenance is emphasized.

Binary sequence representation of alarm data proposed in Chapter 3 facilitates advanced analysis of alarm events. Two graphical tools that are specially designed to efficiently identify nuisance alarms are presented and their characteristics are discussed using two industrial datasets. The three dimensional HDAP encapsulates the information from alarm data for a given period of time. HDAP not only shows the progression of top alarms with time but also highlights apparently redundant and chattering alarms. A similarity measure is proposed for this application and justified through its physical interpretation. ASCM shows the similarity measure between rearranged unique alarms in a color coded matrix format and is useful for identifying groups of related alarms providing insights into process interactions. These two graphical representations of the alarm data provide quick and valuable feedback to make improvements to the alarm system in several other steps in the alarm management lifecycle and contribute to reduction in nuisance alarms.

In chapter 4, an index to measure the amount of chatter in alarms is proposed based on run-length distributions using only the alarm data which is readily available in historical databases. A variant of the chatter index, with flexible assumptions is also proposed and compared in closeness to the original chatter index for practical purposes. Improvements in chatter index are illustrated using industrial data after making appropriate alarm design changes. Chatter index can be calculated automatically given the alarm data for an alarm over a period of time and hence reduces the effort required for identifying top chattering alarms as part of routine assessment of alarm systems. A limit on the proposed chatter index has been calculated to identify the worst chattering alarms based on a rule of thumb.

In Chapter 5, the performance of two commonly used techniques to reduce nuisance alarms, namely, delay timers and latches is discussed. The performance of these techniques in terms of detection accuracy is discussed in the ROC framework by modeling them using Markov chains. ROC curves are useful for visualizing the performance of alarm limits and are particularly appreciated in cases with skewed class distributions and unequal classification error costs. It has been shown that the performance of combined delay timers is much better than the pure delay timers of the same length. Alarm latches do not perform anywhere close to delay timers when it comes to detection accuracy but they may be used to monitor critical variables without detection delay and chattering problems. A design methodology using historical alarm and Return To Normal (RTN) data to select the type and length of delay timers is illustrated using a real industrial case study. The tools for advanced analysis proposed in Chapter 3 and for redesign in chapter 5, serve as excellent additions to control engineers toolkit by helping them identify nuisance alarms and contribute to improving the alarm system by making sure each alarm has a purpose and is activated in a timely manner.

Chapter 6 probes into evaluating efficiency of multivariate alarming techniques. As complexity in process plants increase, univariate process monitoring becomes a herculean task. It is necessary to switch to multivariate alarms which appear to take advantage of analytical redundancy in the process networks. The PCA based T^2 and Q statistic incorporate information from several variables simultaneously to flag a fault and thus are more efficient compared to univariate alarming techniques. Two simulation case studies, first one a linear process and second one on the TE benchmark process and a simple industrial case study illustrated the efficiency of PCA based T^2 and Q statistic in minimizing nuisance alarms.

7.2 On going work and recommendations for future work

The area of alarm management is relatively new to academia and at the time this thesis is being written, there has been some parallel work which is related to the work outlined in this thesis. This section focuses on acknowledging some of the related work and suggestions for future research directions.

Modern day DCSs have capabilities to log not only alarm events but also other events such as Return To Normal (RTN), Operator Interventions and Operator Acknowledgements. Using the methodology adopted in this work, these events can also be represented as binary sequences. Binary data representation facilitates further analysis of these events to extract knowledge about similarities in event occurrences. This opens up pathways to develop novel strategies to extract valuable information from vast event bases which often are used only during abnormal event investigations. Such a knowledge extraction would be a big leap forward in improving process operations from being more or less reactive to proactive. Studies on alarm system performance, plant response, root cause analysis and operator behavior can be performed. On similar lines, work presented in [2] focuses on identifying alarm floods with and without chattering alarms and comparing them for similarity. As an improvement to the temporal similarity measure based on binary sequence representation proposed in this thesis, [5] has put forward a method to generate pseudo continuous time series based on Gaussian kernel method to minimize the influence of false, missed and chattering alarms.

The chatter indices proposed in this work can be used in optimal design of a suitable filter in order to minimize chattering. The work presented in [3] proposed a method to estimate the chattering index based on statistical properties of the process variable as well as alarm design parameters. Such an estimation would be useful to optimally design various alarm generation techniques.

An important aspect of alarm generation that was not touched upon in this thesis is the detection latency that is introduced due to various filtering techniques. The work presented in [1] provides a means to calculate the detection delay for two commonly used design techniques, deadbands and delay timers.

There is an abundance of process data based multivariate techniques for fault detection [4] but no single method has all the desirable features. It would be a good exercise to evaluate the performance of various diagnostic techniques in the ROC framework. Another area of research would be developing guidelines or at least showcasing case studies where intelligent alarming is adopted for fault specific triggering of multivariate alarms while suppressing unimportant nuisance alarms.

Bibliography

[1] N. A. Adnan, I. Izadi, and T. Chen, *On expected detection delays for alarm systems with deadbands and delay-timers*, Journal of Process Control **21** (2011), no. 9, 1318 – 1331.

[2] K. Ahmed, *Similarity analysis of industrial alarm flood data*, Master's thesis, University of Alberta, 2011.

[3] E. Naghoosi, I. Izadi, and T. Chen, *Estimation of alarm chattering*, Journal of Process Control **21** (2011), no. 9, 1243 – 1249.

[4] V. Venkatasubramanian, R. Rengaswamy, S. N. Kavuri, and K. Yin, *A review of process fault detection and diagnosis: Part iii: Process history*

based methods, Computers & Chemical Engineering **27** (2003), no. 3, 327 – 346.

[5] F. Yang, S. L. Shah, D. Xiao, and T. Chen, *Improved correlation analysis and visualization of industrial alarm data*, ISA Transactions **51** (2012), no. 4, 499 – 506.

A

Quantification of alarm chatter

A.1 Theoretical bounds on Ψ

$$r \geq 1 \;\forall\;\; r \in \mathbb{N}$$
$$\Rightarrow 0 < \frac{1}{r} \leq 1 \;\forall\;\; r \in \mathbb{N}$$

Multiplying with $P_r \geq 0$,

$$\Rightarrow 0 \le \frac{P_r}{r} \le P_r \ \forall \ \ r \in \mathbb{N}$$
$$\Rightarrow \sum_{r \in \mathbb{N}} 0 \le \sum_{r \in \mathbb{N}} \frac{P_r}{r} \le \sum_{r \in \mathbb{N}} P_r$$

But $\sum_{r \in \mathbb{N}} \frac{P_r}{r} = \Psi$ and $\sum_{r \in \mathbb{N}} P_r = 1$ according to definition

$$\Rightarrow 0 \le \Psi \le 1$$

A.2 Fictitious example of a chattering alarm

Assume a unique alarm which has a 10 second reset (like a failed pump which is a system alarm). It means that whenever an abnormal event occurs, the alarm rings once every 10 seconds throughout the duration of the abnormal event. Also assume that the abnormal event lasts for 10 minutes each time and this event happens once every 8 hours (operator shift duration). Assuming alarm data for a duration of one week is available,

$$P_{r=10} = \frac{(60-1)*3*7}{60*3*7-1} \text{ and}$$
$$P_{r=(8*60*60-(9*60+50))} = \frac{7*3-1}{60*3*7-1}$$

For all other r, $P_r = 0$. Then,

$$\Psi = \frac{P_{r=10}}{10} + \frac{P_{r=28210}}{28210} = 0.0984 \approx 0.1$$

In this fictitious example, the calculated Ψ is very close to the theoretical value of the frequency of alarm occurrence during the abnormal event (1 alarm in 10 seconds or 0.1 alarms/second).

A.3 Proof that $\Psi_{\tau_1} \geq \Psi_{\tau_2} \ \forall \ \ \tau_1 < \tau_2$

Consider a unique alarm with alarm counts in the RLD represented by AC_r. For a finite time τ,

$$\text{if } \tau_1 < \tau_2,$$

then,

$$\sum_{r=1}^{r=\tau_2} AC_r \geq \sum_{r=1}^{r=\tau_1} AC_r$$

Since $AC_r \geq 0 \ \forall \ r \in \mathbb{N}$,

$$\frac{AC_r}{\sum_{r=1}^{r=\tau_1} AC_r} \geq \frac{AC_r}{\sum_{r=1}^{r=\tau_2} AC_r} \ \forall \ r \in \mathbb{N}$$
$$\Rightarrow P_{r,\tau_1} \geq P_{r,\tau_2} \forall \ r \in \mathbb{N}$$

Dividing both sides with r and summing up over all $r \in \mathbb{N}$,

$$\sum_{r \in \mathbb{N}} P_{r,\tau_1} \frac{1}{r} \geq \sum_{r \in \mathbb{N}} P_{r,\tau_2} \frac{1}{r}$$
$$\Psi_{\tau_1} \geq \Psi_{\tau_2}$$

Hence it has been proved that $\Psi_{\tau_1} \geq \Psi_{\tau_2} \ \forall \ \ \tau_1 < \tau_2$

A.4 Non-uniqueness of Ψ

Given

$$P_{1,r} \geq 0 \ \forall \ r \in \mathbb{N},$$
$$P_{2,r} \geq 0 \ \forall \ r \in \mathbb{N},$$
$$\sum_{r \in \mathbb{N}} P_{1,r} = 1,$$
$$\sum_{r \in \mathbb{N}} P_{2,r} = 1 \text{ and}$$
$$\sum_{r \in \mathbb{N}} \frac{(P_{2,r} - P_{1,r})}{r} = 0$$

Can we prove

$$P_{1,r} = P_{2,r} \ \forall \ r \in \mathbb{N} \ ?$$

No, it can be disproved. Consider a counter example where
$P_{1,10} = 1$ and $P_{1,r} = 0 \ \forall \ r \in \mathbb{N} - \{10\}$, and
$P_{2,5} = 0.25, P_{2,15} = 0.75$ and $P_{2,r} = 0 \ \forall \ r \in \mathbb{N} - \{5, 15\}$
The above mentioned counter example shows two different probability distributions, P_1 and P_2 that satisfy all the given conditions.