Imperial College London
Department of Chemical Engineering

Kernel Methods for Monitoring Processes with Multiple Operating Modes

by Ruomu Tan

September, 2020

Supervisors: Prof. Nina F. Thornhill
Dr. James R. Ottewill
Prof. Yi Cao

Submitted in part fulfilment of the requirements for the degree of
Doctor of Philosophy in Chemical Engineering of
Imperial College London

Declaration of copyright

Declaration of originality

I hereby confirm that the works reported in the thesis are research outcomes of my PhD project. My contributions have been clearly stated for those works that were done in collaboration with other researchers. Any other work has been properly referenced.

Abstract

Systems for monitoring processes with multiple operating modes should be able to distinguish between changes in operating modes and developing faults. Process operators will make decisions regarding production and maintenance according to the information about faults, such as the occurrence of faults and the locations of faults, so that the process can run safely and efficiently. Whilst the development and application of kernel methods can improve the performance of monitoring systems, inappropriate usage of these methods can diminish the effectiveness of the methods.

In this thesis the industrial considerations of operators are summarized and these considerations are incorporated into the development of kernel methods for process monitoring. The research in the thesis shows that kernel methods need to be designed and implemented properly for monitoring processes with multiple operating modes. The research in the thesis also aims to develop kernel methods that can generate useful results when applied to monitoring of processes with multiple operating modes.

The thesis reports the following research outcomes:

- A benchmark multimodal dataset from a pilot-scale experiment rig;
- An investigation of the tuning of kernel methods and a tuning strategy for the radial basis function kernel;
- A new kernel that can improve the monitoring performance when applied to multimodal data;
- An on-line monitoring framework which can account for new operating modes in the process;
- A way to define the contributions of process variables to a fault detection, in order to support fault diagnosis.

The thesis delivers novel kernel methods for monitoring processes with multiple operating modes and gives guidelines for proper implementation of these methods. These outcomes extend the field of process monitoring. The research outcomes are relevant for industrial application because the practical considerations of end-users are incorporated in the development of the kernel methods. The thesis also contributes to the theory of process monitoring by proposing novel kernel methods for fault detection and diagnosis.

The results in the thesis demonstrate that the new development of kernel methods can improve monitoring performance when applied to processes with multiple operating modes.

Moreover, the monitoring results achieved by the new kernel methods can be interpreted and used by process operators.

Acknowledgements

First and foremost, I would like to express my wholehearted gratefulness to my PhD supervisor, Prof. Nina Thornhill. The thesis and many opportunities I seized in the past four years would not have existed without Nina's help. I have learnt many things from Nina and there will surely be many more to learn, but the most important thing, in my opinion, is that good researchers do innovative research with practical impact and support their colleagues, especially the younger generations.

I would also like to cordially thank my industrial supervisor, Dr. James Ottewill. James helped me understand and formulate the real-life problems that exist in industry and these problems then became the foundation of the research in the thesis. In research, it is impressive that James always makes the industrial relevance a highlight whilst being sharp and precise about the tiniest technical details.

With the help of Nina and James, I am pleased to find out that real-life problems in industry can lead to new advances in the theory. I shall always remember the time I spent joyfully discussing with Nina and James and I wish this to continue.

The two-year experience with ABB PLCRC was amazing. The help from Dr. Piotr Lipnicki and Dr. Marcin Firla for designing and conducting the compressor experiment is gratefully acknowledged. Apart from that, I have been supported by ABB PLCRC in many other ways. This experience motivated me to join ABB and I am grateful that my colleagues at DECRC have been considerate when I am finishing off this thesis in the past three months.

My sincere gratitude also goes to Dr. Yi Cao, who supervised me for the first half of my PhD that I spent at Cranfield University. Thank you Yi for having me on board and for helping me to make my PhD study fruitful. In particular, the PRONTO benchmark case study would not have been possible without the support from Yi and other colleagues of the Process Systems Engineering lab. It is a pleasure that we have stayed in touch and I hope there will be opportunities for us to collaborate in the future.

I am grateful to the administrative colleagues at Imperial, PLCRC, and Cranfield. They went out their way to help me dealing with all the administrative issues that I have encountered. Their support made my life much easier.

The financial support from the EU H2020 project PRONTO is sincerely acknowledged. It is a great pleasure to be part of the project. The project provides an opportunity of pursuing a PhD degree with significant industrial relevance and a decent salary. Moreover, the consortium is an incubator of new ideas and collaborations between researchers. In particular, I really enjoyed the time I spent with Dr. Anna Stief in Cranfield and in Kraków. I also want

to thank Dr. Marta Zagorówska and Francesco Borghesan for making my time at Imperial enjoyable. The project runs only for several years, but the collaboration, the friendship, and the willingness to support each other cultivated in the project continue beyond it. At present, this is particularly valuable during such an extraordinary period of time.

I would like to express my deepest gratitude to my parents for their unwavering trust and love which have been and will always be motivating for me. I also thank myself in the past, for making the right decision to take the leap.

It was precisely because the old theory gave results that were fairly good by the measurement standards of the time that kept it in being so long.

Isaac Asimov, The Relativity of Wrong.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This chapter gives the background of the research and outlines the research questions that will be answered by the research presented in the rest of the thesis. The contents of the chapters are introduced. The chapter summarizes the contributions and the publications generated by the research presented in the thesis.

1.1 Process monitoring for decision support

This section briefly introduces the existence of multiple operating modes in process operations, followed by the challenges in process monitoring posed by multiple operating modes. The section then outlines the research questions that the rest of the thesis will answer.

1.1.1 Multiple operating modes

The typical practice in process industries is that human operators, with the assistance of digital control systems, operate a process to meet the requirements of production whilst avoiding potential hazards. The operators make a variety of decisions and take actions accordingly. Such decisions might include what a new set-point of a controlled variable should be, how to schedule the production and when to carry out inspections and maintenance. In particular, the operators might initiate a start-up or shut-down of the process,

change conditions to produce a different grade of product, or increase or decrease the production rate to meet market demand. Each of these operator actions varies the production regime and the varying production regime results in multiple operating modes of a process. On the other hand, changes in the process may not always be due to the decisions made by the operators. For example, fluctuations occurring either upstream or downstream of the process may also result in changes in the production regime of the current process. Changes in the process may also be indicative of adverse operation and potentially the occurrence of faults and failures. For example, a leakage may occur in a pipeline of a gas transportation process, resulting in reduced outlet gas flow rates. The process operators need to be aware of such unwanted changes so that they can make decisions to improve the situation. Therefore, operators must know about the true process performance in order to make correct decisions.

Process monitoring is the activity of inspecting information collected during process operations, such as the measurements of process variables, to determine if any unwanted change has occurred in the process. Data-driven process monitoring methods compare the current operation of the process against data recorded from previous, historical operations, which are collected when the performance of the process is known.

Data-driven process monitoring methods can support operators by detecting abnormal behaviour and drawing it to their attention. Multiple operating modes pose a challenge in data-driven process monitoring. It is necessary for a monitoring technique to distinguish between the changes which occur when the process has moved to a new production regime and the changes due to degraded process performance. The reason is that the process operators will make different decisions when facing the two types of change. The operators will acknowledge the first type of change as a healthy process behaviour, whereas they will take actions to prevent the second type of change developing further into a significant failure. Furthermore, it is often observed in processes that early-stage degradation may only cause small variations in the process performance while the difference between two operating modes may be more substantial. It may be difficult to distinguish the small variations which might be indicative of failures from the variations due to changes between multiple operating modes. Therefore, a monitoring approach should identify the performance degradation without reacting to the change in production regimes.

The aim of the research reported in this thesis is to develop data-driven monitoring methods which can cope with the challenge posed by multiple, varying operating modes. These

methods can provide better decision-support to operators by achieving a more accurate evaluation of the process performance. Firstly, practical considerations, which describe what information is required to support an operator, are characterised in the thesis. The challenges caused by multiple operating modes will be demonstrated using real-life case studies. Then monitoring methods are developed with respect to the practical considerations.

1.1.2 Challenges in kernel methods

Kernel methods are a category of data analytics methods which apply kernel transformation to nonlinear data such that linear methods can be applied to the transformed data. The thesis will explore kernel methods in data-driven process monitoring for addressing the challenges of providing operator decision support across multiple operating modes. Kernel methods can describe complex relationships in process data. However, the literature study in Chapter 3 uncovers significant problems showing that kernel methods need to be investigated further in order to be useful for accounting for multiple operating modes. The main problem is the tuning of kernels which influences the monitoring results. Kernel methods may also output monitoring results that are not easily understood by operators. Moreover, existing kernels have limitations in describing relationships in process data from multiple operating modes.

The thesis aims to make significant theoretical contributions towards solving the problems that arise when kernel methods are applied to the monitoring of processes with varying production regimes. A new kernel will be developed so that the kernel methods can distinguish between changes in operating modes and changes due to performance degradation. To summarize, the thesis will develop new kernel-based monitoring methods that can achieve desired monitoring results for processes with multiple operating modes. These methods can better support process operators for decision making. Moreover, the PhD work presented in the thesis will provide novel insight into the kernel methods themselves, including recommendations on their proper usage.

1.1.3 Research questions

The following four research questions need to be answered in order to develop kernel methods for monitoring processes with multiple operating modes:

1. What information does an operator need from a monitoring method in order to make decisions related to the operation of a process?
2. Are kernel methods suitable for monitoring processes with multiple operating modes?
3. How should kernel methods be configured so that they perform properly when applied to monitoring processes with multiple operating modes?
4. How can kernel methods form part of a monitoring system?

The four research questions will be addressed in the thesis.

1.2 Introduction to the thesis

The thesis has nine chapters. Chapter 1 introduces the topic of the research in the thesis and the structure of the thesis. Chapter 2 presents the background of the research topic, summarizes the practical requirements when developing monitoring methods for use in industry, and identifies challenges in multimodal process monitoring. The work in Chapter 2 addresses the research question regarding what information an operator needs from a monitoring method. Chapter 3 reviews the state-of-the-art in process monitoring and identifies open questions in kernel methods. The literature review answers the research question of whether kernel methods are suitable for monitoring processes with multiple operating modes and partially describes how kernel methods should be configured.

Chapters 4 to 7 are the technical chapters. Chapter 4 presents an experimental benchmark dataset for validating data-driven monitoring methods. The insights from this dataset also contribute to answering the research question regarding how kernel methods should be configured properly. The dataset is also used to verify if the kernel methods are configured properly and if these methods can generate results that are useful for an operator.

Chapter 5 investigates the tuning of kernel methods and the behaviour of kernel methods when used for process monitoring. This answers the question regarding how kernel methods should be configured.

Chapter 6 demonstrates that existing kernel methods may not be sufficient for monitoring processes with multiple operating modes and proposes a new kernel that is suitable for multimodal process monitoring. This chapter also answers the question of how kernel methods can form part of a monitoring system by proposing an on-line monitoring framework using the new kernel.

Chapter 7 proposes a new definition of contributions of process variables for diagnosing process faults. This work allows kernel methods to form part of a monitoring system by enabling kernel-based fault diagnosis.

Chapter 8 gives a critical evaluation of the research presented in the thesis and outlines directions for future research. Chapter 9 summarizes the whole thesis and provides conclusions.

1.3 Contributions and publications

The research outcomes presented in the thesis have been published in peer-reviewed journal publications and international conferences. The publications and presentations are listed as follows.

1.3.1 Journal articles

- Ruomu Tan, James R. Ottewill, and Nina F. Thornhill. Non-stationary discrete convolution kernel for multimodal process monitoring. *IEEE Transactions on Neural Network and Learning Systems*, (Early Access), 2019. doi: `https://doi.org/10.1109/TNNLS.2019.2945847`.

- Ruomu Tan, Tian Cong, James R. Ottewill, Nina F. Thornhill, and Jerzy Baranowski. An on-line framework for monitoring nonlinear processes with multiple operating modes. *Journal of Process Control*, Volume 89, 2020, Pages 119-130. doi: `https://doi.org/10.1016/j.jprocont.2020.03.006`.

- Ruomu Tan, James R. Ottewill, and Nina F. Thornhill. Monitoring statistics and tuning of kernel principal component analysis with radial basis function kernels for anomaly detection. *IEEE Access*, [Submitted]

- Ruomu Tan and Yi Cao. Deviation contribution plots of multivariate statistics. *IEEE Transactions on Industrial Informatics*, Volume 5, Issue 2, 2019, Pages 833-841. doi: https://doi.org/10.1109/TII.2018.2841658.

- Anna Stief, Ruomu Tan, Yi Cao, James R. Ottewill, Nina F. Thornhill, and Jerzy Baranowski. A heterogeneous benchmark dataset for data analytics: Multiphase flow facility case study. *Journal of Process Control*, Volume 79, 2019, Pages 41-55. doi: https://doi.org/10.1016/j.jprocont.2019.04.009.

- Ruomu Tan and Yi Cao. Multi-layer contribution propagation analysis for fault diagnosis. *International Journal of Automation and Computing*, Volume 16, Issue 1, 2019, Page 40-51. doi: https://doi.org/10.1007/s11633-018-1142-y.

1.3.2 Conference proceedings

- Ruomu Tan, Tian Cong, James R. Ottewill, Nina F. Thornhill, and Jerzy Baranowski. Statistical monitoring of processes with multiple operating modes. *12th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems (DYCOPS2019)*, *IFAC-PapersOnLine*, Volume 52, Issue 1, 2019, Page 635-642. doi: https://doi.org/10.1016/j.ifacol.2019.06.134.

- Anna Stief, Ruomu Tan, Yi Cao, and James R. Ottewill. Analytics of heterogeneous process data: multiphase flow facility case study, *10th IFAC Symposium on Advanced Control of Chemical Processes (ADCHEM2018)*, *IFAC-PapersOnLine*, Volume 51, Issue 18, 2018, Pages 363-368. doi: https://doi.org/10.1016/j.ifacol.2018.09.327.

- Ruomu Tan, Raphael T. Samuel, and Yi Cao. Nonlinear dynamic process monitoring: the case study of a multiphase flow facility. *Computer Aided Chemical Engineering*, Volume 40, 2017, Pages 1495-1500. doi: https://doi.org/10.1016/B978-0-444-63965-3.50251-8.

- Ruomu Tan and Yi Cao. Contribution plots-based fault diagnosis of a multiphase flow facility with PCA-enhanced canonical variate analysis, *23rd International Conference on Automation and Computing (ICAC'17)*, Huddersfield, UK, Sep 7-8, 2017. doi: https://doi.org/10.23919/IConAC.2017.8081992.

1.3.3 Other presentations

- Ruomu Tan. Developing data analytic techniques for monitoring industrial processes. In *STEM for Britain 2019*, House of Parliament, London, UK, March 2019.

All publications generated from the PhD work in the thesis have open access. The paper entitled "Statistical monitoring of processes with multiple operating modes" has been nominated as the finalist for the Young Author Award in DYCOPS2019. The paper entitled "An on-line framework for monitoring nonlinear processes with multiple operating modes" is an invited extension of this conference paper. The paper entitled "Contribution plots-based fault diagnosis of a multiphase flow facility with PCA-enhanced canonical variate analysis" received the Best Student Paper award in ICAC'17. The benchmark dataset presented in the paper "A heterogeneous benchmark dataset for data analytics: Multiphase flow facility case study" is available in Zenodo with Open Access. This paper has been one of the most-downloaded papers published recently in the Journal of Process Control.

1.4 Chapter summary

This chapter introduced the research reported in the thesis. In this chapter, the research questions in the context of monitoring of processes with multiple operating modes are outlined. This chapter presents an overview of the contents of the chapters and briefly introduced how these contents address the research questions. The contributions and publications generated by the research in the thesis are also listed.

Chapter 2

Process monitoring and the challenge

This chapter focuses on the challenges introduced by varying production regimes in process monitoring. It begins with an introduction to the duties and decision-making procedures of process operators, followed by an overview of methods that use observations from process operations to provide operators with information about process performance. The chapter then takes a closer look at data-driven methods using process data, especially multivariate statistical process monitoring methods, because they may be able to accommodate the varying production regimes in the process. Since the monitoring methods need to be useful for end-users, the chapter also summarizes several practical considerations that a monitoring method should fulfil. The desirable performance and the undesirable performance of monitoring are also compared.

The aim of the thesis is to propose monitoring methods that can achieve the desirable performance for processes with varying production regimes. The illustrative examples in the chapter will demonstrate that the varying production regimes may make it difficult for monitoring methods to distinguish between abnormal process performance, which is often indicative of faults and failures in the process, and changes in production regimes. The challenges posed by varying production regimes will limit the effectiveness of monitoring methods especially in a practical setting. The technical chapters of the thesis will propose multivariate statistical approaches to cope with the challenges and to achieve the desirable

monitoring performance. This chapter will summarize the objectives and the solutions proposed in these chapters, followed by the chapter summary.

2.1 Decision making and process monitoring

The motivation of the research in the thesis is to support process operators in decision-making. This section briefly describes how an operator behaves in process operations and how monitoring systems may assist him or her in decision making. Process operators oversee the overall procedure of production and operate the process. For example, they can adjust the set-points of controlled variables, such as the flow rates of raw materials, in order to produce a different type of product. Moreover, operators are responsible for the quality of production, the efficiency, and the safety of the process. Therefore, they also need to make decisions. When the production quality is deteriorating, it may be necessary to maintain process equipment, e.g. by adding catalyst to reactors or by removing residues in heat exchangers. Suspension of production and process shut-downs may also be necessary if there are emergencies. For example, delivery of natural gas may need to stop to release the pressure when the pressure in a storage vessel is excessively high and there is a risk of explosion. Since emergency shut-downs may cause damage to process equipment and reduce profitability, it is preferred to avoid them by maintaining the equipment when the abnormal situation in the process is less severe.

If the performance of the process is abnormal, it is often indicative of faults and failures in the process. The residues in heat exchangers and the excessively high pressure in a vessel are examples of faults and failures that may exist in processes. Process data collected when the process performance is abnormal are anomalous when compared with the process data collected when the process is healthy. In order to know about the real performance of the process, the time trends of process variables, especially quality- and safety-related variables are visualized in real-time for operators in the control room. Operators usually inspect the time trends and decide if there is any abnormal situation in the process that requires maintenance actions. The real-time control system in the process may also be able to report when the value of one variable is too high or too low.

The inspection of the time trends requires time and effort. Moreover, the decision-making procedure depends highly on the experience and the expert knowledge of operators. Monitoring systems using data analysis approaches can reduce the time and effort of inspecting multiple variables by presenting the time trend of an overall indicator of process performance. In particular, the systems are often able to identify abnormal relationships between process variables when there is no visible impact on the time trends of individual variables. The monitoring systems can be further designed to help operators to pin-point the degraded equipment in the process that causes the abnormal situation, which is likely to indicate the existence of faults and failures. Thus operators can maintain the degraded equipment to mitigate the abnormal situation before it becomes significant and results in an emergency shut-down.

The British Standards Institute (BS ISO 13372:2012, BSI (2012)) defined *faults* as conditions of a machine that occur when one of its components or assemblies degrades or exhibits abnormal behaviour, which may lead to failures. BSI (2012) defined *failures* of an equipment as the termination of the ability of the equipment to perform a required function. The task of *process monitoring* defined by Isermann (1997) is to check if the data are within the range of healthy operations and to trigger alarms for the information of process operators. Later on Chiang et al. (2000a) extended the content of process monitoring to the actions from identifying abnormal process behaviour to ensuring the planned production.

Monitoring systems need to reflect true performance of the process. Also, the systems need to present the result clearly to operators. Unlike process variables, the indicator produced by monitoring systems may not have physical meaning, making it difficult for operators to translate the magnitude of the indicator to the severity of a fault. The behaviour of the indicator with respect to abnormal process performance should be clear and easy to understand by operators. Hence the design of monitoring systems should aim to both improve the accuracy of the results and enhance the delivery of such results. The rest of this chapter will review how monitoring systems work and will summarize the desirable behaviour of the indicator achieved by these monitoring systems. In particular, this chapter will discuss how varying production regimes may influence the behaviour of the indicator.

Section 1.1 pointed out that operators must monitor the process performance in order to make correct decisions. The aim of process monitoring is therefore to evaluate the process performance using observations collected from the process. Process monitoring answers questions such as is the process running at healthy conditions, have faults occurred in the

process, and if so, where are these faults are located. Operators can then take the monitoring results into consideration and make decisions for process operations and maintenance accordingly.

2.2 Using observations to monitor process performance

This section first gives an overview of methods for process monitoring using observations from various sources in the process. Then the section reviews the monitoring methods using each type of observation. A benchmark dataset is one of the research outcomes of the thesis. The dataset recreates a dataset containing observations from heterogeneous sources. The section discusses how the thesis will proceed to address the challenges existing in monitoring multimodal processes.

2.2.1 An overview of monitoring methods

The hierarchical tree in Fig. 2.1 summarizes the sources of observations in the process and the methods to use these observations. The first level of the tree shows the categories of monitoring methods using observations from various sources. Measurements of variables in the process are an example of observations. Sensors enable the measurement of process variables, such as temperature, flow rate, level and pressure, during process operations. Industrial Supervisory Control And Data Acquisition (SCADA) systems make it possible to gather and store the operational data collected by sensors. The observations may also include other types of measurements, such as vibration measurements, acoustic signals, and voltage measurements collected from electrical and mechanical sub-systems. Moreover, observations do not necessarily take the form of time series. Such observations include alarm logs generated by the SCADA system, records of technicians conducting maintenance works, and process-specific information, such as process topology representing the connectivity between equipment and sub-systems in the process.

All of the types of observations mentioned above can be used for evaluating process performance. Nevertheless, it may be difficult to achieve reliable evaluation of process performance by visual inspection of observations because the observations may be heterogeneous and numerous. Therefore, operators may not be able to make decisions directly based on these observations.

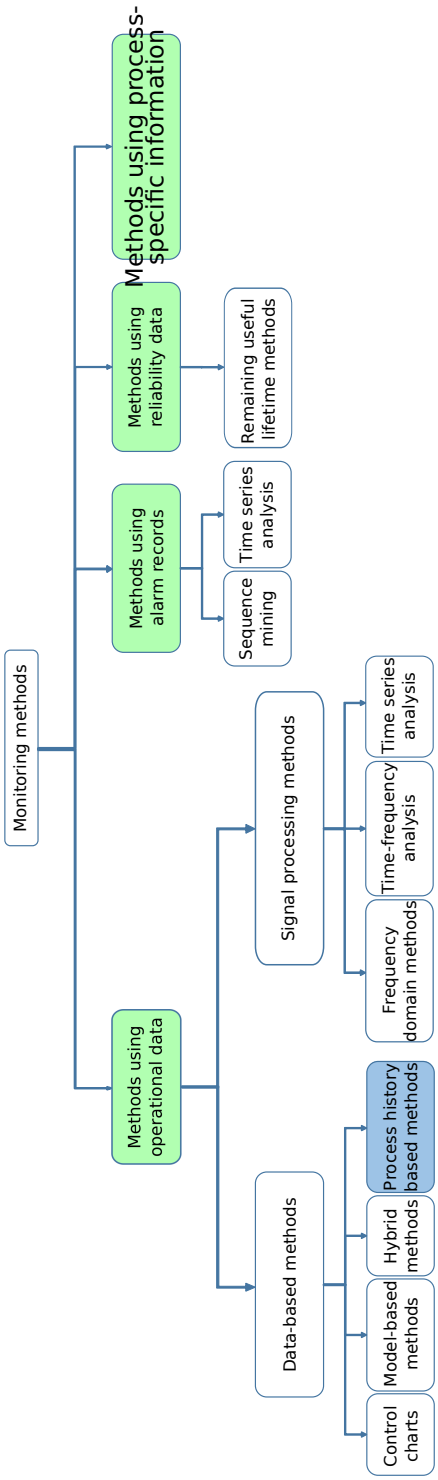


Figure 2.1: Hierarchical tree of monitoring methods using various observations. Blue box: research focus of the thesis. Green boxes: the PhD project produces data in these categories and the thesis reports the data.

Instead, monitoring methods designed for specific types of observations can generate monitoring results that give a more compact representation of process performance than the individual observations. Therefore, the monitoring methods can better help operators in decision making when compared with using observations directly.

The second level of the tree in Fig. 2.1 presents the categories of monitoring methods for utilizing observations from various sources for monitoring the process. In particular, the methods which utilize operational data can be divided into two categories according to the sampling rate of measurement, namely data-based methods and signal processing methods. The next sections will briefly discuss the observations from various sources and the monitoring methods designed for using these observations.

2.2.2 Monitoring methods using operational data

Operational data are measurements of variables collected during process operations. The level and the pressure of a tank, the voltage and the current of a pump, and the vibration of a compressor are several examples of operational data. This section discusses monitoring methods using operational data.

Data-based monitoring methods

The monitoring methods introduced here are based on operational data, especially measurements of process variables. The term *process data* refers to the measurements of process variables collected during process operations.

For process data, Shewhart (1931) first developed control charts for monitoring off-line laboratory measurements of quality-related variables. A control chart is a chart that visualizes the trend of a variable over time and plots the control limits for this variable. The control limits define the interval that the variable should stay within. If the variable violates the control limits, this indicates that the requirement of quality of production is not fulfilled.

Hotelling (1947) developed the Hotelling's T^2 statistic. Multivariate control charts calculate statistics such as the Hotelling's T^2 using process variables and compare the statistics with their control limits. These control charts can detect the situation when the process variables deviate from their healthy operating ranges and draw it to the attention of operators. More

advanced methods are able to build monitoring models that describe the relationship between process variables in healthy operating conditions. The process may not be healthy if the process data do not follow the monitoring models.

A series of review papers summarized the monitoring methods using process data based on quantitative models such as state-space models created based on first principles relationships (Venkatasubramanian et al., 2003c), qualitative models (Venkatasubramanian et al., 2003a) and process history (Venkatasubramanian et al., 2003b). These monitoring methods can provide useful information about process performance for operators so that the operators can know if the process operating condition is healthy without inspecting the trends of all process variables in real-time. These methods may also improve the level of process autonomy by reducing the work load of operators in decision making.

Signal processing methods for process monitoring

Since the time constant of some sub-systems, such as electrical and mechanical sub-systems, can often be much smaller than the time constant of the process, variables in these sub-systems usually change much faster than process variables. Measurements taken from these sub-systems, such as vibration measurements, acoustic emissions, voltage, rotational speed and torque, often have a fast sampling rate. Abnormal conditions in these sub-systems are likely to influence these measurements. For example, cracks in a pipeline may change the natural frequency of the pipeline. The vibration measurement of the pipeline may have a different amplitude and a different frequency due to the cracks. Signal processing methods are applied to extract features from these data when the data have fast sampling rates.

Jardine et al. (2006) reviewed three categories of signal processing methods including time-series analysis, frequency-domain analysis, and time-frequency analysis used for extracting useful information from high-frequency electrical and mechanical measurements. *Time-domain analysis* uses statistics, such as mean, standard deviation, and kurtosis, of the time series data. This analysis is useful for measurements such as torque and rotational speed in motors and turbomachines. *Frequency-domain analysis* often calculates the spectrum of the measurements so that the components at specific frequencies associated with one or more faults, can be highlighted. *Time-frequency analysis* applies to signals of which the characteristics in frequency domain change over time. The wavelet transform is an example of such

analysis. The wavelet transform uses wavelets, which are oscillatory functions, to express the measurement signal as several wavelets with different frequencies and different time shifts. Signal processing techniques can detect changes in higher order statistics, frequency, and amplitude of the measurements whereas these changes may not be easily visible in the original measurements.

In practice, these signal processing methods are particularly suitable for identifying a piece of degraded equipment in the process and locating the degraded equipment, which will be useful for planning of maintenance. The signal processing methods can also be used in combination with the data-based methods introduced in Section 2.2.2.

2.2.3 Monitoring using alarm records

Alarms are audible and visible means of indicating to the operator an equipment malfunction, process deviation, or abnormal condition that requires a timely response (BSI, 2015). Industrial SCADA systems are often able to trigger alarms and warnings that indicate violations of built-in rules in the system, such as a process variable exceeding a limit or the interruption of communication between a sensor module and the SCADA system. Previous works investigated the behaviour of alarm systems. For example, Xu et al. (2012) proposed an approach for designing univariate alarm systems and performance assessment. Wang et al. (2016) presented an overview of industrial alarm systems and the problems existing in such systems.

Due to the existence of a large number of variables, instrumentation and connections in the process, the flood of alarms triggered by the SCADA system may be beyond the capacity of process operators to handle. Therefore, although the alarms in the SCADA system are initially designed to facilitate the operators, they may lead to a more difficult decision-making procedure. The alarm flood needs to be managed in order to extract useful information. Lucke et al. (2019a) reviewed the two categories of alarm data analysis methods for alarm data in the format of sequences and time series, respectively. These methods the alarm records and detect abnormal conditions, and they also include pre-processing to reduce irrelevant alarm records. In an on-line system they can provide refined information to support decision-making.

2.2.4 Monitoring using reliability data

Reliability is the ability of a system to perform its required functions under stated conditions for a specific period of time (IEEE, 1991). Information regarding reliability of process equipment may further improve the evaluation of process performance. *Degradation* is the detrimental change in physical condition, with time, use or external cause (BSI, 2010). Degradation of process equipment is an example of unwanted changes in the process that operators should be aware of. Reliability data may be used to predict the degradation of process equipment and the process performance in the future. The aim of such a prediction is to give operators an estimation of the *remaining useful lifetime* of the equipment in a process, which is the time remaining before the equipment fails so that operators can arrange shut-downs and maintenance to enhance process safety. To predict the degradation of process equipment, degradation models of process equipment may be established based on empirical relationships or using observations from the process. For example, the statistics of the past failures of an equipment may provide a reference for future failures. Understanding the physics of the degradation in a process component may also be useful for predicting the development of degradation and estimating when a failure might occur.

2.2.5 Monitoring using process-specific observations

Process-specific observations, such as the topology and the P&ID of the process, can also assist process monitoring. An overview of the ways to describe the connectivity in industrial processes can be found in Yang et al. (2014). This book also reviewed how the information can be extracted. Jiang et al. (2009) applied the concept of adjacency matrix to describe the relationship between components in order to diagnose the root cause of oscillation in the plant. Yang et al. (2010) described the causal relationship between process variables using the signed directed graph and applied it to fault diagnosis. Yim et al. (2006) presented a software prototype integrating the topology of a process, including the physical structure of the process components and their connectivity, for users to make queries about process variables and to find the root cause of faults.

Moreover, some processes may have other types of observations available in addition to the data types mentioned in previous sections. Jampana et al. (2010) developed a vision sensor using videos taken by a camera from a sight glass of a separation vessel for oil and water

so as to achieve an accurate estimation of the interface level in the vessel. This information is then used as the feedback to the control loop that maintains the interface level.

2.2.6 The PRONTO benchmark dataset

One contribution of the thesis is the PRONTO benchmark dataset (Stief et al., 2019b). This is a joint work conducted in collaboration with Anna Stief. This benchmark dataset is a real-life example of observations from various sources. The dataset contains observations of the following four categories presented in Fig. 2.1: operational data, electrical and mechanical data, alarm records, and process-specific information. All the observations were collected by conducting experiments on a pilot-scale multiphase flow rig. The dataset is available to the public and can be used for development and validation of the monitoring methods for data of these categories. Chapter 4 will provide a detailed description of the experiment design and data collection for the PRONTO benchmark dataset.

2.2.7 Relevance of process data to the research problem

Previous sections reviewed monitoring methods which make use of various types of observations collected from process operations. This section describes how the thesis will proceed to deal with the challenges posed by varying production regimes. Section 1.1 has briefly introduced these challenges.

The thesis will investigate monitoring methods using process data. The reason for focusing on methods using process data is that faults in the process are reflected by process data. Moreover, process variables are often measured for control and safety reasons. The signals are already available and there is no need for investment for additional data acquisition. The sub-systems in a process are often coupled and faults may result in many observations diverging from their typical values during healthy operation. Therefore, the methods mentioned in previous sections may reach the same conclusion using a variety of observations. On the other hand, one type of observation may be more suitable than the other types for identifying some abnormal behaviours in the process.

Process operators often adjust the production regime by changing the set-points and the operating ranges of process variables. In different regimes, the equipment in the process may be connected in a different way and the chemical reactions occurring in the process

may also be different. Therefore, process data are often influenced by varying production regimes. Furthermore, process data will be abnormal if the process is running at an abnormal condition. It is necessary to use process data because they are the most relevant observations for accounting for varying production regimes whilst detecting abnormal process performance. The methods investigated in the thesis fall into the category of process history-based methods in Fig. 2.1.

The second level of the branch of the data-based methods in Fig. 2.1 has three blocks and the thesis will proceed with the process history-based methods. The process history-based methods use historical measurements of process variables collected from healthy process operations to build monitoring models. Faults are detected when new data are not consistent with the monitoring models. These methods are often referred to as *data-driven methods*. The thesis will use the term *data-driven methods* from now on. It is also necessary to note that here *data-driven methods* are different from *data-based methods* reviewed in Section 2.2.2. Data-based methods are methods that generally use operational data in various ways and data-driven methods use the data to train monitoring models.

The reason for choosing data-driven methods is that process data from various production regimes are often available for building monitoring models in practice. For example, the PRONTO dataset has a large assembly of process data collected from several operating modes. Moreover, other methods in the same branch using qualitative and quantitative models may not be used easily. When building monitoring models, additional information of the process may be required. For example, the quantitative model-based techniques require an accurate state-space model of the process (Venkatasubramanian et al., 2003c), which may be established by first principles modelling. However, it may be difficult to build an accurate first principles model for sophisticated industrial processes with a large number of process variables and items of equipment. Furthermore, first principles models under varying production regimes are often different from each other, increasing the difficulty of first principles modelling. Therefore, the thesis will investigate data-driven process monitoring methods for processes with varying production regimes. It will be shown that a good data-driven monitoring model can account for the varying behaviour of several different healthy regimes whilst being able to detect the abnormal operating conditions in the process.

2.3 Data-driven process monitoring

This section first presents the steps in process monitoring, including fault detection, fault diagnosis, and fault prognosis. Then the multivariate statistical process monitoring will be introduced. This section also describes the structure of monitoring systems using data-driven methods.

2.3.1 Steps in process monitoring

Section 2.2 established that the analysis in this thesis will focus on data-driven process monitoring methods. One task of data-driven process monitoring is to detect and diagnose faults in the process. BSI (2012) defined *faults* as conditions of a machine that occur when one of its components or assemblies degrades or exhibits abnormal behaviour, which may lead to failures. Chiang et al. (2000a) also referred to *faults* as undesirable deviations in process variables or process characteristics, which may impact the efficiency of the process and the quality of the production. Such faults often cause changes in process data when compared to the data collected in healthy operating conditions. The task of *process monitoring* as defined by Isermann (1997) is to check if data are within the range of healthy operations and to trigger alarms for the information of process operators. Later Chiang et al. (2000a) extended the content of process monitoring to the actions from identifying abnormal process behaviour to ensuring the planned production.

According to the specific tasks when dealing with faults, the following steps are often taken in process monitoring:

- *Fault detection*: determine the existence of the fault in the process;
- *Fault diagnosis*: identify characteristics of the fault, including its type, location, severity, relevant process variables, and the root cause;
- *Fault prognosis*: predict the development of the fault and estimate the time duration before process failure.

The monitoring results obtained by these methods can facilitate control, maintenance and optimization of the process. For example, a monitoring system can be set up for monitoring a process using data-driven monitoring methods. If the monitoring system reports

that there is a fault in the process, the process equipment may be further inspected and maintained.

2.3.2 Multivariate statistical process monitoring

Process data are typically multivariate with measurements of various process variables in various locations in the process being recorded. The multivariate data often follow statistical distributions due to measurement noise and process disturbances. As shown in the example with two variables x_1 and x_2 in Fig. 2.2, the healthy data follow a bivariate Gaussian distribution. In multivariate process monitoring, a collection of measurements at a specific time point is a *sample*. Each circle in Fig. 2.2 represents a sample with the measurements of variables x_1 and x_2 taken at the same time. Sample 1 shows a fault case where the variable x_1 is outside its healthy range of operation $[x_{1,\text{min}}, x_{1,\text{max}}]$. For example, the liquid level in a tank represented by x_1 may be lower than its limit due to a blockage in the inlet flow pipeline represented by x_2. Sample 2 represents another type of fault where the relationship between x_1 and x_2 does not hold. For example, the flow rates at the entrance and the exit of a pipeline may be different due to a leakage existing in the pipe. Both types of faults may exist simultaneously, e.g. for Sample 3. Instead of the univariate control chart for monitoring the process variables individual, multivariate statistical process monitoring methods were developed to monitor both the individual behaviour of all process variables and the relationship among them. For example, Wise and Gallagher (1996) demonstrated that multivariate statistical process monitoring methods, such as Principal Component Analysis (PCA), can be applied to multivariate process data to calculate the Hotelling's T^2 statistic and the squared prediction error. The two statistics can then detect the faulty case where process variables exceed their healthy operating ranges and the faulty case where the monitoring model learnt by PCA is violated, respectively.

In order to identify the existence of faults in all three faulty scenarios, multivariate statistical process monitoring methods for fault detection infer statistical monitoring models from healthy data and calculate a *monitoring statistic* using the models and the data. The value and the distribution of the monitoring statistic can reflect the behaviour of the healthy data, such as the Hotelling's T^2 statistic (Hotelling, 1947) and the squared prediction error (Jackson and Mudholkar, 1979). *Control limits* of monitoring statistics are defined based on the healthy data. Monitoring statistics of a sample will exceed their control limits if the sample

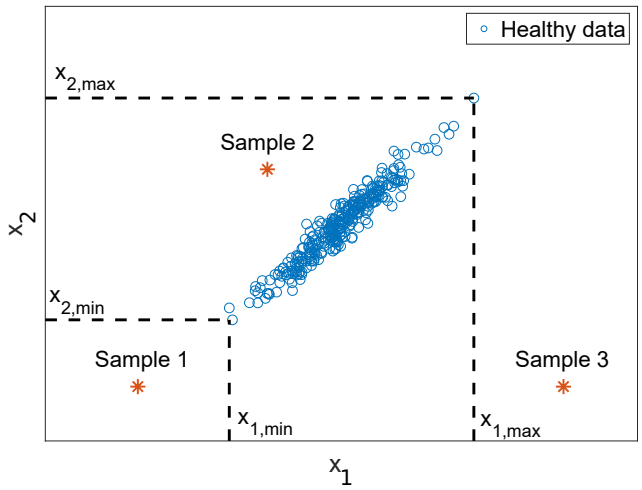


Figure 2.2: Bivariate illustrative example for multivariate process monitoring

does not follow the monitoring model. When used in combination with its control limits, a monitoring statistic can detect violations of the relationships between variables. Therefore, the monitoring methods identify this sample as an anomalous sample.

The monitoring model, the monitoring statistic and the control limit may be used for further fault diagnosis and prognosis. For example, when a fault has been detected, Miller et al. (1998) defined the *contributions* of process variables to the monitoring statistic when a fault is detected. Process variables with large contributions are considered the most relevant for the fault occurrence. The fault can then be located in the process and inspection and maintenance can be arranged accordingly.

2.3.3 A monitoring system

Multivariate statistical process monitoring can be used for monitoring industrial processes. The thesis focuses on the workflow of the monitoring system shown in Fig. 2.3. The left part of Fig. 2.3 pertains to the building of a monitoring model from historical data. The right part shows how the model might subsequently be used with on-line data to detect faults.

The training data are assumed to be collected only from healthy process operations. Fig. 2.3 shows that these training data are used to train a monitoring model using multivariate statistical methods and to tune the parameters needed in the model training. The monitoring statistics are calculated for the training data and the control limits of the monitoring

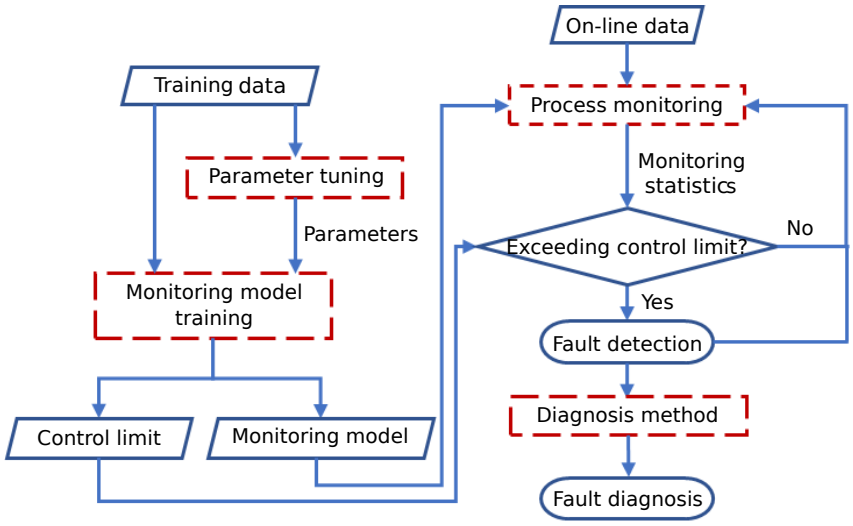


Figure 2.3: General layout of a data-driven monitoring system. Red-dashed box: the thesis develops techniques for this component

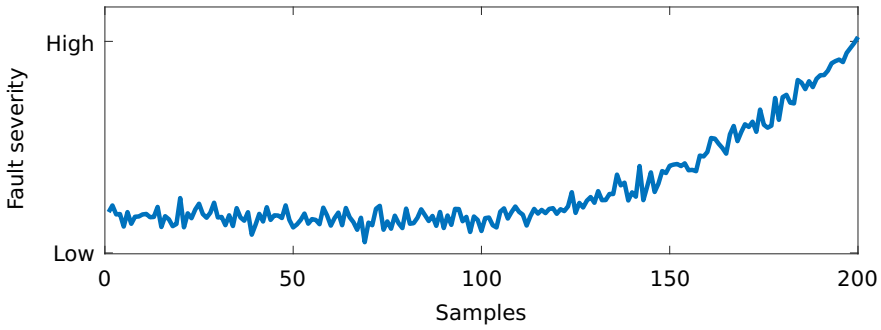
statistics are set accordingly. This monitoring system can then monitor the process performance using on-line data. In the process monitoring step, the monitoring model calculates the monitoring statistics using on-line data. If at least one of the monitoring statistics exceeds its control limit, a fault is detected in the process. Fault diagnosis methods can further diagnose the fault and provide more information for operators to make decisions.

The research in the thesis focuses on the red-dashed boxes in Fig. 2.3, including:

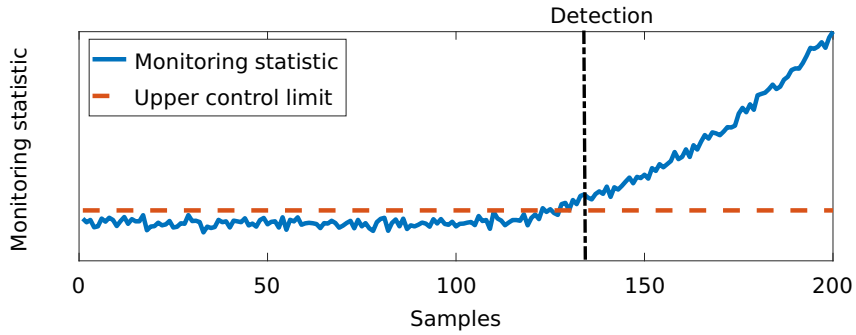
- Development of multivariate statistical process monitoring methods for fault detection;
- Tuning of the monitoring methods;
- Selection of monitoring statistics;
- Contribution-based methods for fault diagnosis.

The overall research objective is to achieve monitoring results that are useful for decision support for process operators. The following example shows how a monitoring statistic should react to a developing fault in the process.

As the fault becomes more severe, the monitoring statistic should increase. An alarm will be triggered and the fault will be detected when the monitoring statistic of a sequence of samples continuously exceeds the control limits. A fault in the process can often be considered as incipient at its early stages (e.g. Sample 100 to Sample 140 in Fig. 2.4(a)). In



(a) Fault severity in the process



(b) A monitoring statistic. Blue curve: monitoring statistic. Red dashed line: control limit

Figure 2.4: An example of desirable monitoring statistic behaviour

this case, the fault will only cause minor violations in the relationships between process variables whereas none of the variables exceeds their healthy range of operation. The monitoring statistics obtained using multivariate monitoring models can detect the existence of such faults. In contrast, univariate control charts can only detect faults when the process variables exceed their healthy ranges. Therefore, a multivariate monitoring system should be able to generate a monitoring model such that the monitoring statistic is more sensitive to the fault occurrence and development than any individual process variable. To achieve the desirable behaviour, the tuning and training of monitoring model, the selection of monitoring statistics and the setting of control limits should be done properly in a monitoring system.

Fig. 2.4(b) has illustrated the desirable behaviour of a monitoring statistic. Since the monitoring results are to be considered by end-users for decision support, it is important to design monitoring systems that are useful to end-users. The next section will review the consideration in developing monitoring methods for industrial practice.

Table 2.1: False alarms and missed alarms

	Healthy	Faulty
No alarm	-	Missed alarm
Alarm	False alarm	True alarm

2.4 Practical applications of monitoring methods

The first part of the classic review paper series on process fault detection and diagnosis (Venkatasubramanian et al., 2003c) presented the desirable characteristics of process monitoring systems. The characteristics include ability to respond quickly, robustness to process noise, adaptability to changes in the process, and being computationally economical.

The development of monitoring systems should consider the following factors: the process, the data, the algorithm, and the use case. Ding et al. (2011); Yin et al. (2014); Ge (2017); He and Wang (2018) highlighted the challenges in the development caused by the complexity in the process and the data. Qin and Chiang (2019) discussed the challenges and opportunities in the various techniques applied to process data analytics.

The last factor, which is less considered in literature, is the use case for monitoring systems. In practice, process operators are end-users of monitoring systems. When scheduling production and maintenance of the process, operators consider the information regarding the occurrence and the characteristics of faults provided by the monitoring system. Therefore, it is essential for a monitoring system to provide results that can be easily understood and used by end-users.

2.4.1 Minimizing false alarms and missed alarms

A monitoring system should reflect the true operating condition of a process. When the operating condition indicated by the monitoring algorithm does not match the true operating condition of the process, false and missed alarms occur. As shown in Table 2.1, a *false alarm* is triggered if the monitoring algorithm detects a fault while the process is running in normal conditions. A *missed alarm* means the monitoring algorithm fails to detect a fault that exists in the process. Fewer false alarms indicate that the monitoring system is robust to the process noise and uncertainties in healthy operating conditions. Fewer missed alarms represent that the monitoring system is sensitive to the fault occurrence.

Both false alarms and missed alarms may have significant implications. False alarms may lead to unnecessary maintenance and shut-downs of the process, which interrupt process production and increase maintenance costs. Missed alarms may result in process faults being unattended. The unattended faults may develop into process failures that may cause health and safety issues. Therefore, the main objective of developing data-driven monitoring algorithms is to minimize the false alarms and the missed alarms. The false alarm rate and the missed alarm rate have been adopted as a major evaluation criteria when comparing the performance of various algorithms for fault detection on benchmark datasets (Downs and Vogel, 1993; Chiang et al., 2000b; Yin et al., 2012; Odgaard and Stoustrup, 2012; Stief et al., 2019c). Moreover, Isermann (1997) noted that monitoring methods should be able to detect faults at their incipient stage in order to leave enough time for maintenance and process recovery. Odgaard and Stoustrup (2012) also used the time difference between fault occurrence and fault detection when applied to a benchmark dataset where the time of fault occurrence was known. The time difference also reflects the sensitivity of monitoring algorithms.

Many works consider the false alarm rate as the major objective for tuning monitoring algorithms because anomalous data are often not available for training monitoring models. When several values of the tuning parameter are possible, Choi and Lee (2004) tuned the monitoring algorithm by selecting the value of the tuning parameter that achieves the minimum false alarm rate. Due to measurement noise, Chen and Zhang (2010), Cai et al. (2017), and Gajjar et al. (2018) also tuned the algorithms such that the false alarm rate is reasonably low. Chen (2010) developed a second-level control limit, which is based on the statistical distribution of the occurrence of false alarms, for further reducing the false alarms in process monitoring.

2.4.2 Interpreting the monitoring results

In addition to minimizing false and missed alarms, several other aspects should be taken into consideration when applying monitoring methods to real-life processes. For example, Qin and Chiang (2019) highlighted the trade-off between the accuracy and the interpretability of machine learning methods applied to process data analytics. The *interpretability* of the methods is high if they can be easily understood by end-users. Data analytics with high interpretability are preferred in practice because end-users can establish a level of trust (Qin

and Chiang, 2019). In contrast, sophisticated algorithms will result in complex models with a favourable level of accuracy whilst sacrificing the interpretability.

The interpretability of monitoring results is important because end-users make decisions based on the results. The multivariate statistical process monitoring methods are preferable due to their ability of summarizing the process behaviour by a single monitoring statistic. The monitoring statistic enables end-users to draw conclusions about process performance by inspecting one indicator instead of all process variables.

A monitoring statistic with good interpretability should have the following behaviour:

1. The same magnitude of the monitoring statistic should be indicative of the same level of fault severity;
2. The control limit should not vary significantly as the process operates;
3. The monitoring statistic should change monotonically with respect to the development of fault severity. This behaviour makes further fault diagnosis and prognosis possible.

Fig. 2.4(b) is therefore a good example because the monitoring statistic clearly reflects the development of the fault. This example gives guidance to the design of monitoring systems. As the workflow in Fig. 2.3 shows, a monitoring system uses algorithms to generate models and to calculate the monitoring statistic and the control limit. Therefore, the design of monitoring systems should consider the algorithm, the monitoring statistic and the control limit jointly to achieve results with better interpretability.

2.4.3 Improving robustness to unseen operating conditions

Apart from faults, other process behaviour may result in process data that are different from healthy training data. This may trigger alarms in the monitoring system. For example, the operating conditions may vary due to the scheduling of production. The historical data used by the monitoring system for model training may not be comprehensive. The monitoring system may trigger false alarms when the process has moved to a healthy operating condition that is unseen in the training data. When used for decision-making, these false alarms may lead to unplanned and unnecessary maintenance and shut-downs of the

process. Hence it is ideal for a monitoring system to be robust to the appearance of new healthy operating conditions in the process.

However, without expert knowledge of process operating conditions, it is impossible for a data-driven algorithm to determine if the process is operating normally in a regime that has not been observed in the historical data, or if a fault has arisen in the process. Instead, the robustness of a monitoring system can be enhanced by incorporating the expert knowledge of operators so that the new healthy operating condition can be acknowledged. This can be achieved by updating the monitoring model using the data from the new operating regime. Monitoring algorithms that can train and update the data-driven monitoring model easily are therefore preferred for designing robust monitoring systems that can account for new operating regimes in the process.

2.4.4 Detecting faults in new operating conditions

Expert knowledge may improve the robustness of the data-driven algorithm towards unseen operating conditions and operators can acknowledge that the unseen behaviour in the new data is healthy. As the process continues operating in the new condition, sufficient data may be collected for the monitoring algorithm to update the monitoring model in order to accommodate the new data behaviour. The sensitivity of the monitoring algorithm can therefore be improved because the updated monitoring model should then be able to identify abnormal process behaviour that has not been observed in the historical data or in the data from the new operating condition.

Updating of the model should be easily implementable in on-line operation, which requires the monitoring method used for training to be flexible to include new data in training. Moreover, the previous expectations should be maintained after model update. The updated monitoring model should still have low false and missed alarm rates and the monitoring results should be easily interpreted by end-users.

2.5 The multimode problem

This section presents several examples of processes with multiple operating modes. It also gives the mathematical description of such processes.

2.5.1 Multiple operating modes in processes

A process may operate in various production regimes according to the scheduling of production in order to cope with changes in the raw materials, to account for changes in the environment and equipment, and to meet changing market demand. The behaviour of the process often changes as the production regime changes. Real-life examples of processes with multiple operating modes can be found in various industrial processes. For example, Grasso et al. (2015) discussed the cylindrical grinding process of steel rolls. The speed of the grinding wheel, the diameter of the cylinder, and the trajectory for grinding the cylinder may change. Another example is the multiphase flow facility presented by Ruiz-Cárcel et al. (2015), where the flow regime in the pipeline varies due to the changes of water and air flow rates. This mimicks the typical multimodal behaviour in the oil and gas industry. Offshore riser-pipe systems may be multimodal due to the varying flow regimes in pipelines and risers. Factors that influence flow regimes include the oil and gas flow rates that the reservoir can produce (Yochum, 1973), physical characteristics such as density and viscosities of oil and gas (Thorn et al., 2012), and the geometry of the pipeline (Xing et al., 2013). Other examples are found in pharmaceutical processes and semiconductor etch processes. Wang et al. (2019) studied multimodal operations of penicillin fermentation. Different feed components will result in a logarithmic or exponential culture growth rate, making the behaviour of the fermentation process and the relationships between process variables different. Wise et al. (1999) mentioned that residue accumulated in the etcher will make the behaviour of the etching process different.

The main characteristic of such processes is that the operating range of process variables in multiple operating modes may be different from each other and the relationship between process variables in multiple modes can also vary. For simplicity, the thesis will refer to "a process with multiple operating modes" as "a multimodal process".

2.5.2 Theoretical description

This section presents a mathematical formulation of the process model for a multimodal process. Such formulation will also be used for developing illustrative examples of multimodal processes.

Assuming J static operating modes exist in the process, the process model for a vector $\boldsymbol{x}$ of process variables can be written as:

Mode 1:

$$\mathbf{0} = \boldsymbol{f}_1(\boldsymbol{x}; \Theta_1) + \boldsymbol{e}_1 \qquad \text{for} \quad \boldsymbol{x} \in X_1 \tag{2.1}$$

Mode 2:

$$\mathbf{0} = \boldsymbol{f}_2(\boldsymbol{x}; \Theta_2) + \boldsymbol{e}_2 \qquad \text{for} \quad \boldsymbol{x} \in X_2 \tag{2.2}$$

...

Mode J:

$$\mathbf{0} = \boldsymbol{f}_J(\boldsymbol{x}; \Theta_J) + \boldsymbol{e}_J \qquad \text{for} \quad \boldsymbol{x} \in X_J \tag{2.3}$$

where X_i is the operating range of the process variables $\boldsymbol{x}$ for the i-th mode. $\boldsymbol{f}_i$ and Θ_i are the model structure and the parameters for the i-th mode. For example in Fig. 2.2, the structure of the equation for generating the data will be linear and Θ are the gradient and the y-axis intercept. The measurement noise $\boldsymbol{e}$ is often assumed to be zero-mean for all operating modes.

The equation for the j-th mode only applies when $\boldsymbol{x}$ is in the operating range X_j for this mode. Again in Fig. 2.2, the operating ranges of the two variables x_1 and x_2 are $[x_{1,\text{min}}, x_{1,\text{max}}]$ and $[x_{2,\text{min}}, x_{2,\text{max}}]$, respectively. Therefore, when the objective of process monitoring is fault detection, the task is to determine if a test sample vector $\boldsymbol{x}_{\text{test}}$ belongs to a given operating range and if $\boldsymbol{x}_{\text{test}}$ follows the process model. If so, the following conditions simultaneously hold for any $j \in \{1, 2, \ldots, J\}$

$$\boldsymbol{x}_{\text{test}} \in X_j, \tag{2.4a}$$

$$\boldsymbol{f}_j(\boldsymbol{x}_{\text{test}}; \Theta_j) = \mathbf{0}. \tag{2.4b}$$

The conditions indicate that, given a sample follows the process model of one operating mode, it may still be an anomalous sample if it does not belong to the corresponding operating range for this mode.

Therefore, a multivariate statistical method for multimodal process monitoring should be able to train monitoring models that can account for the individual models for all operating modes existing in the training data. Moreover, these models should be integrated with the operating ranges of process variables in each mode.

2.6 Challenges in multimodal process monitoring

This section presents an illustrative mathematical model that could represent a multimodal process with three operating modes. This example will demonstrate the challenges in multimodal process monitoring and the ideal behaviour of the monitoring statistics achieved by the monitoring algorithms.

2.6.1 Fault detection in multimodal processes

The considerations concerning sensitivity, robustness, and interpretability of monitoring algorithms introduced in Section 2.4 also apply to multimodal process monitoring. However, the existence of multiple operating modes may pose additional challenges when designing monitoring algorithms. The following bivariate example with three modes is used to generate data for illustration:

Mode 1:

$$\begin{aligned} x_1 &= e_{11} + 8 \\ x_2 &= -0.2x_1 + 5 + e_{12} \end{aligned} \tag{2.5}$$

where $e_{11} \sim N(0, 2.25)$ and $e_{12} \sim N(0, 0.25)$.

Mode 2:

$$\begin{aligned} x_1 &= e_{21} \\ x_2 &= 3x_1 + 5 + e_{22} \end{aligned} \tag{2.6}$$

where $e_{21} \sim N(0, 0.25)$ and $e_{22} \sim N(0, 1)$.

Mode 3:

$$\begin{aligned} x_1 &= e_{31} + 15 \\ x_2 &= -x_1 + 20 + e_{32} \end{aligned} \tag{2.7}$$

where $e_{31} \sim N(0, 0.25)$ and $e_{32} \sim N(0, 0.09)$.

Fig. 2.5 presents the healthy data from Eqns (2.5) - (2.7) and some anomalous data. The anomalous data are generated by inducing a fault to the variable x_2 in the process and the severity here is the deviation in x_2. The healthy data in Fig. 2.5(a) are from Mode 1, Mode 2 and Mode 3 with 50 samples of each mode. Monitoring models and control limits for monitoring statistics are learnt using the healthy data. When generating the anomalous data, the process first operated at Mode 1, moved to Mode 2 and Mode 3, then returned to

Mode 1. Fig. 2.5(c) shows the operating sequence. After the 100th sample, a fault occurs such that a bias is induced to the variable x_2, so that x_2 gradually drifts from its nominal value. The anomalous data are plotted in Fig. 2.5(b). The severity of the fault in x_2 is visualized in Fig. 2.5(d). While the speed of fault development may vary as the operating mode changes, the fault severity will continue developing regardless of operating modes if no maintenance action is taken. On the other hand, the comparison of Fig. 2.5(b) and 2.5(d) shows that the visibility of process faults may be limited because of the multiple operating modes.

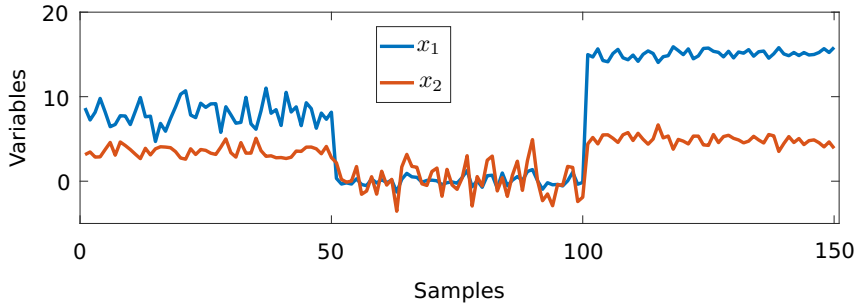
The challenge posed by multimodal processes to process monitoring is that the multiple operating modes cannot be easily distinguished from the faults. Failing to address this challenge will result in missed detection of faults and a fault can develop into severe failures if the process is not maintained timely. Therefore, monitoring algorithms should generate monitoring statistics that can identify the trend of process data deviating from the healthy condition without reacting to mode changes. As described in Section 1.1.3, the research in the thesis addresses this challenge by proposing kernel methods that are suitable for monitoring multimodal processes and forming a monitoring system using such kernel methods.

The monitoring statistics are often functions of the process variables $\boldsymbol{x}$:

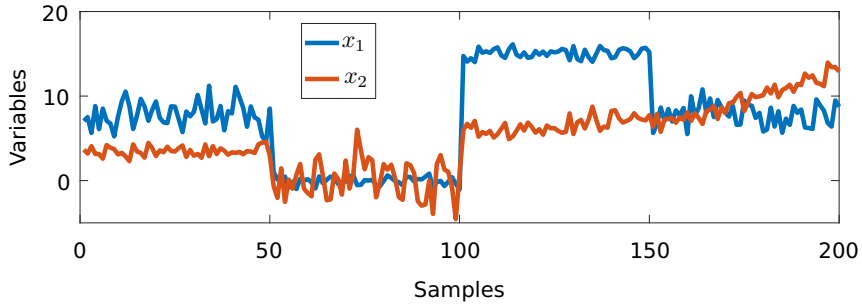
$$I = \boldsymbol{f}_{\text{mon}}\left(\boldsymbol{x}, \Theta_{\text{mon}}\right) \tag{2.8}$$

where I stands for a monitoring statistic. The function f_{mon} and the parameters Θ_{mon} depend on the monitoring model obtained by monitoring algorithms. Various algorithms will result in different functions for the statistic I for the same variables.

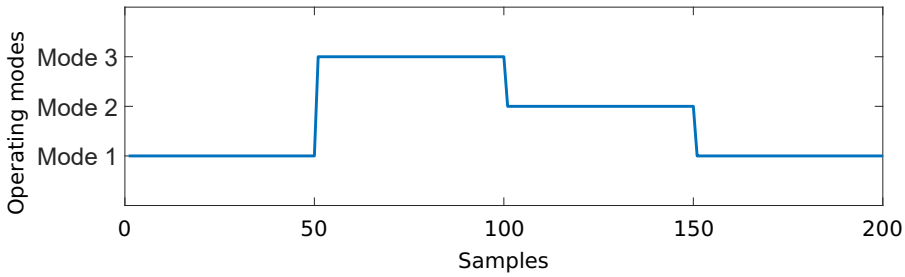
Two examples of the undesirable behaviour and desirable behaviour of monitoring statistics are compared in Fig. 2.6. Fig. 2.6(a) shows the Hotelling's T^2 achieved by linear PCA. This statistic is sensitive to the change in operating modes whilst being able to capture the development of the fault. Such behaviour will lead to increased false and missed alarms which cannot be mitigated by adjusting the control limit. For example in Fig. 2.6(a), the control limit trained from the healthy data will result in missed alarms for the samples between T_1 and T_2 where the drifting has become significant. If the control limit is set lower, false alarms occur between the 50th and the 100th samples where the samples are fault free. Moreover, it may be difficult to interpret the monitoring result because the same magnitude



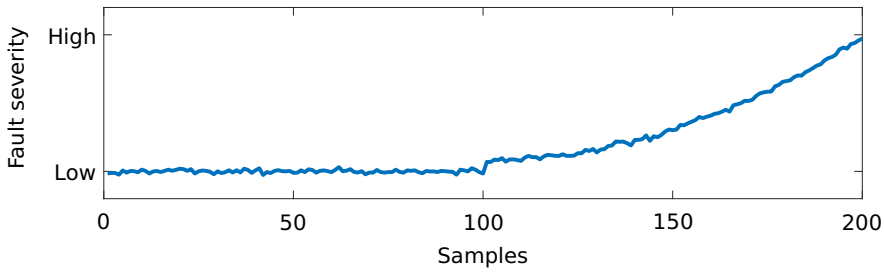
(a) Healthy process data



(b) Process data with a developing fault



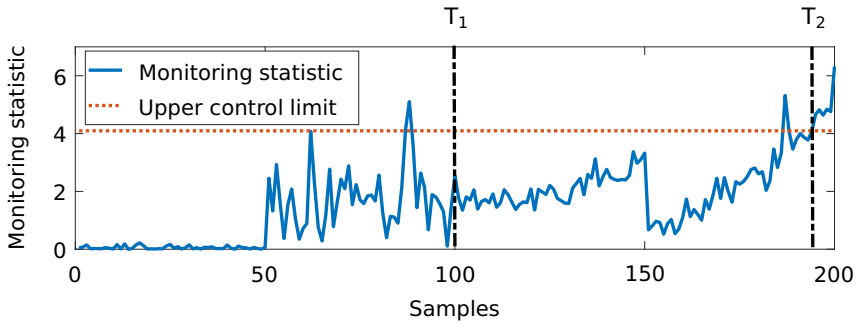
(c) Operating sequence for anomalous data



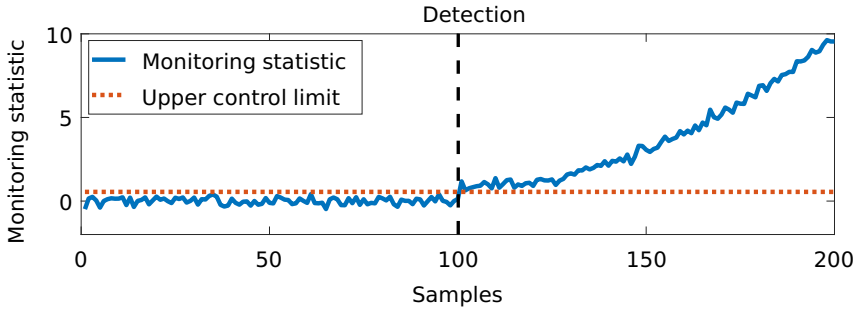
(d) Data drifting occurs after sample 100

Figure 2.5: Illustrative example of a multimodal process

of the monitoring statistic may correspond to different levels of fault severity. For example, the monitoring statistic with a value around 2 represents healthy process operations with no degradation when the process is in Mode 2. In Mode 3 and Mode 1, monitoring statistics



(a) Undesirable monitoring statistic



(b) Desirable monitoring statistic

Figure 2.6: Undesirable and desirable monitoring statistics for the illustrative example

with the same magnitude correspond to anomalous data, with the same value representing a more severe fault in Mode 1. To summarize, a monitoring statistic should satisfy the following criteria:

1. The monitoring statistic should increase monotonically as the fault severity develops;
2. The monitoring statistic should be of the same magnitude for various operating modes if the process is healthy.

If a monitoring algorithm produces a monitoring statistic with the behaviour in Fig. 2.6(a), the algorithm is not reliable because it does not fulfil the first two considerations in Section 2.4, namely that false alarms and missed alarms should be minimized and that the monitoring result should be interpretable. The behaviour presented in Fig. 2.6(a) is often observed when linear algorithms are applied for multimodal process monitoring because linear algorithms are sensitive to the change in operating modes and steady states. On the other hand, Fig. 2.6(b) shows the desirable behaviour of a monitoring statistic. No false alarm is triggered when the process is healthy. Additionally, the statistic can identify the bias starting from Sample 100 and changes monotonically with respect to the fault development, which may be useful for prognosis of the fault behaviour.

2.6.2 Fault diagnosis in multimodal processes

One objective of fault diagnosis is to determine which process variable is related to a fault. This can be done by evaluating the contributions of process variables to monitoring statistics when a fault occurs. Contribution of a variable refers to the extent to which the variable in question influences the value of the monitoring statistic. The thesis will focus on contribution-based fault diagnosis and Chapter 3 will give a detailed review of fault diagnosis techniques, especially contribution-based methods.

The following equation is an example of the relationship between contributions of variables and the monitoring statistic:

$$I = \sum_{i=1}^{m} \mathscr{C}_i \tag{2.9}$$

where $\mathscr{C}_i$ is the contribution of the i-th variable x_i to the monitoring statistic I. The symbol m represents the number of variables in the vector $\boldsymbol{x}$.

To enable fault diagnosis, the monitoring statistic I should also have the desirable behaviour, as introduced in the previous section, so that the contributions of process variables to the monitoring statistic in several operating modes can be compared. Fig. 2.7 gives a graphical representation that shows how the monitoring statistic can be decomposed according to the contributions of the variables $\mathscr{C}_i$.

Fig. 2.7(a) presents the contributions of the two variables to the undesirable monitoring statistic that was shown in Fig. 2.6(a). For Sample 93, which is a healthy sample, the contributions of both x_1 and x_2 to the monitoring statistic are the highest among the four samples, which indicates that both x_1 and x_2 are anomalous for this sample. This contradicts the fact that Sample 93 is healthy. According to the description in Section 2.6.1, the variable x_2 drifts in this faulty case and x_1 does not drift. However, contribution of x_1 may be higher than the contribution of x_2, making the identification of influential variables difficult. Sample 113 was generated when the fault was induced and hence x_2 of Sample 113 should have a large contribution to the monitoring statistic. However in Fig. 2.7(a), the contribution of x_1 is higher than the contribution of x_2. The monitoring statistic shown in Fig. 2.7(a) cannot detect the fault in Sample 113, hence it is undesirable as a monitoring statistic. Such behaviour of contributions of variables is a result of the undesirable monitoring statistic, of which the same magnitude has different meanings in different operating modes. The large T^2 value around Sample 93 in Fig. 2.7(a) means that the contributions of x_1 and x_2 will

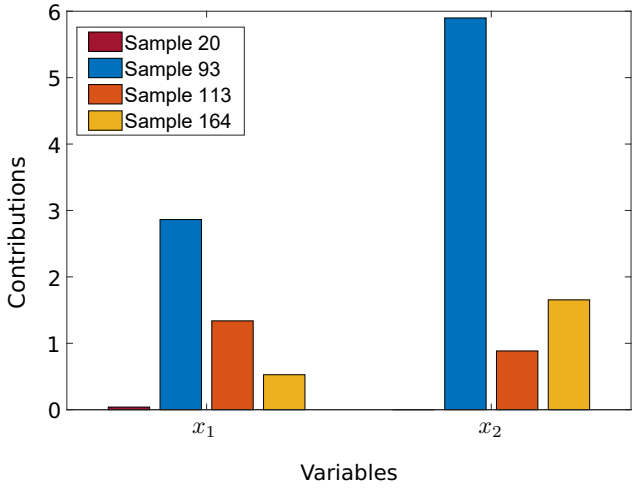
be large, while the small T^2 value around Sample 113 makes it impossible for x_1 and x_2 to have large contributions.

In contrast, the contributions of variables in Fig. 2.7(b) give a better indication of the variable that is influenced by the fault occurrence. Fig. 2.7(b) is the example of contributions of variables to the desirable monitoring statistic shown in Fig. 2.6(b). Variables x_1 and x_2 both have small contributions when the sample is healthy, such as Sample 20 and Sample 93. For Sample 113 and Sample 164, the contribution of x_2 to the is significantly larger than the contribution of x_1 when the fault occurs. By comparing the contributions of variables in healthy and faulty cases, it can be concluded that variable x_2 is related to the fault occurrence.

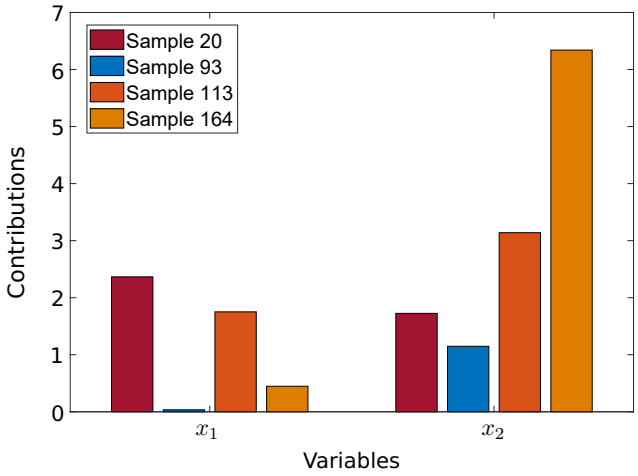
The example shows that the desirable and undesirable behaviours of monitoring statistics also influence the results of fault diagnosis. The desirable monitoring statistic increases when the fault develops whilst being robust to the operating mode changes. This makes the contributions of variables comparable even when the samples are from different operating modes. Therefore, fault diagnosis in multimodal processes will also require monitoring statistics to have the same magnitude in multiple healthy operating modes.

Additionally, the historical data used for training the monitoring model may not always be able to cover the entire operation space of the process since new healthy operating modes may emerge during process operations in order to account for the changes in the market or in the process. Therefore, previously unseen operating conditions may be observed in multimodal process monitoring. Without updating of the model, the current monitoring model will continuously trigger false alarms when a new healthy operating mode appears in the process. A monitoring system should be able to update the monitoring model using data from new operating modes. This can reduce false alarms caused by new operating modes and can improve the monitoring performance for the new modes.

To conclude, monitoring algorithms should be designed properly to cope with the challenges posed by multiple operating modes. The proper design includes the monitoring model training and parameter tuning, the monitoring statistic selection and control limit setting, the diagnosis method associated with the detection result, and the model update procedure.



(a) Contributions of variables to the undesirable monitoring statistic



(b) Contributions of variables to the desirable monitoring statistic

Figure 2.7: Contributions of variables in the illustrative example

2.7 Summary of requirements

Algorithms for fault detection must be sensitive and robust because the result of fault detection will be used for other purposes, including fault diagnosis and fault prognosis. Moreover, the practical considerations introduced in Section 2.4 are important particularly when the monitoring algorithm needs to deliver the monitoring result to operators for decision support. When applied to multimodal process monitoring, the challenges presented in Section 2.6 need to be addressed. Therefore, the research in the thesis aims to develop monitoring methods that satisfy the following requirements:

1. The monitoring statistic should increase monotonically as the fault severity develops;

2. The monitoring statistic should be of the same magnitude for various operating modes if the process is healthy;

3. The contribution of a faulty variable to the monitoring statistic should be higher than the contributions of fault-free variables. This should be valid across multiple operating modes.

2.8 Chapter summary

This chapter reviewed the methods for process monitoring. In particular, it introduced the concepts of data-driven process monitoring and multivariate statistical process monitoring. It also explained the problems posed by processes with multiple operating modes. The thesis will develop multivariate monitoring methods for multimodal process monitoring. The practical considerations summarized in this chapter guide the development and implementation of monitoring algorithms for supporting end-users to make decisions. The monitoring algorithms should reduce false alarms and missed alarms. Also, the algorithms should generate monitoring results that are easy for end-users to interpret. In order to facilitate understanding, this chapter attempted to visualize the desirable and undesirable performance of the monitoring results by demonstrating the way end-users view and interpret the monitoring results.

The chapter also used illustrative examples to demonstrate the challenges in fault detection and diagnosis of multimodal processes posed by varying production regimes. The examples showed that monitoring algorithms for multimodal processes should be designed to detect process faults whilst being robust to mode changes. This chapter also mentioned how the technical chapters will develop monitoring algorithms that can cope with the challenges in multimodal process monitoring.

Chapter 3

State-of-the-art in data-driven process monitoring

This chapter will review the state-of-the-art in process monitoring based on process data, with an emphasis on how the outputs of the various methods employed are presented to users. The chapter will also evaluate the monitoring methods according to the desired behaviours of monitoring systems proposed in Chapter 2 and identify open questions in the existing methods that need to be resolved in order to achieve the desired behaviours in multimodal process monitoring. This chapter will formulate technical tasks for the technical chapters based on the open questions.

3.1 Process monitoring using process data

This section reviews methods using process data. Chapter 2 gave an overview of process monitoring methods using various types of observations (Fig. 2.1) and discussed why the thesis proceeds with the methods based on process data for solving the multimode problem. Fig. 3.1 presents a hierarchical tree of monitoring methods that use process data. The first level of the tree presents the categories of methods and the second level presents examples of specific methods that fall in each category. As discussed in Chapter 2, the focus of the thesis is the multivariate statistical approach in the category of data-driven methods.

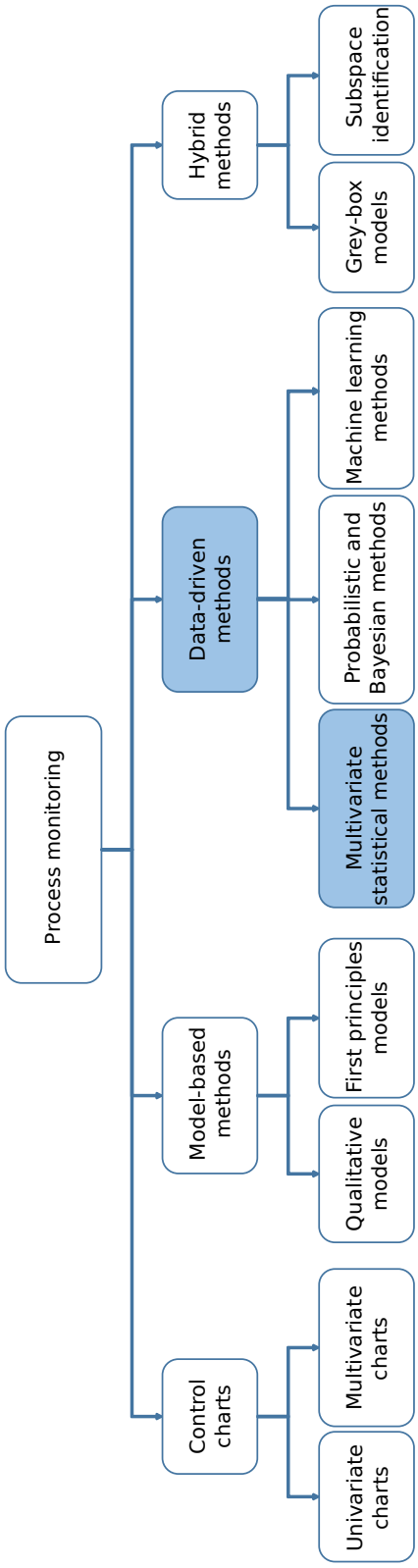


Figure 3.1: Hierarchical tree of process monitoring methods using process data. Blue boxes: research focus of the thesis.

3.1.1 Control charts

Control charts have a long-established history of application to process monitoring and control, which may date back to the Shewhart chart developed in the 1920s (Shewhart, 1931). Statistical process control aims to guarantee the quality of products. Hence the performance of statistical process control can be reflected by the behaviour of quality-related variables, which are process variables that directly reflect the quality of the production in a process. Examples of quality-related variables are the amount of hydrocarbons and the sulfur content in the refined oil produced by refinery.

Hypothesis testing formulates the basis of control charts in order to monitor the quality-related variables from a statistical perspective. For example, one can assume that a quality-related variable follows a statistical distribution, e.g. Gaussian distribution, and its mean and variance in the fault-free condition can be estimated from historical data. The t test and the F test on new measured data of this variable are standard statistical tests for testing if the mean and the variance of the new data are statistically equal to the mean and the variance from the healthy condition, respectively (Oakland, 2007). The method of *univariate control charts* plots the time trend of one quality-related variable with the nominated mean value of this variable and the upper and lower control limits estimated from historical measurements of this variable. Operators can therefore inspect the chart to determine if the quality-related variable falls into its range of healthy operations. It is then possible to determine if the process is operating at a healthy condition.

MacGregor and Kourti (1995) pointed out that univariate control charts used for statistical process control can only analyse several quality-related variables individually and as a result, several control charts need to be monitored by the operators at any given time. To cope with multiple quality-related variables that may be correlated with each other, Jackson (1985) proposed *multivariate control charts* for monitoring the mean vector and covariance matrix of a vector of quality-related variables. Other extensions of control charts include the cumulative sum control chart aiming for accounting sequential data (Barnard, 1959) and the exponentially weighted moving average chart for non-Gaussian data (Borror et al., 1999). Moreover, Xie et al. (1995) proposed the conforming run length chart that counts the number of samples between two non-conforming samples with respect the control limits. Rakitzis et al. (2019) reviewed synthetic-type control charts that combine the traditional control charts with the conforming run length charts.

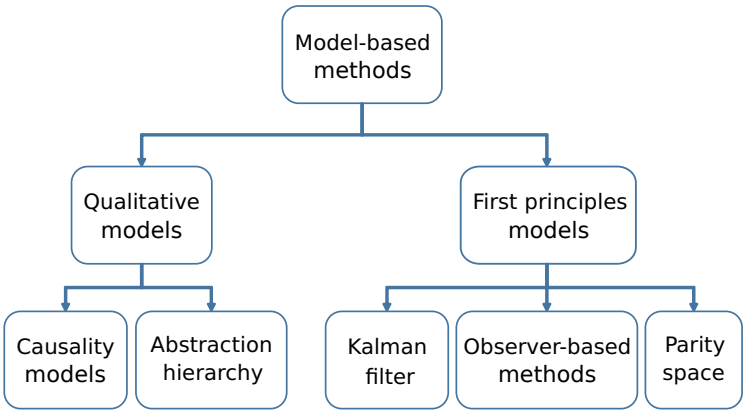


Figure 3.2: Model-based monitoring methods

Control charts have been widely adopted in industrial processes for statistical process control and decision support because their presentation of monitoring results is straightforward and informative for operators. Statistical hypothesis testing originating from the control charts also formulates the basis of other fault indicators used for fault detection.

3.1.2 Model-based methods using process knowledge

Process models describe the behaviour of the process. The process models discussed in this section are built using knowledge about the process, such as the connectivity of process equipment, the mass and energy balances, and the chemical reactions in the process. Monitoring methods that use such models compare the current process behaviour as reflected by the current process data, against the process model built using a-priori information of healthy operating conditions of the process. If the process data do not follow the process model obtained in healthy conditions, then one can infer that there might be a fault in the process. As shown in the second level of Fig. 3.2, descriptive process models fall into two categories. The two categories are qualitative models and quantitative models, according to the types of information available for modelling.

Qualitative models are particularly suitable for fault diagnosis in order to determine the specific faulty scenario in the process and to trace the root cause of the fault. Kramer and Palowitch (1987) used signed digraphs to describe the relationship between process variables in order to establish *causality models* which describe the causality between the process variables and the presence of a certain fault. The faulty scenario is then diagnosed by checking the current process data with respect to the causality models. Finch and Kramer (1988) abstracted the process and decomposed the process into sub-systems such that the

fault can be located by searching for the faulty sub-system. The fault is then diagnosed by searching for the faulty component within the sub-system using the *abstraction hierarchy*. Such analysis is often done off-line when a fault has been detected so as to provide more information about the fault for operators to make decisions to fix the faulty components.

Quantitative models reflect the physical relationships between process variables, such as the mass, component, and energy balances. First principles models based on such relationship often take the form of a set of differential-algebraic equations. These models can be reduced to the forms of state space models, autoregressive models, or input output models. Given that an accurate monitoring model can describe the healthy behaviour of the process, the fault can be detected by running the process model in parallel with the process such that the model outputs and the real measurements can be compared. The residuals, which measure the difference between the model outputs and the current process data, can be used as fault indicators. Isermann (1984) and Gertler (1991) reviewed how the first principles relationship can be reduced and how the reduced models can be used for process monitoring.

There are several ways to construct process models from first principle relationships. Kerr (1977) used Kalman filters to estimate the state variables and their confidence regions. A fault is detected if the current states estimated using current process data deviate significantly from the original state-space model. Kiasi et al. (2013) applied a marginalized likelihood ratio test to the residuals generated by Kalman filter-based estimators. The maximum likelilood estimation of the time of fault occurrence can be obtained by maximizing the marginalized likelihood of the residuals. *Observer-based* methods can generate residuals that quantify the difference between the predicted model outputs and the real process outputs (Ding, 2008). *Parity space* methods build a temporary model between the process inputs and outputs, enabling the residual calculation between the predicted outputs and the measured outputs. Patton and Chen (1991) reviewed the usage of parity space for fault diagnosis and an example of parity space methods is given in Ding et al. (2009). Quantitative model-based approaches can only perform well if they have access to an accurate process model. It is also necessary to notice that, although the process models may be known a-priori, real-time process data are still needed for the residual calculation.

3.1.3 Data-driven methods using historical data

Data-driven methods depend on historical process data that are made available by industrial SCADA systems. Monitoring models discussed in this section are data-driven, which means that the models are trained using historical data. Developments in the area of data analysis and machine learning include artificial neural networks and methods for dimension reduction such as principal component analysis. These developments inspire the application to data-driven process monitoring, enabling the analysis of process data and the detection and diagnosis of process faults based on process data.

In practice, despite the fact that some variables are difficult to measure, abundant data may be available through data collection from process operations. The case study in Chapter 4 is an example. The flow regimes of the multiphase flow in the pilot plant may change rapidly as the operating mode changes while the parameters needed for building first principles models, such as the friction in the pipeline, may not be measurable. On the other hand, process data from numerous process variables under varying flow regimes were collected during the experiment. These data make it suitable to apply data-driven process monitoring methods. Therefore, the thesis focuses on data-driven process monitoring. Section 3.2 gives a detailed review of the state-of-the-art in this area.

3.1.4 Hybrid methods

Venkatasubramanian et al. (2003b) highlighted that no single method that can cope with all process monitoring problems exists. To alleviate this, *hybrid methods* combine various process monitoring techniques which are complementary to each other in order to achieve better performance and to overcome the drawbacks of individual techniques (Venkatasubramanian, 2019). For example, model-based methods using process knowledge and data-driven methods can be combined when there is some physical knowledge of the process available and process data are also collected.

When building the causal map for a complex process with a variety of process variables, Yang et al. (2014) presented an overview of the approaches to capture the causality and the connectivity between process variables. Several examples of using causality for process monitoring are as follows. Jiang et al. (2009) used the adjacent matrix to describe such causal relationship and diagnosed the root cause of faults accordingly. Suresh et al. (2019)

calculated the transfer entropy of process variables using process data so as to quantify the causality between process variables. Bauer et al. (2007) used similar techniques for identifying the path of the disturbance propagation through the process variables based on process data.

Data-driven monitoring models can also be made more practical by incorporating physical knowledge of the process. For example, van Lith et al. (2003) combined the general first principles relationship of distillation columns with the data-based process dynamics for a specific batch distillation column to build a *grey-box* model for monitoring this column. Another way forward is the process models built by data-driven *subspace identification* approach. For example, Treasure et al. (2004) adopted a residual-based monitoring framework for data-driven state-space models for fault detection.

The general objective of developing hybrid methods is to make use of all available types of observations of the process to obtain more accurate evaluation of the process performance.

3.2 Data-driven process monitoring

Data-driven process monitoring requires availability of large assemblies of process data and the application of data analytic techniques. Data-driven methods use process data to build monitoring models that can accurately describe the process running in healthy conditions. The first level of the hierarchical tree in Fig. 3.3 summarizes the categories of data-driven methods for process monitoring. The second level presents examples in each category.

3.2.1 Multivariate statistical methods

Multivariate Statistical Process Monitoring (MSPM) inherits the idea of monitoring the changes in mean values and covariances of quality-related variables from control charts (Wise and Gallagher, 1996). In real-life processes, however, quality-related process variables may not be adequate to describe the operating conditions of the process. Therefore, MSPM methods use process data such as flow rate, temperatures and pressures in additional to quality-related variables for process monitoring. For example, the flow rates of feed flows, the

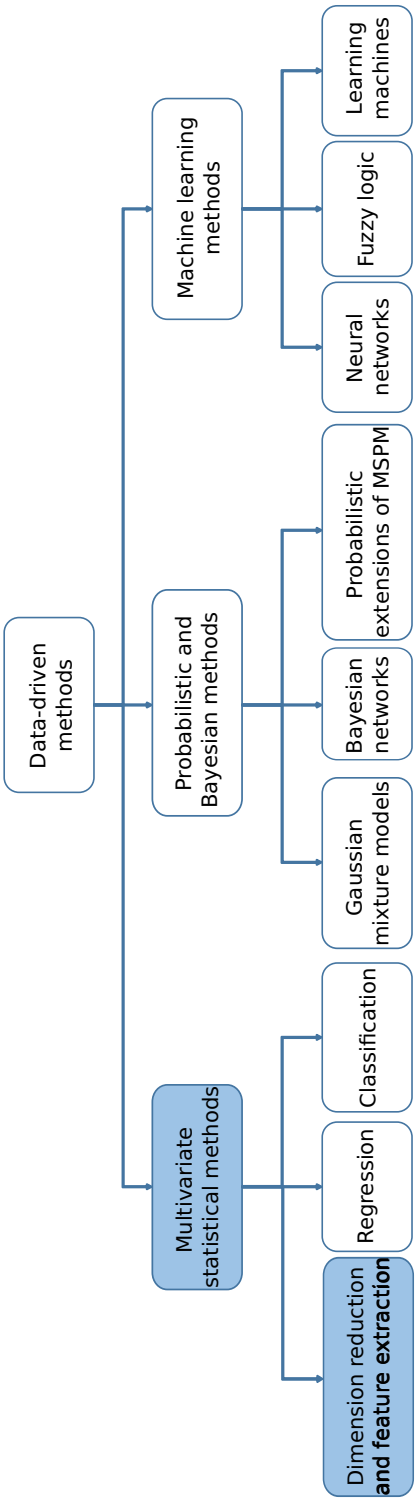


Figure 3.3: Hierarchical tree of data-driven monitoring methods. Blue box: research focus of the thesis

temperature and the pressure of a reactor may be needed to monitor the production in the reactor.

MSPM methods aim to automatically capture the underlying relationships of process variables by analysing process data. MSPM methods extract *features* from process data. These features are intermediate variables derived from original process data and are often representative of the underlying relationships in the process. *Monitoring statistics* are defined as functions of features. Similarly to control charts that monitor means and variances of quality-related variables, MSPM methods inspect the monitoring statistics to determine if the underlying relationships are violated. The usage of monitoring statistics in MSPM methods maintains the advantage of the control charts, which is the straightforward and informative presentation of monitoring results for end-users.

The desirable behaviour of monitoring systems presented in Section 2.5 applies to the MSPM methods for multimodal process monitoring. The MSPM methods are expected to detect the faulty behaviour and track the development of fault severity whilst being robust to changes in the operating mode. The thesis will focus on the development of MSPM methods that fulfil these expectations. In order to do so, Section 3.3 of this chapter will review MSPM methods and discuss if the state-of-the-art in MSPM can achieve the desirable behaviour.

3.2.2 Probabilistic and Bayesian methods

Control charts can also be seen from a probabilistic perspective. The control charts build a probabilistic model with certain parameters to describe the distributions of quality-related variables. Probabilistic and Bayesian methods are developed to quantify how likely it is to observe the current process data given that the process is healthy. If the likelihood is low, one may conclude that the process is abnormal. If faulty data are also available for training the monitoring model, probabilistic and Bayesian methods can determine if a specific faulty scenario has arisen, making fault diagnosis possible.

The distributions are often assumed to be Gaussian in control charts such that the mean and the variance are sufficient for the monitoring model. However, process variables do not necessarily follow a Gaussian distribution especially if the process is running in several operating modes. In order to account for non-Gaussian data, Yu and Qin (2008) and

Xie and Shi (2012) used *Gaussian Mixture Models* (GMMs) to extend the multivariate Gaussian assumption to a combination of several Gaussian distributions, making it particularly suitable for multimodal process monitoring. Gonzalez et al. (2015) adopted kernel density estimation which is an extreme case of GMMs. Kernel density estimation assumes that each data sample is drawn from a specific Gaussian distribution with certain parameters in order to build a non-parametric descriptive model of the overall distribution of the data. Chen and Zhang (2010) and Jiang and Yan (2019) also improved GMM-based methods by combing them with MSPM approaches where the representative features are more suitable for applying GMMs.

The Bayes' theorem was first proposed in 1763 (Bayes, 1763). It is widely used for calculating conditional and posterior probabilities based on prior probabilities and observations. For more than two variables that are dependent on one another, *Bayesian networks* applies Bayesian inference to process data in order to learn the dependency relationship of multiple process variables, making it possible to identify the root cause of a fault in fault diagnosis. For example, Cai et al. (2017) reviewed the method of applying Bayesian networks to infer causal relationships between variables for fault diagnosis. The learning procedure may be the other way around. Heckerman (1995) formulated the Bayesian network first, and then learnt the causal map of process variables under the Bayesian network formulation when there are observations available.

The original MSPM methods assume the representative features are deterministic and the randomness in the measured variables are due to measurement noise. An example is the PCA model used by Yin et al. (2014). In such a PCA model, the principal components are calculated by projecting the measured variables and the there is no assumption of the uncertainty or the distribution of the principal components. By contrast, Kim and Lee (2003) and Ge (2018) assumed the representative features to be random variables with certain distributions and the model with the highest probability are learnt. Such assumptions can lead to a probabilistic extension of MSPM methods. Such a formulation can be compatible with the GMM assumption. The mixture of probabilistic MSPM models can be used for multimodal process monitoring, where each probabilistic model describes one operating mode of the process. A Bayesian approach can be used to determine the current operating mode of the process. Choi et al. (2005b), Ge and Song (2010), and Raveendran and

Huang (2017) presented examples of the mixture of probabilistic MSPM models. Probabilistic MSPM models can obtain representative features and monitoring statistics, making the probabilistic formulation useful for process monitoring.

3.2.3 Machine learning methods

Robinson (2018) distinguished between *data science* and *machine learning* and suggested that data science produces insights while machine learning produces predictions. In particular, Robinson (2018) mentioned that the methods of data science often have better interpretability, while machine learning methods are more difficult to explain. When applied to process monitoring, the previously introduced multivariate statistical methods lean more towards the data science category because the model structure in these methods is rather simple. There are also examples of machine learning methods adopted for process monitoring. In general, machine learning methods may have several layers to improve the accuracy of monitoring models obtained. On the other hand, it may take some effort to understand how the monitoring result produced by these methods reacts to the fault occurrence in the process.

Among the most fast-developing and widely adopted machine learning techniques, *neural networks* are an advanced technique for accurately modelling the input-output relationship between variables. Lennox et al. (2001) reviewed the industrial application of artificial neural networks. The applications include process modelling, process monitoring, and process control. Sorsa et al. (1991), Pirdashti et al. (2013), and Nor et al. (2019) reviewed the neural network-based methods and their roles in data-driven fault detection and diagnosis. Another example is the *fuzzy logic*-based method. Dash et al. (2003) and Musulin et al. (2006) used fuzzy logic to further improve the robustness of fault diagnosis methods in presence of the uncertainty in the process. Evsukoff and Gentil (2005) combined fuzzy logic with neural networks in order to enhance the generalizability of the neural network models. Other *learning machines*, such as the support vector machine (Xiao et al., 2016; Onel et al., 2019), the support vector data description (Li et al., 2017; Zhou et al., 2019) and the extreme learning machine (Xu et al., 2015b), have also be used for data-driven process monitoring. The authors did not, however, discuss issues concerning interpretability.

3.3 Multivariate statistical process monitoring

This section suggest a framework for applying MSPM methods to process monitoring and the commonly adopted algorithms in these steps. The necessary background and theory are explained. Also discussed is how the commonly adopted algorithms fit to the framework in order to deliver diagnostic results to end-users.

The framework for MSPM comprises the general steps shown in Fig. 3.4. Other steps, such as data cleaning (Xu et al., 2015a), may also be needed when analysing real-life datasets. However, Fig. 3.4 does not include these steps because they are not the focus of the thesis.

The objective of using MSPM methods is to provide end-users with the evaluation of process performance using process data. The *feature extraction* step aims to extract representative features from data collected from the process. A *feature* is often a variable defined as a combination of process variables. The *fault detection* step sets up criteria for fault detection and delivers the conclusion regarding fault occurrence to end-users. If a fault has been detected, the *fault diagnosis* step analyses the faulty data and generates more information about the fault. If there are data available from faulty scenarios and quality-related measurements associated with them, features can be extracted such that they are the most representative with respect to the quality-related measurements.

3.3.1 Feature extraction

Section 3.2.1 discussed that the MSPM approaches extend the traditional control charts by extracting representative features from the original process variables so that the behaviour of the process data can be better captured. Therefore, the objective of the feature extraction step is to obtain the most representative features. By setting different objective functions to measure how representative a feature is, various feature extraction methods have been developed. Wise and Gallagher (1996) reviewed several examples of linear multivariate approaches and their application to data from chemical processes. Fig. 3.5 presents a hierarchical tree of the categories of feature extraction methods.

Many linear feature extraction methods have similar mathematical structures. Assuming a process variable vector $\boldsymbol{x} \in \mathbb{R}^{m\times 1}$, a feature vector $\boldsymbol{z} \in \mathbb{R}^{m\times 1}$ is extracted from $\boldsymbol{x}$ using a projection matrix $P \in \mathbb{R}^{m\times m}$ by $\boldsymbol{z} = P\boldsymbol{x}$. The objective is to find P such that $\boldsymbol{z}$ is representative of $\boldsymbol{x}$. Assuming that the first variable $z_1 = p_1^\top \boldsymbol{x}$ in $\boldsymbol{z}$ is the most representative one,

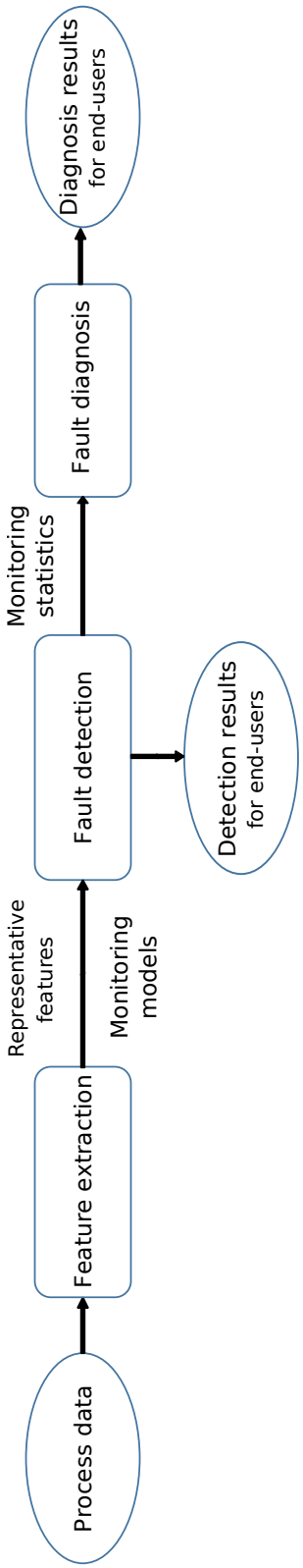


Figure 3.4: Steps of applying MSPM methods

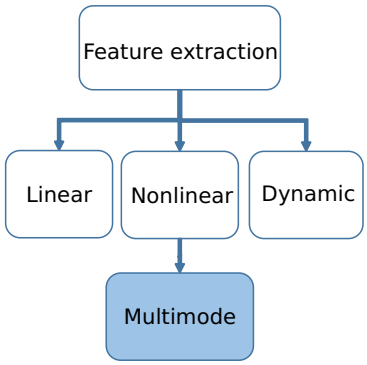


Figure 3.5: Hierarchical tree of feature extraction methods. Blue box: research focus of the thesis. Orange box: additional data needed.

the projection vector $p_i \in \mathbb{R}^{m\times 1}$ is obtained by solving the following optimization problem:

$$p_1 = \arg\max \frac{p^\top Ap}{p^\top Bp} \tag{3.1}$$

In Eqn (3.1), A and B are data-related covariance matrices calculated using X and the columns of X are data samples from $\boldsymbol{x}$. In $P = [p_1, ..., p_m]$, p_i are ranked as the value of the objective function in Eqn (3.1) decreases.

As summarized in Table 3.1, this structure holds for the Principal Component Analysis (PCA) (Chiang et al., 2000a), Partial Least Squares (PLS) (MacGregor and Kourti, 1995), Fisher Discriminant Analysis (FDA) (Chiang et al., 2000b) and Canonical Variate Analysis (CVA) (Russell et al., 2000) with various choices of A and B. Table 3.1 gives a few example of A and B used by various methods.

In Table 3.1, $\boldsymbol{x}$ is a vector of process varaibles and X denotes the measurements of $\boldsymbol{x}$. $\boldsymbol{y}$ are quality-related variables when available and Y denotes the measurements of $\boldsymbol{y}$. Σ denotes the covariance matrix, e.g. Σ_{XX} denotes the covariance matrix of X. I is the identity matrix. For PLS and FDA, the objective is to find the features that are most relevant for the quality-related variables or faulty scenario labels. Y stands for the quality-related measurement or discrete faulty scenario labels. Both X and Y are projected using $p_{\boldsymbol{x}}$ and $p_{\boldsymbol{y}}$. Therefore, PLS and FDA belong to the category where quality-related variables are available for feature extraction. In CVA, in order to extract the features that capture temporal correlations between process variables, Σ_{pf} represents the covariance matrix of the past and future measurements and p_p and p_f are the projection vectors for the past vector and the future vector, respectively. Hence CVA is an example of a feature extraction method for dynamic processes in Fig. 3.5. It can be concluded from Table 3.1 that linear feature

Table 3.1: Summary of linear feature extraction methods

	A	B	P	Remarks
PCA	Σ_{XX}	I	$p_{\boldsymbol{x}}$	Features that explain the most of the variability in X
PLS	$\begin{bmatrix} 0 & \Sigma_{XY} \\ \Sigma_{YX} & 0 \end{bmatrix}$	$\begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}$	$\begin{bmatrix} p_{\boldsymbol{x}} \\ p_{\boldsymbol{y}} \end{bmatrix}$	Features that explain the most variability in both X and Y simultaneously
FDA	$\begin{bmatrix} 0 & \Sigma_{XY} \\ \Sigma_{YX} & 0 \end{bmatrix}$	$\begin{bmatrix} \Sigma_{XX} & 0 \\ 0 & \Sigma_{YY} \end{bmatrix}$	$\begin{bmatrix} p_{\boldsymbol{x}} \\ p_{\boldsymbol{y}} \end{bmatrix}$	Features in $\boldsymbol{x}$ that have the highest correlation with $\boldsymbol{y}$
CVA	$\begin{bmatrix} 0 & \Sigma_{pf} \\ \Sigma_{fp} & 0 \end{bmatrix}$	$\begin{bmatrix} \Sigma_{pp} & 0 \\ 0 & \Sigma_{ff} \end{bmatrix}$	$\begin{bmatrix} p_p \\ p_f \end{bmatrix}$	Features with the highest autocorrelation

Table 3.2: Summary of other feature extraction methods

	Objective function	Remarks
ICA	The non-Gaussianity of the feature	Features that are independent from each other are extracted
SFA	The "slowness" of the feature	Features that change the slowest are extracted

extraction methods often seek to obtain features that are most representative in the sense of data covariance. Therefore, the choice of the covariance matrix will have a significant influence on the feature extraction outcome. Moreover, previous works extended these linear methods to dynamic, nonlinear and batch process monitoring by properly specifying the covariance matrices (Chen and Liu, 2002; Choi and Lee, 2004; Odiowei and Cao, 2009; Kourti et al., 1995; Nomikos and MacGregor, 1994).

Moreover, since Eqn 3.1 is mainly suitable for process data that follow linear models, there have been other objective functions, which may not necessarily take the form of Eqn (3.1). Other feature extraction methods use these objective functions to account for other types of behaviour of process data. The Independent Component Analysis (ICA) (Lee et al., 2004b) and Slow Feature Analysis (SFA) (Shang et al., 2015) presented in Table 3.2 are two examples that are customized for the purpose of process monitoring. ICA can achieve features that are not only orthogonal to each other but also independent in a statistical sense (Hyvärinen and Oja, 2000). These features can facilitate fault isolation and root cause analysis. SFA is adopted based on the assumption that, by nature, the slowly varying features in process data may reflect the dynamics of the process while the fast-varying ones are more likely to be random disturbance and noise.

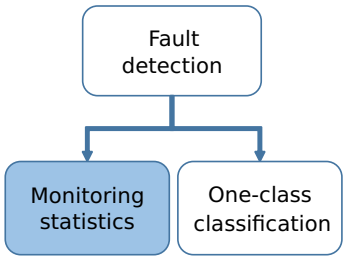


Figure 3.6: Hierarchical tree of fault detection methods. Blue box: research focus of the thesis.

To conclude, basic feature extraction techniques may be able to handle complicated process data if the covariance matrix is constructed properly.

3.3.2 Fault detection

It might be difficult for process operators to draw conclusions directly from the representative features extracted in the previous step because these features are multidimensional and the relationship between them may not be explicit. Instead, *monitoring statistics* are defined as functions of the representative features such that a few monitoring statistics can reflect the behaviour of all features and all process variables. Therefore, the actual fault detection step in MSPM is realized by designing monitoring statistics and setting control limits for these statistics.

Another approach for fault detection is based on classification. One-class classification methods assume all training data belong to one class and detect which new samples are similar to the training data. The monitoring method answers the question of whether the new data sample belongs to the historical healthy data set when a one-class classification method is adopted. For example, Sukchotrat et al. (2009) proposed how several one-class classification methods can be applied to improve the multivariate control chart. Mahadevan and Shah (2009) adopted one-class support vector machine to determine if a sample is located far away from the healthy data.

Fig. 3.6 summarizes the two categories of presenting fault detection results, namely the monitoring statistics and the classification, to end-users.

Eqns (2.4a) and (2.4b) in Chapter 2.5.2 described the two conditions that a data sample should satisfy if the process is healthy. Monitoring statistics used in fault detection methods aim to detect if these two conditions are satisfied. In MSPM, the monitoring statistics are

defined as functions of the representative features $\boldsymbol{z}_r$ and the residuals $\boldsymbol{z}_e$ extracted from the process variable vector $\boldsymbol{x}$. The representative features and the residuals are defined by the two sub-matrices of the projection matrix P, namely P_r and P_e. P_r has the first r rows of P and P_e contains the rest $m - r$ rows of P. Given $P_r = [p_1, \cdots, p_r] \in \mathbb{R}^{r \times m}$, the representative features $\boldsymbol{z}_r$are defined as

$$\boldsymbol{z}_r = P_r \boldsymbol{x} \tag{3.2}$$

Given the projection matrix $P_e = [p_{r+1}, \cdots, p_m] \in \mathbb{R}^{(m-r) \times m}$ for the residuals, the residuals $\boldsymbol{z}_e$ are defined as:

$$\boldsymbol{z}_e = P_e \boldsymbol{x} \tag{3.3}$$

For example, Lowry and Montgomery (1995) presented the usage of the Hotelling's T^2 statistic in multivariate control charts for monitoring the mean value of process variables. When adopted in MSPM, T^2 is calculated as the normalized sum of squares of $\boldsymbol{z}_r$:

$$T^2 = \boldsymbol{z}_r^\top \Lambda_r^{-1} \boldsymbol{z}_r, \tag{3.4}$$

where Λ_r is a diagonal matrix with the variances of $\boldsymbol{z}_r$ as its diagonal elements. The corresponding fault detection criterion is

$$T^2 > T^2_{\text{UCL}} \tag{3.5}$$

where T^2_{UCL} is the upper control limit of T^2. When $\boldsymbol{z}_r$ is extracted by linear algorithms, $\boldsymbol{z} = P\boldsymbol{x}$ where P is the linear projection matrix. Therefore, T^2 being smaller than T^2_{UCL} indicates that $\boldsymbol{z}_r$, the projection of $\boldsymbol{x}$ to the representative feature space, lies within a bounded region in this space. This bounded region represents the healthy operating ranges of all process variables.

As introduced in the previous section, one of the objectives when retaining representative features is to retain the features that can explain the maximum variability in the original data. By doing so, the monitoring model can capture the maximum information carried by the original healthy data and is inclusive of the healthy data. Therefore, a new data sample that violates the monitoring model is likely to be faulty. The Squared Prediction Error (SPE) measures the mismatch between the model and the data (Qin, 2012). SPE is

often defined as the difference between the original process variables and the values of the variables reconstructed using the monitoring model and the original measurements.

$$\text{SPE} = \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2 \tag{3.6}$$

where $\hat{\boldsymbol{x}}$ is the reconstructed value of $\boldsymbol{x}$ using the projection model. For example, when PCA is applied to $\boldsymbol{x}$, the SPE becomes

$$\text{SPE} = \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2 = \|\boldsymbol{x} - P_r^\top P_r \boldsymbol{x}\|_2 \tag{3.7}$$

In the SPE test, the corresponding hypothesis of $\boldsymbol{x}$ being collected from the healthy process operating conditions is

$$\text{SPE} = \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2 = 0. \tag{3.8}$$

and it can also be interpreted as that the following model holds for $\boldsymbol{x}$

$$\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2 = 0 \tag{3.9}$$

which corresponds to Eqn (2.4b). Similarly to T^2, a large SPE value means that the data sample does not follow the monitoring model. Therefore, a faulty sample is detected using SPE if

$$\text{SPE} > \text{SPE}_{\text{UCL}} \tag{3.10}$$

where SPE_{UCL} is the upper control limit of SPE.

3.3.3 Fault diagnosis

After a fault has been detected, additional information regarding the fault can assist end-users to evaluate the situation and take maintenance actions if necessary. Therefore, fault diagnosis methods aim to locate the fault and to identify the type of fault. Fig. 3.7 summarizes the categories of fault diagnosis methods.

After feature extraction and fault detection, the fault needs to be diagnosed by fault diagnosis. Qin (2012) reviewed the techniques used for fault diagnosis, including identification of the direction of faults for simple sensor faults and classification of the fault when a set of possible faults are known in advance. Thornhill et al. (2001), Ahmed et al. (2017) and

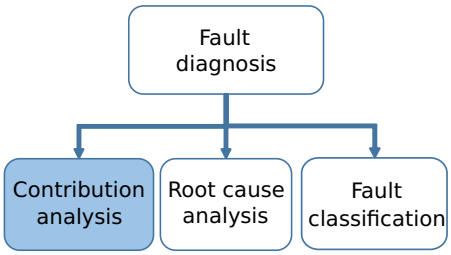


Figure 3.7: Hierarchical tree of fault diagnosis methods. Blue box: research focus of the thesis.

Amin et al. (2018) showed how the root cause of faults can be traced using the original measurements and the features in combination with the knowledge of process connectivity, process topology, and digraph-based models for variable causality. Root cause analysis is often applied off-line to the faulty data collected during process operations.

On the other hand, Chiang et al. (2000a) proposed a way to quantify the contributions of process variables to a fault using the monitoring statistics, the monitoring models, and the original measured data. The contributions of process variables can be calculated in real-time along with the monitoring statistics and variables with large contributions when a fault occurs are likely to be responsible for this fault. There are various ways to define the contribution of process variables. For example, Cho et al. (2005) proposed the contribution rate as the first order derivative of the monitoring statistics with respect to each process variable. This reflects the rate of change in monitoring statistics caused by the change in each process variable. Alcala and Qin (2009) proposed reconstruction-based contribution for fault diagnosis in nonlinear processes. Wang et al. (2017) and Shang et al. (2019) also came up with revised definition of contributions such that the cause variable can be highlighted while irrelevant or unaffected variables can be down-weighted. The definition of contributions may also be specific for the feature extraction approach, such as the contribution plots for the CVA defined by Jiang et al. (2015). By applying root cause analysis and contribution analysis, one may be able to locate the fault in the process.

Another task is to classify the type of fault. In order to do so, faulty data from various faulty scenarios are needed in the training step. Previous works, such as Lu et al. (2018), Jiang et al. (2019), and Lucke et al. (2020), utilized classification techniques to classify the faulty data and diagnose the type of the fault.

The work in the thesis focuses on the contribution-based fault diagnosis. Since contribution analysis does not require additional process knowledge, it can be done in real-time

in order to flag relevant variables with respect to the fault occurrence to the end-users as soon as a fault is detected. In addition, the contribution plots, which visualize the contributions made by the process variables to the monitoring statistics (e.g. Fig. 2.7), may provide additional reference for the end-users. Nevertheless, the idea of contributions of process variables is proposed based on the magnitude of monitoring statistics (Miller et al., 1998). Therefore, when the monitoring statistics are sensitive to mode changes, the contributions of process variables to these statistics will also be sensitive to mode changes. Fig. 2.7 gives an example that contribution-based fault diagnosis may be difficult when a process has multiple operating modes because the magnitude of the monitoring statistic changes as the operating mode changes. Therefore, another task of the thesis is to define contributions of process variables on the basis of the monitoring statistics for multimodal process monitoring.

3.3.4 Delivering fault detection and diagnosis results to end-users

The objective of deploying monitoring methods is to present the monitoring results to end-users to support operational decisions. The design of monitoring methods needs to take the needs of end-users into account. For example, when constructing the covariance matrices discussed in Section 3.3.1, it is important to take the specific objective of process monitoring into consideration so that the extracted features can satisfy the need of the end-users.

Furthermore, the presentation and delivery of monitoring results are equally important as the accuracy of monitoring models. This section will review the delivery of the fault detection and diagnosis results to end-users.

Table 3.3 summarizes the indicators used for fault detection according to the methods reviewed in previous sections. Control charts use quality-related variables as the indicator and use statistics derived from these variables as the control limits. MSPM methods generate monitoring statistics based on process data as the indicator of faulty behaviours. Residuals that quantify the mismatch between models and process data are used as the indicator by model-based monitoring methods. The probabilistic and Bayesian methods calculate the probability of the process being healthy. From the perspective of presentation, all four types of indicators can be visualized in the format of control charts along with their control limits. Operators will inspect the control charts to monitor if the indicator exceeds its

Table 3.3: Delivery of results in fault detection

Indicator	Remarks	Interpretation when exceeding control limits
Quality-related variables	Used in univariate control charts in industrial SCADA systems.	A single quality-related variable is exceeding its healthy range of operation.
Monitoring statistics calculated from process data	Calculated for multivariate control charts or by MSPM approaches.	The process is not operating in healthy operating conditions or the process model has changed.
Residuals	Used to quantify the model-data mismatch; the model may be obtained by first principles or data-driven approaches.	The process data are not consistent with the process model.
Probabilities	Obtained by probabilistic and Bayesian methods.	The process is unlikely to be operating at healthy conditions.

control limits. The situation where an indicator exceeds its control limits means that the process has become abnormal.

Table 3.4 presents the frequently used diagnosis results. Typical fault diagnosis results include the type of the fault and the location in the process where the fault originated. Monitoring methods can also diagnose which fault the process has if these methods can incorporate faulty data collected when various faults exist in the process. Probabilistic extensions of such methods may further provide the probabilities of the process running in specific faulty modes. Gertler (2017) defined *fault isolation* as the procedure of determining the exact location of the fault and the component which is faulty. In process monitoring, these tasks are achieved by identifying process variables that are likely to be the cause of the fault.

It is always important to deliver the monitoring result in a clear way such that end-users can easily interpret. For example, the undesirable monitoring statistic presented in Section 2.4.2 will not be useful for operators. Therefore, it will be insufficient to only develop advanced data-driven techniques without considering the influence of the new techniques on the result presentation and delivery. The thesis will develop MSPM methods for multimodal process monitoring that are can build accurate monitoring models for multimodal data whilst presenting monitoring results clearly to end-users.

Table 3.4: Delivery of results in fault diagnosis

Objectives	Result	Remarks	Interpretation
Fault identification	Mode identity	Obtained by classification methods, fault tree-based methods, and maximum likelihood-based probabilistic methods	The process has this specific fault.
	Probabilities	Obtained by probabilistic and Bayesian methods	The process has this specific fault with this level of probability.
Fault isolation	Root cause	Obtained by causality methods, such as causal maps and Bayesian networks.	The root cause of the faulty scenario is this piece of equipment or this process variable.
	Contributions	Calculated by contribution methods in MSPM.	This process variable has the largest contribution when the fault occurs. Hence it is related to the fault.

3.4 Kernel-based methods for the multimode problem

This section first reviews the state-of-the-art in the monitoring of multimodal processes. The section then introduces kernel methods and justifies the application of kernel-based MSPM methods. Moreover, open research questions existing in improving the performance of kernel-based MSPM approaches for multimodal process monitoring are identified.

3.4.1 State-of-the-art in multimodal process monitoring

The recent review paper Quiñones-Grueiro et al. (2019) on multimode process monitoring has divided methods for multimodal process monitoring into three categories:

1. A single monitoring model;
2. Multiple models with model selection;
3. Multiple models with Bayesian fusion.

Fig. 3.8 compares the procedures of the methods in the three categories. The single model method builds one monitoring model for the overall multimodal data and detects the fault using this model. Adaptive techniques have been adopted in this category. Ma et al. (2013)

introduced the nearest neighbour method and Fazai et al. (2016) used a forgetting factor to adapt the monitoring model to the current operating mode in the process. Zhang et al. (2018) proposed another way to build monitoring models by separating the mode-irrelevant behaviour, such as the connectivity of the process which does not often change with respect to operating modes, and the mode-dependent behaviour in the data. The multiple model approach with a decision step has been popular because a variety of modelling techniques can be adopted simultaneously for different modes under this framework. Examples can be found in Chen and Liu (1999), Zhu et al. (2012) and Afzal et al. (2017). Similarly to the GMM-based method, Feital et al. (2013) and Zhou et al. (2018) combined a multiple model approach with a Bayesian fusion step. They built localized models for each mode and constructs the mixture model under the Bayesian framework.

Quiñones-Grueiro et al. (2019) compared the three procedures and concluded that the single model approaches have several drawbacks over the multiple model approaches, such as the computational cost and the difficulty in pre-processing. Nevertheless, Fig. 3.8 visualizes the three procedures with the fault detection results delivered to end-users from each. It can be observed that the single model layout can provide end-users with a single control chart. A single control chart is the most straightforward for end-users to interpret. In contrast, the layout of multiple models with a decision step may first require a classification step to determine the operating mode of the process and the corresponding control chart to be used. Although several control charts for multiple modes may use the same monitoring statistic, there is no guarantee that the same magnitude of this statistic can have the same meaning because the monitoring models are different. Moreover, the monitoring performance will also rely heavily on the classification step.

As for the layout with multiple models and a Bayesian fusion step, the same issue exists such that the monitoring statistics obtained from different modes using different modelling approaches may be have different magnitudes. Therefore, it may not be reasonable to fuse several monitoring statistics with various magnitudes. Moreover, due to the fusion step, it may be difficult to further diagnose the fault because it will be difficult to interrogate such a model to discover which variables have contributed to the monitoring statistic.

If all the three procedures can achieve monitoring models with a similar level of accuracy, the single model approach may be preferred in consideration of real-life application and delivery if results to end-users because this procedure generates a single control chart that is easy for end-users to read. The thesis will focus on the improvement of the monitoring

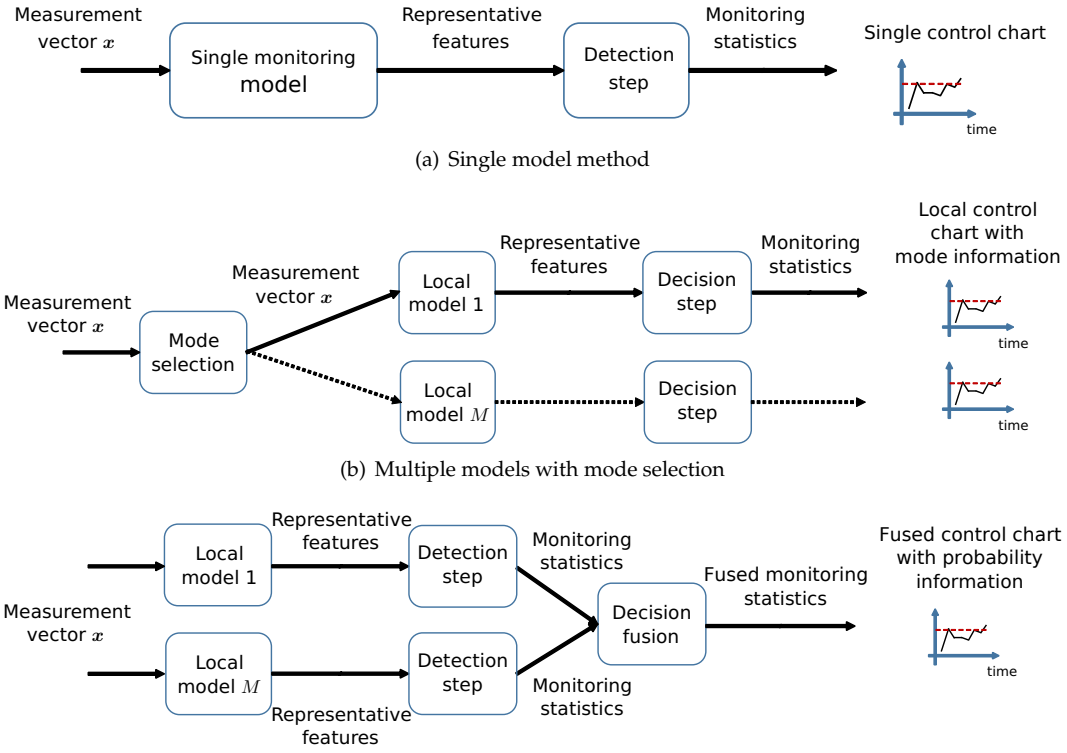


Figure 3.8: Procedures for multimode process monitoring algorithms

techniques to build an accurate overall monitoring model for multiple operating modes in the process.

3.4.2 Kernel methods for multimodal process monitoring

Kernel-based MSPM methods

In Section 3.3.1, it has been shown that the covariance matrices Σ used for eigenvalue decomposition differentiate between various linear feature extraction algorithms. However, if the relationship between the original process variables is nonlinear, the covariance matrix may not be representative of the data relationship and the concept of correlation and covariance must be extended to the unknown feature space. A feature space is a variable space consisting of the features that will be extracted by the feature extraction algorithm. The rest of this section gives a mathematical description of kernel methods.

The dot product of two zero-mean vectors is equal to the covariance of these two vectors. Boser et al. (1992) defined the dot product in the feature space as a kernel function in the variable space. A *kernel* function in the variable space X is a function defined for a pair

of vectors in X. The kernel function maps $X \times X$ to real values $\mathbb{R}$. Schölkopf et al. (1998) incorporated the kernel concept in the PCA formulation to create the Kernel Principal Component Analysis (KPCA), where the kernel Principal Components (PCs) can be extracted and the data nonlinearity can be captured by the kernel PCs. In the field of process monitoring, KPCA has also been widely adopted for handling data nonlinearity in fault detection (Lee et al., 2004a; Ge et al., 2009; Li and Yang, 2015; Jiang and Yan, 2018).

KPCA first projects the measurement vector $\boldsymbol{x}$ to a higher-dimensional nonlinear variable space Φ. This means that a vector of measurements $\boldsymbol{x}$ undergoes a mapping to the feature space $\Phi(\boldsymbol{x})$, $\boldsymbol{x} \mapsto \Phi(\boldsymbol{x})$, where $\boldsymbol{x} = [x_1, x_2, \ldots, x_m]$ and $\Phi(\boldsymbol{x}) = [\phi_1(\boldsymbol{x}), \phi_2(\boldsymbol{x}), \ldots, \phi_\infty(\boldsymbol{x})]$. The kernel features $\Phi(\boldsymbol{x})$ are the projected variables in the new feature space. Although $\Phi(\boldsymbol{x})$ cannot be calculated directly, $K \in \mathbb{R}^{n \times n}$, the covariance matrix of $\Phi(\boldsymbol{x})$ can be obtained using the kernel function.

Instead of assuming functional structures with respect to x for these nonlinear variables, kernel functions are defined for $\boldsymbol{x}$ in order to obtain K directly, the covariance matrix of unknown nonlinear variables. For example, K is defined by the RBF kernel function for i-th and j-th samples of $\boldsymbol{x}$ in Eqn (3.11):

$$K_{i,j} = k\left(\boldsymbol{x}(i), \boldsymbol{x}(j)\right) \tag{3.11}$$

where k is the user defined kernel function. Schölkopf et al. (1998) proved that PCA can be applied to K and feature extraction may be realized in the nonlinear variable space by solving the following eigenvalue problem. Assuming n samples of $\boldsymbol{x}$ are available for training, the eigenvalue problem is written as

$$\begin{aligned} n\lambda\boldsymbol{\alpha} &= K\boldsymbol{\alpha} \\ \boldsymbol{z} &= \sum_{i=1}^{n} \alpha_i \boldsymbol{k}_i \end{aligned} \tag{3.12}$$

where $\boldsymbol{\alpha} = \{\alpha_1, \ldots, \alpha_n\}$ and $\boldsymbol{k}_i = \{K_{1,i}, K_{2,i}, \ldots, K_{n,i}\}$. The representative features $\boldsymbol{z}$ are extracted from the nonlinear variable space using K in KPCA. Similarly to PCA-based MSPM, these features are further divided into kernel PCs and residuals according to their eigenvalues and the monitoring statistics are calculated accordingly. The monitoring statistics and their control limits delivered by KPCA resemble those from PCA. Choi and Lee (2004) provided a detailed description of the KPCA-based MSPM approach.

In general, Ge et al. (2013) mentioned that kernel-based MSPM methods are adopted in MSPM in order to handle process nonlinearity. Previous works, such as Lee et al. (2007), Zhang et al. (2007), Hu et al. (2013) and Luo et al. (2015), extended linear MSPM approaches to nonlinear process monitoring by replacing the covariance matrix by the kernel matrix. As for the user-defined kernel function, although a variety of kernel functions exist (Wilson, 2014), the literature survey in Pilario et al. (2020) showed that most of the works in kernel-based process monitoring use the following Radial Basis Function (RBF) kernel since it can capture the nonlinear relationship between variables: The RBF kernel of two variables $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ is defined as:

$$k_{\text{RBF}}(\boldsymbol{x}_1, \boldsymbol{x}_2) = \exp\left(-\frac{1}{\delta^2}(\boldsymbol{x}_1 - \boldsymbol{x}_2)^\top(\boldsymbol{x}_1 - \boldsymbol{x}_2)\right) \tag{3.13}$$

where δ is the kernel width that regulates the behaviour of the kernel and the kernel-based methods. The thesis will first consider if the RBF kernel-based methods with a well-tuned kernel width can fully solve the multimodal process monitoring problem. If the RBF kernel has limitations when applied to multimodal process monitoring, the next step will be developing a new kernel that may overcome these limitations.

Motivation for the use of kernel methods

The thesis proceeds with kernel-based MSPM methods for the following reasons.

First, as discussed in the previous section, a single monitoring model for multimodal processes can provide straightforward fault detection results to the end-users. However, Hwang and Han (1999) demonstrated that linear methods may be insufficient for multimodal processes. As a consequence, Zhao et al. (2015) also used linear methods under a multiple model framework. Moreover, the process variables in real-life processes, such as the opening of a valve and the flow rate passing through this valve, may often be nonlinearly correlated. Hoffmann (2007) provided examples in other fields of research where the kernel-based methods can handle general nonlinearity in the data and have performed well in anomaly detection.

The papers by Choi and Lee (2004) and Willis (2010) gave examples of extending feature extraction methods in MSPM to kernel-based methods by replacing the covariance matrices with the corresponding kernel matrices in their formulations. The fault detection procedure

using monitoring statistics in these examples remains the same as the procedure presented in Fig. 3.8(a).

Moreover, one of the tasks of the thesis is to investigate the behaviour of monitoring statistics and to select the monitoring statistic that is robust to mode changes and sensitive to fault occurrence and increases monotonically as the fault develops. In order to evaluate a monitoring statistic, the change of the statistic in reaction to the development of the fault should be analysed mathematically. Such analysis is possible when applying kernel-based feature extraction methods. In contrast, while a sophisticated multi-layer neural network model may be more accurate, the analysis of the corresponding fault indicator may be intractable.

3.5 Open questions and technical tasks of the thesis

The previous sections discussed why the thesis develops kernel-based methods for multimodal process monitoring. This section will review the open questions in the application of kernel-based methods to multimodal process monitoring. The open questions include how to achieve the desirable behaviours presented in Section 2.4 and to address the challenges identified in Section 2.6.

3.5.1 Open questions

Proper tuning of kernels

The kernel width of an RBF kernel is a tuning parameter for the kernel-based methods using RBF kernels. The kernel width influences the performance of kernel-based methods. However, the tuning of the kernel width in RBF kernels still remains an open question and various approaches have been proposed. Empirical values (Navi et al., 2018) and empirical equations (Lee et al., 2007) have been used for selecting the kernel width. Researchers have also adopted more systematic approaches, such as the cross-validation approach (Choi et al., 2005a) and optimization of the correct fault detection (Jia et al., 2012; Lahdhiri et al., 2019). To demonstrate the influence of the tuning, He et al. (2018) and Pilario et al. (2019) carried out empirical comparisons of the performance of the kernel-based methods with various values of kernel widths.

There are some research works focusing on the theoretical analysis of the influence of kernel widths. For example, Twining and Taylor (2003) and Keerthi and Lin (2003) have proven that exceedingly large kernel width will make the RBF kernel reduce to a linear kernel. Recently, it is pointed out by Pilario et al. (2019) that the RBF kernel will approach to zero when the faulty sample is very far away from the training data. In other kernel-related fields of research, such as the support vector machine, Keerthi and Lin (2003) conducted asymptotic analysis on the kernel-based methods with RBF kernels. In order to achieve an accurate monitoring model, the thesis will first investigate the influence of the kernel width and propose an appropriate tuning strategy.

A better kernel for the multimodal data

Although the RBF kernel has proven ability for modelling nonlinear data, many previous works, including Deng et al. (2017), Li et al. (2017), and Yu et al. (2018), have claimed that it may not be sufficient for multimodal process monitoring. Besides the potential tuning issue, the RBF kernel may also have limitations when dealing with multimodal data. One of the features of the multimode problem is that the correlation structure of process variables may be different for different operating modes. Similar phenomena have been seen in the field of natural language modelling (Garg et al., 2018), geostatistics (Higdon, 1998), and terrain surface estimation (Singh et al., 2016). The RBF formulation in Eqn (3.13) shows that it is a stationary kernel. The value of a stationary kernel function only depends on the distance between the two input data samples regardless of the location of the two samples. However, in a multimodal process, the correlation structures for two pairs of samples with the same distance in between might be different if the samples are collected from two different operating modes. Therefore, the RBF kernel may not be able to fully capture the varying relationship between process variables caused by multiple operating modes, making RBF kernel-based MSPM approaches incapable of building a single monitoring model.

A kernel should be able to adapt to the varying relationship in several operating modes when used for multimodal process monitoring. Amari and Wu (1999) and Paciorek (2003) designed other types of kernels which are data-dependent and non-stationary. Gönen and Alpaydin (2008) also proposed to use a combination of multiple kernels. Luo et al. (2015) and Pilario et al. (2019) applied the new kernels and the combination of kernels to process

monitoring. However, the authors did not study the performance of these kernels on multimodal data. A major task of the thesis is to evaluate the performance of the RBF kernel-based method in monitoring multimodal processes, identify the limitations, and propose a new kernel that is more suitable than the RBF kernel for describing multimodal data.

The behaviour of monitoring statistics

Under the linear and Gaussian assumption of process variables, Wise and Gallagher (1996) and Qin (2003) investigated the statistical distribution of T^2. Similarly for the squared prediction error, Jackson and Mudholkar (1979) gave the function of setting upper control limits when using SPE. Recent works, such as Tong et al. (2017) and Sánchez-Fernández et al. (2018), still used these monitoring statistics and their control limits. Li et al. (2017); Zhang et al. (2017); Pilario and Cao (2018) used other monitoring statistics that are suitable for specific feature extraction approaches applied.

Odiowei and Cao (2009) reported that, when the Gaussian assumption of the extracted features is no longer valid, for example when using the independent component analysis, the χ^2 assumption of T^2 may not be valid. Instead, Odiowei and Cao (2009) proposed to set the control limit of T^2 by kernel density estimation.

Ge et al. (2009), Alcala and Qin (2010), and Deng et al. (2018) used the standard Hotelling's T^2 as the monitoring statistic for kernel-based methods in the same way as T^2 is used for linear methods. Chakour et al. (2018), Deng et al. (2018), and Pilario et al. (2019) defined the SPE in various ways and used it for fault detection in kernel-based methods. However, monitoring statistics achieved by kernel-based methods may not have the same behaviour as the statistics obtained by linear methods. Thus, the use of monitoring statistics may be different in kernel-based methods. Recently, Pilario et al. (2019) proved that the T^2 monitoring statistic will converge to a constant value if a faulty sample moves infinitely far away from the healthy samples for KPCA with RBF kernels. Therefore, the χ^2 assumption for T^2 will be challenged because random variables with χ^2 distributions take values from zero to infinity. Moreover, such behaviour may also undermine the monotonically increasing assumption, making it questionable if T^2 changes monotonically as a fault develops. A task of the thesis is to analyse the behaviour of the monitoring statistics from a theoretical perspective and to propose a desirable monitoring statistic for kernel-based methods.

Table 3.5: Summary of technical objectives and development in each chapter

Ch.	Technical objectives	New development
5	Tuning and monitoring statistics	Asymptotic analysis
6	Modelling of multimodal data	A new kernel
	Fault detection in new modes	An on-line monitoring framework
7	Identification of influential variables	A new type of contribution plots

Contribution concepts in kernel-based methods

Miller et al. (1998), Chen and Sun (2009), and Alcala and Qin (2010) proposed various definitions of contributions of variables for dealing with various MSPM problems. However, the assumption for defining and interpreting the contribution concepts may be similar. In general, the contribution of a variable is defined as a metric of the influence of the measured value of a particular variable on the final monitoring statistics. When a fault occurs, the monitoring statistic exceeds its control limit. If one process variable has a large contribution to the monitoring statistic, it is more likely for this variable to be related to the fault. The assumption behind such a conclusion is that the monitoring statistic will react to the fault, change monotonically and exceed its control limit only when the fault occurs and develops. However, in multimode process monitoring, this assumption does not hold for a monitoring statistic which is sensitive to mode changes, as shown in Chapter 2.6. Consequently, the magnitudes of contributions of variables defined for such monitoring statistics may not be comparable with each other. Hence the objective of defining contributions of variables is to achieve a monitoring statistic that reacts to the fault occurrence whilst being robust to the mode changes. The thesis will further investigate the contribution-based diagnosis after obtaining a desirable fault detection framework.

3.5.2 Technical objectives and development of kernel methods

The aim of the thesis is to develop data-driven process monitoring approaches that cope with the challenges posed by multiple operating modes, achieve more accurate and clear evaluation of the process operating conditions, and deliver the results clearly to end-users. Sections 2.4 and 2.6 in Chapter 2 discussed desirable behaviours of monitoring systems and the challenges posed by multiple operating modes. From the technical perspective, Section

3.5.1 discussed the open questions existing in monitoring methods when applying these methods to multimodal process monitoring.

The first research question stated in Section 1.1.3 is addressed in Chapters 2 and 3. Section 2.1 briefly introduced the decision making procedure of process operators. Section 2.4 presented the expectation of operators for process monitoring and Section 2.7 outlined the requirements that a monitoring system should fulfill in order to provide useful information for operators. Section 3.3.4 discussed how a process monitoring system should present useful information to an operator.

Section 3.4 addressed the section research question in Section 1.1.3 by evaluating existing kernel methods. The conclusion is that kernel methods can be suitable for monitoring process with multiple operating modes if a new kernel is developed for addressing the multiple modes and the associated kernel method is configured properly.

Based on the discussions in previous sections, the third research question stated in Section 1.1.3 can be decomposed into several technical objectives. Kernel methods will be developed in order to achieve these technical objectives such that kernel methods need to do to be suitable for multimodal process monitoring. Table 3.5 specifies the technical objectives and the new development and configuration of kernel methods in Chapters 5 to 7, respectively. Chapters 6 and 7 then answer the fourth research question, regarding how kernel methods can formulate part of a monitoring system.

3.6 Chapter summary

This chapter presented a top-down overview of process monitoring methods, data-driven methods, multivariate statistical process monitoring methods, kernel-based methods, and the techniques applied to multimodal process monitoring.

By reviewing the monitoring methods, the following open questions in applying kernel-based methods for multimodal process monitoring were identified:

1. What is the optimal method for tuning kernel parameters?
2. Is there a kernel suitable for multimodal process monitoring?
3. How do monitoring statistics behave when kernel methods are applied?

4. How to apply contribution-based fault diagnosis in kernel methods?

This chapter then proposed the following technical tasks to address these open questions.

1. Developing methods for tuning kernels and investigate the behaviour of monitoring statistics;
2. Create an approach for modelling multimodal data;
3. Enable monitoring models to detect faults in new operating modes;
4. Propose a method to identify influential variables when kernel methods are used.

The technical chapters of the thesis will address these open questions individually.

Chapter 4

The PRONTO dataset

This chapter presents the PRONTO benchmark dataset. This dataset, obtained from a physical pilot plant, may be used for developing and verifying process monitoring methods. As previously introduced in Chapter 2, the dataset comprises heterogeneous types of data collected from various sources. The experiment was conducted in the Process System Engineering Laboratory of Cranfield University together with Anna Stief, a fellow early stage researcher in the PRONTO project.

This chapter describes the background and context for creating this dataset, the layout of the test rig, the design of the experiment, and the data collected during the experiment. The chapter then reviews the usage of the dataset in the thesis, followed by the chapter summary.

4.1 Background

When evaluating data-driven monitoring methods for multimodal processes, the ability of these methods to address the challenges posed by multimodality needs to be verified. Testing the data-driven methods requires benchmark datasets that allow direct comparison of the performance of the methods. The Tennessee Eastman process plant (Ricker, 1995) is a widely used simulator for generating datasets for validating different control and monitoring methods. Odgaard et al. (2013) set up the wind turbine competition benchmark model where the dataset was used for the comparison of fault detection and diagnosis

methods. Van Impe and Gins (2015) provided another benchmark case with a reference dataset for fault detection and identification in batch processes. These benchmark case studies are based on simulations and do not include real-life measurements. In contrast to these case studies based on simulated data, industrial-scale case studies like the multiphase flow benchmark case study in Ruiz-Cárcel et al. (2015) or the carbon capture case study in Kachko et al. (2015) do provide measurements recorded from a real-life system. Some of the datasets, such as the dataset presented by Ruiz-Cárcel et al. (2015), include data from multiple operating modes. However the issue of multiple operating modes is not the focus of these datasets. The data-driven methods developed in the thesis require a dataset for validation where the multimodal behaviour is considered explicitly.

The PRONTO dataset provides real-life data for validating the data-driven methods proposed in the thesis. The experiment on a pilot-scale plant covered a variety of operating conditions. The study collected data from various operational conditions with and without artificial faults to generate a multimodal dataset with data from various sources. The case study was one of the deliverables of the PRONTO [1] project and is a collaborative work of the researchers in this project. The dataset is publicly available in the repository Stief et al. (2019b). A paper (Stief et al., 2019c) describes the benchmark case study and the dataset in detail. The work in this chapter was done in collaboration with Anna Stief from ABB Corporate Research Center in Krakow, Poland. Anna and the author of this thesis collaborated in design and conducting the experiment, collecting the data, preparing the paper, and releasing the dataset to public.

4.2 Process description

The experimental facility used in the experiment is a pilot-scale multiphase flow rig located at the Process System Engineering lab at Cranfield University, UK. This rig demonstrates the mixing, transportation, and separation of multiphase flow with air, water and oil. Such layouts often exist in platforms for off-shore oil production. Fig. 4.1 is the Piping and Instrumentation Diagram (P&ID) for the test rig. It shows the connectivity of the process equipment. The air compressor compresses air from atmosphere and supplies the air flow. The water tank and the oil tank store water and oil. Two pumps transport water and oil

[1] PRONTO (PROcess NeTwork Optimization for efficient and sustainable operation ofEurope's process industries taking machinery condition and process performance into account)

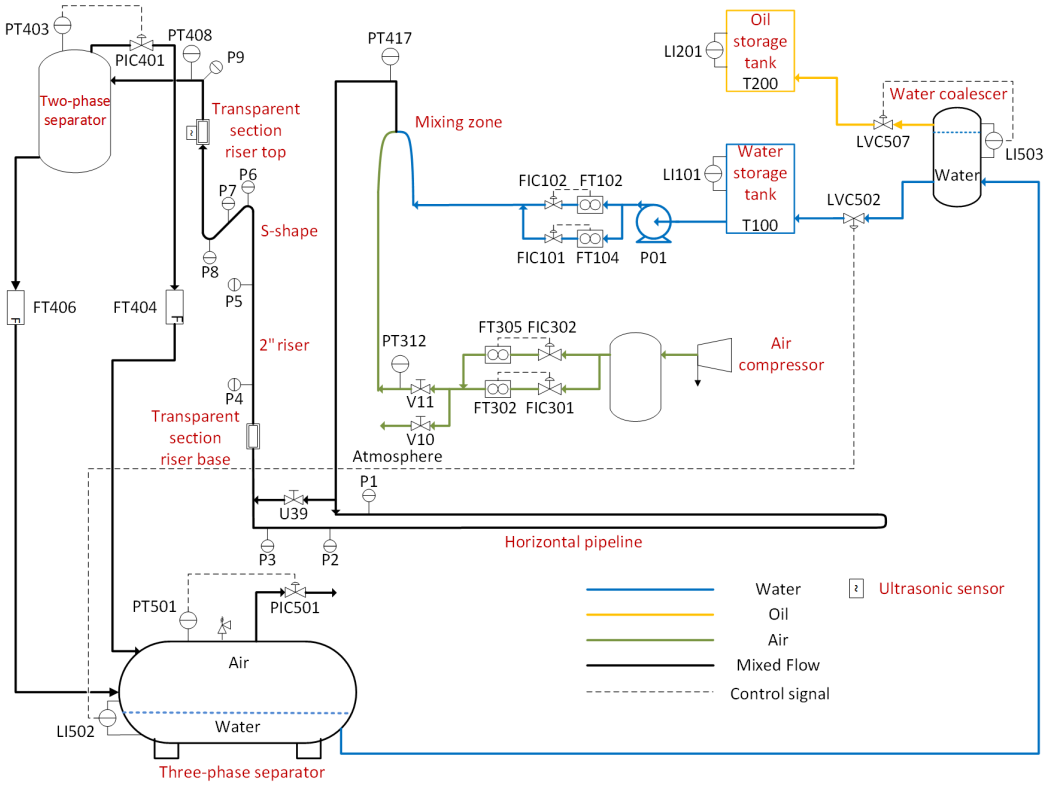


Figure 4.1: P&ID of the test rig from the Process System Engineering lab at Cranfield University (Stief et al., 2019c)

from the storage tanks. In this experiment only the water supply was used. Air and water are mixed in the mixing zone, then enter the horizontal pipeline. Due to the high pressure introduced by the air compressor, the multiphase flow can go up to the top of the two-inch riser with an S-shape section. Two transparent pipelines were installed in the riser for visually inspecting the flow regimes. After the riser top, a two-phase separator separates the multiphase flow. The separated air and water flows then enter a three-phase separator via two individual pipes for further separation. The air is released to atmosphere and the water returns to the water tank by going through a water coalescer. The rig is useful for investigating topics such as the metering and the control of multiphase flow, the influential factors in flow regimes, and process data analytics.

Fig. 4.1 also shows the instruments for measuring process variables. The instruments enable data collection in this rig. The experiment for generating the PRONTO dataset used the rig to generate data in several operating conditions, both with and without faults. These data are suitable for developing and validating data-driven process monitoring methods.

		Water flow rate (kg s^{-1})				
		0.1	0.5	1	2	3.5
Air flow rate (Sm^3 h^{-1})	20	slugging	slugging	slugging	slugging	healthy
	50	slugging	slugging	slugging	healthy	healthy
	100	healthy	healthy	healthy	healthy	healthy
	120	healthy	-	-	-	-
	150	-	healthy	-	-	-
	200	healthy	healthy	healthy	-	-

Table 4.1: Operating modes tested in the experiment

Section 4.3 will review the tested scenarios. Section 4.4 will introduce the data acquisition systems and will provide detailed descriptions of the data types collected during the experiment.

4.3 Experiment design

This section reviews the tested scenarios for generating the dataset. In particular, since the thesis will use this dataset for testing data-driven monitoring methods for mutlimodal processes, the multiple operating conditions and the artificial faults are discussed. An *artificial fault* means that the experimenters induced process conditions that imitated faults. An example is the artificial blockage in the input air line induced by closing the inlet valve V11 in Fig. 4.1.

4.3.1 Multiple operating conditions

The flow regime in the rig depends on the input air and water flow rates. Each flow regime results in a different operating condition. Each operating condition is described by a different model. Table 4.1 summarizes the operating conditions specified by various input air and water flow rates to the rig.

By varying the flow regimes it is possible to simulate healthy and fault conditions because some of the flow regimes are inherently faulty. An example is the slugging condition reported by Jansen et al. (1996). Slugging is an unstable flow regime that occurs in multiphase risers when the gas and liquid flow rates are relatively low. In Table 4.1, "healthy" means that the condition specified by the given water and air flow rates has a normal flow regime

(a) Valve V11 for air blockage (b) Valve V10 for air leakage (c) Upper valve U39 for diverted flow. Lower valve opens in all operations.

Figure 4.2: Manual valves for inducing artificial faults

while "slugging" represents the conditions where slugging occurs due to insufficient water or air supply. The procedure of the experiments was to specify set points for input air and water flow rates, then to wait until the process reaches its steady state. The researchers carrying out the experiments switched the process to another operating mode after collecting sufficient measurements. Thus the dataset contains the measurements from these operating modes and the transitions between them.

4.3.2 Artificial faults

During the experiment, several artificial faults were induced in the rig during process operations. In order to mimic the situation when faults occur in multiple operating modes, all of the artificial faults, except the slugging fault, were tested in two healthy operating conditions. The water flow rate was 0.1 kg s^{-1} and the air flow rate was 120 Sm^3 h^{-1} for the first operating condition. The water flow rate was 0.5 kg s^{-1} and the air flow rate was 150 Sm^3 h^{-1} for the second condition.

Air blockage

The air blockage fault was induced by gradually closing V11, the manual valve on the input air line, as shown in Fig. 4.1. Fig. 4.2(a) gives a photo of V11. The valve opening was reduced from 90° to 10°. This fault causes reduced air supply to the mixing zone and

the transportation and the separation in succession. It was observed during the experiment that insufficient air supply created mild slugging in the rig.

Air leakage

The air leakage fault was induced by gradually opening V10, which allows the input air to leak out to the atmosphere instead of entering the mixing zone. Fig. 4.2(b) is a photo of V10. The valve openings started at 0° and stopped at 15° and 25° in the two nominal conditions, respectively. As the amount of input air to the rig reduced, a periodic behaviour in the flow regime was observed. The cycle started with the period of regular flow, followed by a period of no flow at the riser top. Then a flow of water with large air bubbles appeared in the rig before the flow regime returned to regular. This behaviour was very similar to slugging, although the reason behind it was not only the reduced air flow rate but also the pressure drop in the mixing zone caused by the leakage.

Diverted flow

As shown in Fig. 4.1, the manual valve U39 connects the start and the end of the horizontal pipeline. In healthy operations, U39 is kept closed so that the multiphase flow will pass through the horizontal riser. The flow is diverted directly to the riser when U39 is opened. In the experiment, U39 was gradually opened to 60° to induce the diverted flow fault. The diverted flow fault caused visible differences in the flow regime when compared to the regular flow regime in healthy conditions at the transparent pipe at the riser base, however there was no difference observed at the transparent pipe at the riser top.

Slugging

As described previously, the slugging condition was induced by adjusting the input water and the input air flow rates in this experiment. Slugging is considered as a fault because it is unwanted in off-shore oil and gas production. Slugging often occurs when the pressure of the multiphase flow is insufficient. When slugging occurs, the pressure of the multiphase flow coming from the horizontal pipeline is insufficient to push the liquid to the riser top. Then the liquid builds up at the riser bottom, resulting in a liquid slug that blocks the gas flow. Thus the pressure in the horizontal pipeline increases at the riser bottom until it is

Table 4.2: Data availability in the experiment

Data type	Sampling rate	Storage policy	Platform
Process data	1 Hz	Regularly	DeltaV
Alarm records	Event-driven	Discrete events	DeltaV
High-frequency pressure	10 kHz	60s recordings	LabView
Doppler ultrasonic sensor	5 kHz	60s recordings	LabView
Process-specific information	-	-	Camera, manual logs

sufficient to push the liquid and the gas slug to the riser top. Part of the liquid then falls back down to the riser bottom due to the insufficient pressure. The cycle of slugging then starts again. Such periodic behaviour was also observed in this experiment. In industrial practice, slugging should be avoided because it often leads to significant fluctuations of the pressure and the flow rate in the riser, putting the equipment in the process at risk.

4.4 Data acquisition

Table 4.2 summarizes the data types and their availability from the experiment. These data constitute the PRONTO benchmark dataset which can be used for developing and validating data-driven fault detection and diagnosis techniques. This section reviews the instrumentation for data acquisition and the various data types collected by the instrumentation.

4.4.1 Instrumentation for data acquisition

The experiment and the benchmark dataset benefited from the measurement instruments available in the test rig. Fig. 4.4 shows examples of the sensors installed at various locations of the rig. Fig. 4.3(a) shows the level measurement of the three-phase separator, corresponding to the tag LI502 in Fig. 4.1. The level sensor has a local display in addition to the data recorded by the SCADA system. Fig. 4.3(b) is the high frequency pressure sensor P6. As shown in the photo, as well as the P&ID in Fig. 4.1, this sensor is located at the top of the S-shape riser. Fig. 4.3(c) is the photo of the riser top area, where a piece of transparent pipe was installed for observing flow regimes. The Doppler ultrasonic sensor was clamped on the transparent pipe and was connected to a desktop computer with LabView for data collection and storage.

(a) Level measurement of the three-phase separator LI502

(b) High frequency pressure sensor P6

(c) Riser top: transparent pipe and ultrasonic sensor

Figure 4.3: Measurement instrumentation in the test rig

4.4.2 Data types in PRONTO dataset

The PRONTO benchmark dataset is an example of observations from various sources in process monitoring. The dataset covers the following data types presented in Fig. 2.1 of Chapter 2: process measurements, electrical and mechanical measurements, alarm records, and process-specific information. The following sections review each type individually.

Process data

Process data consist of the measurements of process variables such as pressure, temperature, flow rates, and levels. In this experiment, the process variables were measured at the sampling rate of 1 Hz by the SCADA system for the whole duration of the experiment. A desktop computer located in the control room stored these measurement data. The following table summarizes the tags and the corresponding process variables. A *tag* is the code name of a measurement instrument in the SCADA system. For example, FT102 is the flow meter for the input water flow. FT102 also measures the temperature and the density of the inlet water. Therefore, FT102-T represents the input water temperature and FT102-D represents the input water density in Table 4.3. Valve openings of several control valves were recorded. The process data are suitable for validating data-driven process monitoring algorithms reviewed in Chapter 3. Extra information about the controllers is available in the PRONTO benchmark dataset. However it is not listed in Table 4.3 as it is not used in the thesis. Interested readers can find it in the data repository (Stief et al., 2019b).

Table 4.3: Process variables in PRONTO dataset

Tag name	Measured process variable	Unit
FT305/302	Input air flow rate	$Sm^3\ h^{-1}$
FT305-T	Input air temperature	°C
PT312	Air delivery pressure	barg
FT102/104	Input water flow rate	$kg\ s^{-1}$
FT102-T	Input water temperature	°C
FT102-D	Input water density	$kg\ m^{-3}$
PT417	Mix zone pressure	barg
PT408	Riser top pressure	barg
PT403	Top separator pressure	barg
FT404	Top separator outlet air flow rate	$m^3\ h^{-1}$
FT406	Top separator outlet water flow rate	$kg\ s^{-1}$
PT501	Three-phase separator pressure	barg
PIC501	Three-phase separator air outlet valve opening	%
LI502	Water-oil level in three-phase separator	%
LI503	Water coalescer level	%
LVC502-SR	Water coalescer outlet valve opening	%
LI101	Water tank level	%

High-frequency pressure measurements

In addition to the sensors for process measurements, high-frequency pressure sensors (P1 to P9 in Fig. 4.1) were distributed along the horizontal pipeline and the riser. These sensors were connected to another desktop computer with LabView in the control room. Since the sampling rate is 10 kHz, it is not possible to record the high-frequency measurements continuously due to the limited storage capacity. Instead, the samples were collected for a time window of 60 seconds when the process was stabilized in each operating mode with certain fault severity. The computer triggered the measurements and stored the data during the experiment. Table 4.4 describes the location of the high-frequency pressure sensors. Owing to the fast sampling rate, the pressure measurements might carry information about the process operating status that cannot be seen from the process data. For example, the cyclic behaviour of the flow regime when slugging occurs will cause cyclic changes of the pressure along the horizontal pipeline. The high-frequency pressure measurements can give information regarding the cyclic changes, especially when the period of the cycles is shorter than the sampling interval of the process measurements. Monitoring methods developed for electrical and mechanical measurements can be tested using this dataset.

Table 4.4: High-frequency pressure measurements in PRONTO dataset

Tag name	Location
P1	Before horizontal line
P2	After horizontal line, before riser base
P3	Riser base
P4	Vertical riser after the transparent pipe at riser base
P5	Middle of vertical riser, before S-shape riser
P6	Top of S-shape riser
P7	Middle of inclining part of S-shape riser
P8	Bottom of S-shape riser
P9	After S-shape riser, at riser top

Since there is no process measurement between the mixing zone and the riser top, these high-frequency pressure measurements are complementary to the process data.

Ultrasonic sensor reading

The Doppler ultrasonic sensor clamped on the transparent pipe at the riser top has two transducers. One transducer sends a high-frequency ultrasonic signal and the other transducer receives the signals reflected by the multiphase flow inside the pipe. The received signals have a Doppler shift of frequency and an output voltage of the sensor depends on the Doppler shift. The measurement frequency was 10 kHz and, similarly to the high-frequency pressure measurements, the measurement was recorded for 60 seconds when the process reached a steady state. Moreover, the timings of the measurements of the high-frequency pressure sensors were synchronized with those of the ultrasonic sensor during the experiment. As shown in Fig. 4.3(c), a desktop computer was set up at the riser top for triggering the ultrasonic measurements and collecting the data. It is possible to determine the flow rates of the multiphase flow in the transparent pipe based on the Doppler shift (Nnabuife et al., 2019). Thus the ultrasonic measurement is a fast-rate flow measurement and various signal processing techniques may be adopted to analyse these data.

Alarm records

It is often possible to set high and low limits for process variables in SCADA systems, especially for variables that are critical to the health and safety of the process. The SCADA system will trigger alarms if process variables exceed the limits set in the system. Other

Table 4.5: Examples of alarm settings in the SCADA system

Tag (unit)	HH[1]	H[2]	L[3]	LL[4]	Remarks
LI101 (m)	1.6	1.4	0.2	0.1	Water tank lever, scale between 0 to 2 meters
LI504 (m)	80	60	30	25	Oil-gas interface of 3-phase separator, 18.59%=5.95”
PIC501 (%)	100	95	-	-	3-phase separator outlet valve, not enabled

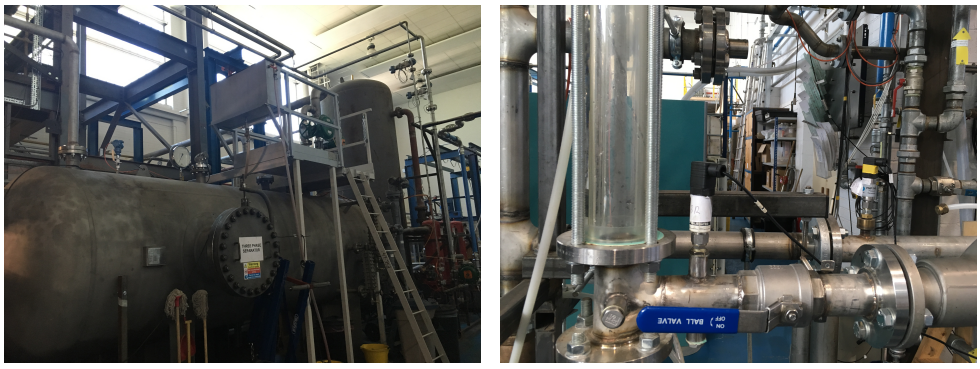
[1] High-High alarm;
[2] High alarm;
[3] Low alarm;
[4] Low-Low alarm.

types of alarms may flag failures in measurement instrumentation and communication modules. In this dataset, the alarm records during the experiment were retrieved from the SCADA system of the test rig. In addition to collecting sensor readings, the SCADA system recorded the alarms triggered by events occurring in the process and the changes made by the operator. A change in the pattern of alarms may indicate the occurrence of one or more faults. Lucke et al. (2019b) gives an example of such analysis.

The dataset also contains information related to the settings of the alarms, such as the alarm thresholds and the status of each alarm, collected manually from the SCADA system. Table 4.5 presents some examples of the alarm setting for several tags. Such information is useful for interpreting the alarms and diagnosing faults. Moreover, it is possible to generate alarms by setting additional alarm thresholds when analysing the process data. For example, Lucke et al. (2020) discussed the statistical alarms generated using standard deviations of process variables. The alarm thresholds may provide guidance for such analysis even if this alarm was originally disabled in the SCADA system.

Process-specific information

The experiment for generating this dataset is well-documented with process-specific information. For example, the P&ID in Fig. 4.1 provides information regarding the connectivity of the equipment in the rig. The operation log taken during the experiment reproduced records of the type that are often kept by process operators in real-life plants. The log tracks the time of actions, such as changing the set-points, adjusting the valve openings, and taking high-frequency measurements. Therefore, it is possible to use the log to label the data. i.e. to identify from which operating mode the process data were collected, and to synchronize the high frequency measurements. Other types of data are available, such as photos



(a) Three-phase separator

(b) Riser bottom: transparent pipe and high-frequency pressure sensor

Figure 4.4: Photos of process equipment

and video recordings. Fig. 4.4 presents two examples of the photos of process equipment and instrumentation. The photos of the process may help users to understand the scale of the process. The photos of process equipment used for fault generation can illustrate how the artificial faults were induced. The video recordings taken from the transparent pipes can facilitate the understanding of various flow regimes caused by varying water and air flow rates, which are examples of multiple operating modes in practice. The process data used in the thesis are labelled using the operation log and the video recordings.

4.5 Typical process data behaviour

The process data often reflect the characteristics and complexities of a process, which should be the guideline for the design of data-driven process monitoring algorithms. This section presents several examples of the behaviours of the process that are recorded by process data. These examples are based on the data plotted in Fig. 4.5. This figure shows the high density plot of the normalized process measurements collected in several healthy and slugging conditions. The observations from this subset of data will motivate the design of data-driven process monitoring methods.

4.5.1 Univariate behaviour

The measurement of any individual process variable is usually contaminated by noise and errors due to measurement instrumentation. Fig. 4.6 gives several examples. In Fig. 4.6(a), the temperature measurement of the input water is quantized. Fig. 4.6(b) shows that the

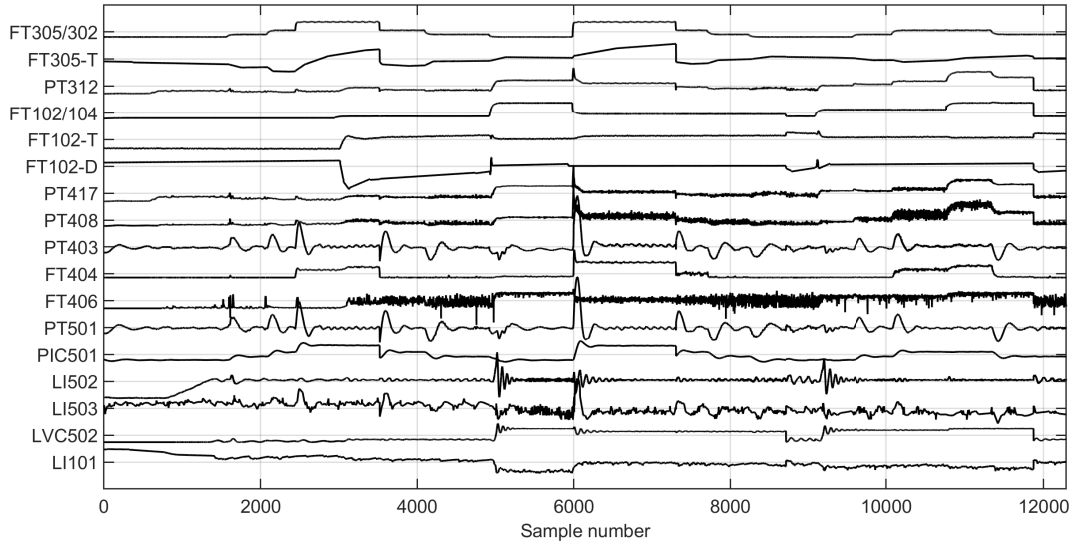


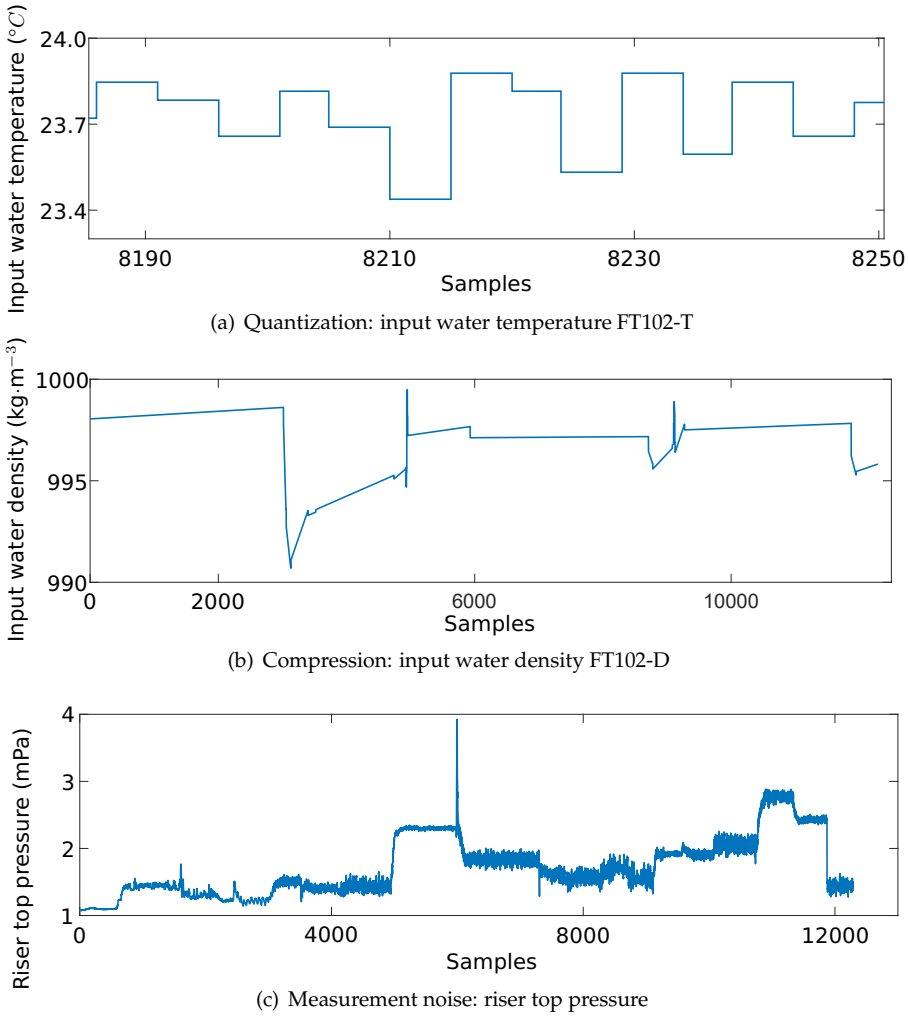
Figure 4.5: Time trends of the process variables in healthy and slugging conditions

density measurement of the input water has a clear compression effect, i.e. the measurements can only be collected when the change in the variable exceeds the resolution of the sensor and the samples between the two measurements are interpolated. Fig. 4.6(c) shows the time trend of the riser top pressure. The magnitude of measurement noise may vary depending on the operating mode.

4.5.2 Multivariate behaviour

Correlations usually exist between process measurements that are physically close to or connected with each other in the process. A process measurement may also be autocorrelated with its historical values. Such relationships reflect the underlying nature of the process, and faults may cause these relationships to change. Visualization of process measurements can indicate the relationships that may exist in the process, which can be traced back to the physical layout and operational history.

Figure 4.7 presents several examples of relationships existing among multiple process variables. Fig. 4.7(a) shows that the mixing zone pressure (PT417) and riser top pressure (PT408) are highly correlated, which can be explained by their proximate location and connection via pipelines in the facility. Figure 4.7(b) shows the process variables related to the water storage tank. The derivative of LI101, the water level in the water tank, should be proportional to the difference between the inflow rate, which is proportional to the outlet valve opening of water coalescer LVC502, and the outflow rate, which is the input water



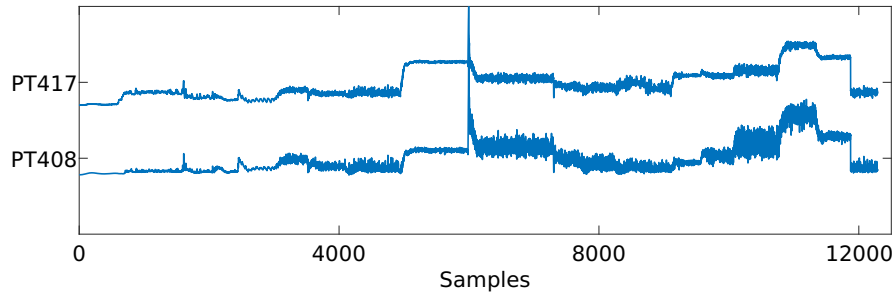
(a) Quantization: input water temperature FT102-T

(b) Compression: input water density FT102-D

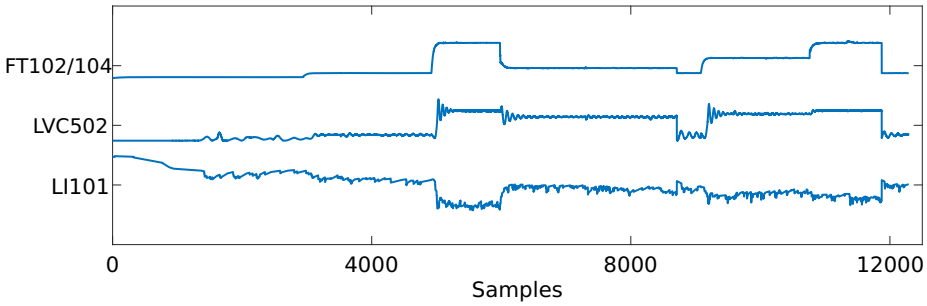
(c) Measurement noise: riser top pressure

Figure 4.6: Examples of noise and errors in process measurements

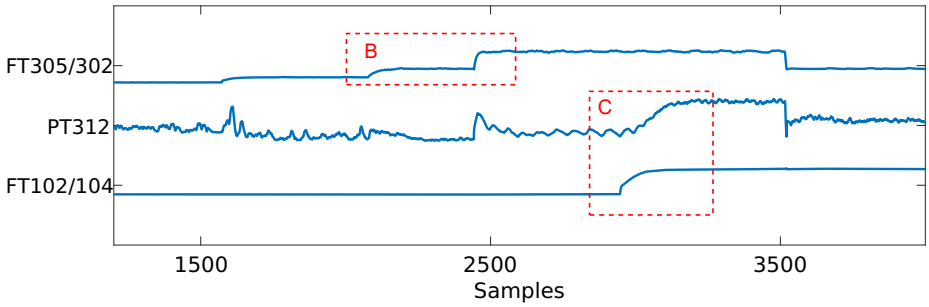
flow rate to the rig FT102/104, according to the mass balance. After changing the set points, transitions may occur before the process reaches the new steady state. Fig. 4.7(c) presents the step responses of the input air flow rate FT302/305, input water flow rate FT102/104, and the pressure in the input air line PT312. In particular, the area with dashed boxes "B" and "C" in Fig. 4.7(c) highlights the transition periods before the process reaches the steady state in the new operating modes. Oscillations in the measurements may exist due to the nature of the mechanical system, such as the vibration of the vertical riser, the control effect, and unstable flow regimes, e.g. slugging, in this case study. Fig. 4.7(d) shows PT501, a controlled variable, and PIC501, the controller output associated with PT501, with oscillations due to the control effect. Such oscillations may trigger alarms during the transition between healthy operating conditions.



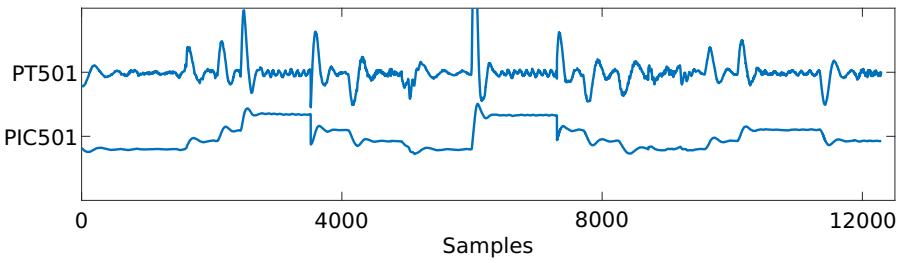
(a) Correlation between pressure variables



(b) First-principles relationship: mass balance in the water storage tank



(c) Step responses to changes of set points



(d) Oscillation: control effect in controlled variable and controller output

Figure 4.7: Examples of multivariate observations

The multiple operating modes will influence both univariate behaviours and multivariate behaviours of the process. In Fig. 4.6(c), the various operating modes result in various corresponding steady states of the pressure measurement. The magnitude of measurement noise depends on the operating mode. The transition periods in Fig. 4.7(c) demonstrate that the process variables may not always react in the same way to the change of operating

modes. For example, the input air pressure PT312 has step changes in response to the step changes in the input air flow rate FT305/302. However, in the area marked by "C", PT312 shows a time delay and a large time constant when the input water flow rate FT102/104 has step changes. The difference due to varying operating modes will have influence on the performance of monitoring systems. For example, Figs 2.6(a) and 2.7(a) in Chapter 2 are examples of fault detection and diagnosis results affected by the varying steady states of process variables. The development of data-driven monitoring methods for multimodal processes should address the multimodal behaviour existing in process data.

4.6 Usage of the dataset

The PRONTO benchmark dataset presents the complexities existing in real-life data from a pilot-scale process facility. This dataset can support researchers in developing and validating data-driven fault detection and diagnosis algorithms for various types of data. It has potential uses in demonstrating on-line implementation of monitoring systems. The process measurements in this dataset are particularly useful for the thesis in the following aspects:

4.6.1 Data pre-processing

Real-life data always need to be pre-processed before they are suitable for testing data-driven process monitoring methods. Data quality will influence the performance of the monitoring systems applied to real-life processes and the pre-processing step should improve data quality. For example, the pre-processing step needs to remove the measurement with compression issue shown in Fig. 4.6(b) because Thornhill et al. (2004) showed that the compressed data may cause challenges to data analysis. For instance, compressed data will give a different indication of the process behaviour than then original non-compressed data would have given. Measurement noise also exists in the dataset, especially in the high-frequency measurements. Hence the data may need to be filtered before using. Moreover, as shown in Table 4.3, the process variables have various units, which result in various scales of the numerical values of the variables and scaling may be needed to prepare the data for further analysis.

4.6.2 Multiple operating modes

The PRONTO benchmark dataset contains several operating modes specified by the input air flow rate and the input water flow rate. Therefore, these data are suitable for testing the methods proposed for multimodal processes.

4.6.3 Multiple process variables

The thesis focuses on multivariate statistical process monitoring. A major task of multivariate process monitoring is to monitor the mathematical relationship between multiple process variables. There are various types of relationships existing in multiple process variables in the PRONTO dataset and Fig. 4.7 gave several examples.

4.6.4 On-line implementation

Since the data were recorded continuously during the experiment, this dataset provides a comprehensive record of the operations and the changes in the process. It is therefore possible to use these data to show the performance of on-line monitoring systems, especially how these systems adapt to new operating conditions that do not exist in the data used for model training. This dataset can test the adaptation ability of on-line monitoring systems because it has multiple scenarios for healthy and faulty operation. Data from selected scenarios can be excluded during training on the monitoring system in order to provide an unseen scenario for testing the ability of monitoring methods to deal with unseen modes.

The technical chapters, Chapter 5 to Chapter 8, will use this dataset for validating and testing the monitoring methods proposed in these chapters.

This benchmark dataset may also contribute to other topics in fault detection and diagnosis. For example, Stief et al. (2018) fused the process data and the alarm data in the PRONTO dataset to diagnose the various types of faults. Stief et al. (2019a) investigated the sensor selection and the feature selection for fault diagnosis using the high-frequency pressure data. Lucke et al. (2019b) used mutual information for process variables and alarm records to select variables to be used in fault detection and identification. Moreover, the dataset may inspire ideas for innovative approaches that take into account important aspects of

real-life industrial data. The benchmark dataset can also be useful as a teaching resource as one may apply fundamental data analytic methods to this dataset.

4.7 Chapter summary

This chapter presented an overview of the PRONTO dataset, including the background for establishing such a case study, the layout of the test rig, the experiment design, and the data acquired during the experiment. The PRONTO benchmark dataset contains a variety of data types and this thesis focuses on the process data. The healthy and the faulty process data in the dataset are from multiple operating modes in the test rig, making the dataset a good candidate for validating the data-driven methods for multimodal process monitoring described in the thesis. Several observations from the process variables were presented as a guidance for developing data-driven process monitoring methods. The chapter then described how the PRONTO benchmark dataset can be useful for the technical works in the thesis.

Chapter 5

Tuning of RBF kernels and monitoring statistics in KPCA

The chapter investigates the tuning of Radial Basis Function (RBF) kernels and the behaviour of monitoring statistics when using Kernel Principal Component Analysis (KPCA) for process monitoring. By doing so, the first open question and the third open question identified in Section 3.5.1 are addressed. The first open question is about the proper tuning of kernels in kernel methods. The third open question is about the behaviour of monitoring statistics achieved by kernel methods. In this chapter, asymptotic analysis is applied to the kernel width of RBF kernels in order to consider the behaviour of RBF-KPCA when the kernel width is large or small. The chapter also considers the monotonicity of the monitoring statistics relative to fault severity achieved by RBF-KPCA with various kernel widths. Such analysis will demonstrate the influence of the kernel width of RBF kernels on the output of the monitoring methods. Based on these findings, in this chapters a tuning strategy for kernel widths will be proposed. several simulated examples are used to illustrate the findings. The findings will also be demonstrated using the PRONTO benchmark dataset.

5.1 Background and introduction

Previous chapters have reviewed the application of kernel-based methods to data-driven process monitoring of nonlinear and multimodal processes. Many researchers have adopted

Kernel Principal Component Analysis (KPCA) for data-driven fault detection (Hoffmann, 2007; Lee et al., 2004a; Ge et al., 2009; Li and Yang, 2015; Jiang and Yan, 2018). The thesis focuses on the KPCA method with various kernels and uses the abbreviation "RBF-KPCA" to refer to KPCA method with RBF kernels. Pilario et al. (2020), which is a recent review paper of kernel-based process monitoring algorithms, identified that most kernel methods use the RBF kernel. However, open questions still exist when applying kernel-based methods with RBF kernels, as discussed below.

The tuning of the kernel widths (denoted as δ in Eqn (3.13)) determines the accuracy of the monitoring models built by kernel-based methods. However, according to the previous works reviewed in Chapter 3, a systematic way of tuning the kernel widths for RBF kernels is still an open question. As for the monitoring statistics, many works, including Lee et al. (2004a), Alcala and Qin (2010), Ge et al. (2009), Deng et al. (2018) and Pilario et al. (2019), used the standard Hotelling's T^2 as a monitoring statistic. Chakour et al. (2018), Deng et al. (2018), and Pilario et al. (2019) defined the Squared Prediction Error (SPE) for KPCA in various different ways. Recently, Pilario et al. (2019) also demonstrated that the value of the RBF kernel will approach zero when the test sample moves sufficiently far away from the training data, leading to the monitoring statistic T^2 approaching a constant value. Indicators based on a combination of T^2 and SPE (Choi and Lee, 2004; Alcala and Qin, 2010), as well as other statistics (Ge et al., 2009; Zhang et al., 2018), have also been proposed to improve the fault detection performance.

The underlying problem motivating these developments is that in RBF-KPCA, as will be shown in this chapter, the value of T^2 does not increase monotonically with respect to the severity of the fault. It is necessary to consider if the desirable behaviour of a monitoring statistic in Section 2.6.1 can be satisfied. This chapter gives new insights into the tuning of RBF-KPCA when applied to fault detection. It investigates the influence of the kernel width from a theoretical perspective and shows that RBF-KPCA leads to increased false alarms when the kernel width is exceeding small, while exceedingly large kernel widths will lead to incorrect models and missed alarms.

The chapter also proves that the Hotelling's T^2 and SPE (squared prediction error) monitoring statistics that are widely used in linear principal component analysis do not have the same interpretation or behaviour when used with RBF-KPCA. The SPE for PCA is sensitive only to faults that cause the correlation structure in the data to break down, whereas the SPE for RBF-KPCA is sensitive to both types faults, when the correlation breaks done or

the process variables exceed their operating ranges. Moreover, the chapter demonstrates that in RBF-KPCA the T^2 statistic is not monotonic with respect to the severity of the fault. Therefore it should not be used because it can lead to missed detection of faults. This observation explains difficulties that other researchers have had in applying T^2 for RBF-KPCA. Based on these findings, this chapter proposes a novel strategy for tuning RBF-KPCA and for setting the thresholds for the SPE monitoring statistic for fault detection. The findings and the strategy are verified using the PRONTO dataset. A paper Tan et al. (2020b) has been submitted for publication and this paper is based on the work in this chapter.

5.2 PCA, KPCA, and RBF kernels

This section first introduces the PCA formulation and how PCA is used for fault detection. Next, the KPCA method with RBF kernels, as well as the associated monitoring statistics, are explained, extending the brief description given in Section 3.4.2. An illustrative example will show that the monitoring statistics, T^2 and SPE, for RBF-KPCA do not perform in the same way as the statistics for linear methods.

5.2.1 PCA for fault detection

The layout of the PCA method for process monitoring is reviewed here. The normalized training dataset $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n\} \in \mathbb{R}^{m \times n}$ includes n data samples of m variables. The dataset is normalized by removing the mean from the original data and dividing the original data by the standard deviations. To distinguish between the variables and the data samples, the thesis uses $\boldsymbol{x}_i \in \mathbb{R}^{m \times 1}$ for i-th data sample of the vector of variables $\boldsymbol{x}$. In the vector of variables $\boldsymbol{x} = [x_1, \ldots, x_n]$, x_j is the j-th process variable in the vector $\boldsymbol{x}$. $x_{i,j}$ denotes the value of the j-th variable in the i-th sample.

Section 3.3.1 briefly reviewed the mathematical formulation of PCA and other linear multivariate methods for feature extraction. PCA extracts features from X by applying eigenvalue decomposition to the sample covariance matrix:

$$\frac{1}{n-1} X^\top X = P_{\text{PCA}}^\top \, \Lambda_{\text{PCA}} \, P_{\text{PCA}} \tag{5.1}$$

where $\Lambda_{\text{PCA}} = \text{diag}\{\lambda_{1,\text{PCA}}, \lambda_{2,\text{PCA}}, \ldots, \lambda_{m,\text{PCA}}\} \in \mathbb{R}^{m\times m}$ is the diagonal matrix that contains non-zero decreasing eigenvalues ($\lambda_{1,\text{PCA}} \geq \lambda_{2,\text{PCA}} \geq \cdots \geq \lambda_{m,\text{PCA}} \leq 0$). $P_{\text{PCA}} \in \mathbb{R}^{m\times m}$ is the projection matrix from the original variable space to the feature space. Often the first r features with the largest eigenvalues are retained as PCs because these features can explain the variability in the original data. This is done by using only the first r columns of P_{PCA} to construct the reduced projection matrix $P_r \in \mathbb{R}^{r\times m}$. The PCs $\boldsymbol{z}_{\text{PCA}} \in \mathbb{R}^{r\times 1}$ are then obtained by projecting the original variable vector $\boldsymbol{x}$ using P_r:

$$\boldsymbol{z}_{\text{PCA}} = P_r \boldsymbol{x} \tag{5.2}$$

This is the PCA model that extracts PCs $\boldsymbol{z}_{\text{PCA}}$ from the original variable vector $\boldsymbol{x}$.

Two monitoring statistics, T^2 and SPE can be defined using the PCA model. The monitoring statistic T^2 is calculated using $\boldsymbol{z}_{\text{PCA}}$:

$$T^2 = \boldsymbol{z}_{\text{PCA}}^\top \Lambda_r^{-1} \boldsymbol{z}_{\text{PCA}} \tag{5.3}$$

where $\Lambda_r = \text{diag}\{\lambda_{1,\text{PCA}}, \lambda_{2,\text{PCA}}, \ldots, \lambda_{r,\text{PCA}}\} \in \mathbb{R}^{r\times r}$ is the reduced diagonal matrix with r eigenvalues.

It is also possible to reconstruct the original variable vector $\boldsymbol{x}$ using the projection matrix. The original projection matrix P_{PCA} obtained by PCA is orthogonal such $P_{\text{PCA}}^\top P_{\text{PCA}} = I$ where I is identity matrix. This gives $P_{\text{PCA}}^\top P_{\text{PCA}} \boldsymbol{x} = \boldsymbol{x}$. However, the retained number of PCs and the reduced projection matrix P_r will result in a mismatch between the original variable $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$, the reconstructed variables using P_r:

$$\hat{\boldsymbol{x}} = P_r^\top P_r \boldsymbol{x}. \tag{5.4}$$

The SPE is therefore defined as the reconstruction error of $\boldsymbol{x}$ using the PCA model:

$$\text{SPE} = \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2 = \|\boldsymbol{x} - P_r^\top P_r \boldsymbol{x}\|_2 \tag{5.5}$$

A fault is detected if at least one of these two criteria is satisfied:

$$T^2 > T^2_{\text{UCL}} \quad \text{or} \quad \text{SPE} > \text{SPE}_{\text{UCL}} \tag{5.6}$$

where T^2_{UCL} and SPE_{UCL} are the control limits for T^2 and SPE, respectively.

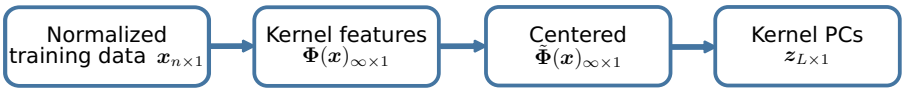


Figure 5.1: Flowchart for kernel principal component analysis

5.2.2 RBF-KPCA for fault detection

RBF-KPCA for feature extraction

Fig. 5.1 presents the procedure of feature extraction using RBF-KPCA. As formulated in Schölkopf et al. (1998), KPCA first projects the original variables to a new feature space; this operation is the kernel transformation. Then dimension reduction is conducted in the new feature space to obtain the kernel Principal Components (PCs). The kernel transformation builds models to describe the training data without specifying a model structure. Any new sample will be compared against the training data and the model output is calculated accordingly.

Assuming the normalized training dataset is $X \in \mathbb{R}^{m\times n}$, X is first projected to another feature space Φ under the KPCA framework. The feature space Φ has infinite dimensions when the RBF kernel is applied. This means that a vector of measurements $\boldsymbol{x}$ undergoes a mapping to the feature space $\Phi(\boldsymbol{x})$, $\boldsymbol{x} \mapsto \Phi(\boldsymbol{x})$, where $\boldsymbol{x} = [x_1, x_2, \ldots, x_m]$ and $\Phi(\boldsymbol{x}) = [\phi_1(\boldsymbol{x}), \phi_2(\boldsymbol{x}), \ldots, \phi_\infty(\boldsymbol{x})]$. The projected variables in the new feature space $\Phi(\boldsymbol{x})$ cannot be calculated directly. Nevertheless, the covariance matrix of $\Phi(\boldsymbol{x})$, $K \in \mathbb{R}^{n\times n}$, can be obtained using the kernel function. The RBF kernel function defines the entries of the kernel matrix K:

$$\begin{aligned} K_{i,j} &= \Phi(\boldsymbol{x}_i)\cdot\Phi(\boldsymbol{x}_j) = \phi_1(\boldsymbol{x}_i)\phi_1(\boldsymbol{x}_j) + \cdots + \phi_\infty(\boldsymbol{x}_i)\phi_\infty(\boldsymbol{x}_j) \\ &= \exp\left(-\frac{1}{\delta^2}(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top(\boldsymbol{x}_i - \boldsymbol{x}_j)\right) \end{aligned} \tag{5.7}$$

where $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are two data samples. The kernel function $K_{i,j}$ is the dot product of $\Phi(\boldsymbol{x}_i)$ and $\Phi(\boldsymbol{x}_j)$ and δ is the kernel width of RBF kernels. As Eqn (5.7) shows, the kernel width δ regulates how the entry of the kernel matrix reduces as the distance between two samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ increases. This will further influence how closely a sample should be located to the training sample in order to be recognized as a healthy sample.

It is not guaranteed that $\Phi(X)$, the projections of X to the kernel feature space, are centered. Therefore the K matrix is centered such that $\Phi(X)$ is also centered:

$$\begin{aligned}\tilde{K} &= (\Phi(X) - \bar{\Phi})^\top \cdot (\Phi(X) - \bar{\Phi}) = \tilde{\Phi}(X)^\top \cdot \tilde{\Phi}(X) \\ &= K - \frac{1}{n}\mathbf{1}_{n\times n}K - \frac{1}{n}K\mathbf{1}_{n\times n} + \frac{1}{n^2}\mathbf{1}_{n\times n}K\mathbf{1}_{n\times n}\end{aligned} \tag{5.8}$$

where $\bar{\Phi} = 1/n\sum_{i=1}^{n}\Phi(\boldsymbol{x}_i)$ is the center of $\Phi(X)$. $\tilde{\Phi}(X)$ is the centered result of $\Phi(X)$. $\mathbf{1}_{n\times n}$ is an $n \times n$ matrix with all entries having value 1. In the second step, PCA is implemented in the Φ space by applying eigenvalue decomposition to the centered kernel matrix $\tilde{K}$:

$$\tilde{K} = \boldsymbol{\alpha}^\top \Lambda^{-1} \boldsymbol{\alpha} \tag{5.9}$$

where $\boldsymbol{\alpha}^{(l)} \in \mathbb{R}^{n\times 1}$ is the l-th eigenvector and $\boldsymbol{\alpha} = \{\boldsymbol{\alpha}^{(1)}, \cdots, \boldsymbol{\alpha}^{(n)}\}$. $\Lambda = \text{diag}\{\lambda_1, \cdots, \lambda_n\}$ where λ_l is the l-th eigenvalue. Assuming the first L kernel PCs are retained, the value of the l-th kernel PC for $\boldsymbol{x}_j$ is $z_j^{(l)}$, given by:

$$z_j^{(l)} = \boldsymbol{V}^{(l)} \cdot \tilde{\Phi}(\boldsymbol{x}_j) = \sum_{i=1}^{n} \tilde{\alpha}_i^{(l)} \tilde{K}_{i,j} \tag{5.10}$$

where $\tilde{\alpha}_i^{(l)}$ is a normalized version of $\alpha_i^{(l)}$ in (5.9) such that $\|\tilde{\boldsymbol{\alpha}}^{(l)}\|_2 = 1/\lambda_l$ for $l = 1, \ldots, L$ and $\boldsymbol{V}^{(l)}$ is the l-th row of the projection matrix $\boldsymbol{V}$. Both the kernel features $\Phi(X) \in \mathbb{R}^{\infty\times 1}$ and the projection matrix $\boldsymbol{V} \in \mathbb{R}^{n\times\infty}$ have infinite dimensions. Although $\Phi(X)$ and $\boldsymbol{V}$ cannot be calculated in RBF-KPCA, the kernel PCs, which are the principal components of $\Phi(\boldsymbol{x})$, can be calculated explicitly using Eqn (5.10) with $\tilde{\boldsymbol{\alpha}}$ and $\tilde{K}$.

Monitoring statistics by RBF-KPCA

The monitoring statistic is usually defined as a function of the retained kernel PCs. For example, the T^2 statistic of the j-th sample $\boldsymbol{x}_j$ is:

$$T_j^2 = \boldsymbol{z}_j^\top \Lambda_L^{-1} \boldsymbol{z}_j = \sum_{l=1}^{L} \lambda_l^{-1} \left(\sum_{i=1}^{n} \tilde{\alpha}_i^{(l)} \tilde{K}_{i,j} \right)^2. \tag{5.11}$$

where Λ_L^{-1} is the reduced diagonal matrix such that $\Lambda_L^{-1} = \text{diag}\{\lambda_1, \ldots, \lambda_L\}$.

For a test sample $\boldsymbol{x}_{\text{test}}$, T^2 is calculated using the corresponding kernel PCs $\boldsymbol{z}_{\text{test}}$. The value of $\boldsymbol{z}_{\text{test}}$ is small when $\boldsymbol{x}_{\text{test}}$ is close to the center of healthy data because in Eqn (5.8), the

projected $\Phi(\boldsymbol{x}_{\text{test}})$ will be close to $\bar{\Phi}$ and the $\tilde{K}$ will be close to zero. A sample located far away from the healthy data is faulty. The value of $\boldsymbol{z}_{\text{test}}$ is also small when $\boldsymbol{x}_{\text{test}}$ is faulty because the projection of $\boldsymbol{x}_{\text{test}}$ to the retained L kernel PCs will be small. Since T^2 will be small for both scenarios, it cannot be an adequate monitoring statistic for RBF-KPCA.

In RBF-KPCA, the SPE of $\boldsymbol{x}_j$ is defined as the second order norm of the difference between $\tilde{\Phi}(\boldsymbol{x}_j)$, the centered projection of a normalized sample $\boldsymbol{x}_j$ in the kernel feature space, and $\hat{\tilde{\Phi}}(\tilde{\boldsymbol{x}}_j)$:

$$\begin{aligned}
\text{SPE}_j &= \|\tilde{\Phi}(\boldsymbol{x}_j) - \hat{\tilde{\Phi}}(\tilde{\boldsymbol{x}}_j)\|_2 \\
&= \|\tilde{\Phi}(\boldsymbol{x}_j) - \boldsymbol{V}^\top \cdot \boldsymbol{V} \cdot \tilde{\Phi}(\boldsymbol{x}_j)\|_2 \\
&= \|\tilde{\Phi}(\boldsymbol{x}_j)\|_2 - 2(\tilde{\Phi}^\top(\boldsymbol{x}_j) \cdot \boldsymbol{V}^\top) \cdot (\boldsymbol{V} \cdot \tilde{\Phi}(\boldsymbol{x}_j)) + (\boldsymbol{V}^\top \cdot \boldsymbol{V} \cdot \tilde{\Phi}(\tilde{\boldsymbol{x}}_j))^\top \cdot \boldsymbol{V}^\top \cdot \boldsymbol{V} \cdot \tilde{\Phi}(\tilde{\boldsymbol{x}}_j) \\
&= \|\tilde{\Phi}(\boldsymbol{x}_j)\|_2 - (\boldsymbol{V} \cdot \tilde{\Phi}(\boldsymbol{x}_j))^\top \cdot (\boldsymbol{V} \cdot \tilde{\Phi}(\boldsymbol{x}_j))
\end{aligned} \tag{5.12}$$

where, as in Eqn (5.10), $\boldsymbol{V}$ is the projection matrix to the kernel PC space such that $\boldsymbol{V} \cdot \boldsymbol{V}^\top = I$. $\Phi_0 = 1/n \sum_{i=1}^{n} \Phi(\boldsymbol{x}_i)$ is the center of the projections of the training samples in the kernel feature space. Hoffmann (2007) referred to this second norm as the reconstruction error of $\tilde{\Phi}(\boldsymbol{x}_j)$ using kernel PCs obtained by KPCA. Although $\tilde{\Phi}(\boldsymbol{x})$ cannot be obtained directly, its second order norm is:

$$\begin{aligned}
\|\tilde{\Phi}(\boldsymbol{x}_j)\|_2 &= \Phi^\top(\boldsymbol{x}_j) \cdot \Phi(\tilde{\boldsymbol{x}}_j) - 2\Phi^\top(\boldsymbol{x}_j) \cdot \Phi_0 + \Phi_0^\top \Phi_0 \\
&= k(\boldsymbol{x}_j, \boldsymbol{x}_j) - \frac{2}{n} \sum_{i=1}^{n} k(\boldsymbol{x}_i, \boldsymbol{x}_j) + \frac{1}{n^2} \sum_{i,j=1}^{n} k(\boldsymbol{x}_i, \boldsymbol{x}_j) \\
&= 1 - 2\bar{K}_j + \bar{K}.
\end{aligned} \tag{5.13}$$

where $\bar{K}_j$ is the mean of the j-th row in K and $\bar{K}$ is the mean of all entries in K. The second term of Eqn (5.12) is the second order norm of the kernel PCs $\boldsymbol{z}_j$ of $\boldsymbol{x}_j$:

$$(\boldsymbol{V} \cdot \tilde{\Phi}(\boldsymbol{x}_j))^\top \cdot (\boldsymbol{V} \cdot \tilde{\Phi}(\boldsymbol{x}_j)) = \boldsymbol{z}_j^\top \boldsymbol{z}_j. \tag{5.14}$$

Therefore, SPE for KPCA can be written explicitly:

$$\text{SPE}_j = 1 - 2\bar{K}_j + \bar{K} - \boldsymbol{z}_j^\top \boldsymbol{z}_j \tag{5.15}$$

For a test sample $\boldsymbol{x}_{\text{test}}$, the SPE is calculated by using the mean of the kernel vector

$$\frac{1}{n}\sum_{i=1}^{n} k(\boldsymbol{x}_i, \boldsymbol{x}_{\text{test}})$$

and $\boldsymbol{z}_{\text{test}}$ as $\bar{K}_j$ and $\boldsymbol{z}_j$ in Eqn (5.15), respectively.

5.2.3 Illustrative examples

The behaviour of monitoring statistics

The following illustrative example compares the performance of T^2 and SPE obtained by PCA and RBF-KPCA for various types of faults. The example demonstrates that the roles of T^2 and SPE, respectively, in RBF-KPCA are not the same as their roles in PCA. The data sets used for training and testing are plotted in Fig. 5.2. The training data are generated from a linear algebraic model with white Gaussian disturbances. Three test samples represent three faulty cases. Test 1 represents the case where the linear relationship between the variables still holds, but the values exceed the healthy range. Test 2 is the case where the measurements of variable 1 and variable 2 each fall within the same range of values as the healthy case, but the relationship between the variables is not the same as in the healthy case. Test 3 combines both cases where the variables exceed the range and follow a different relationship. PCA and RBF-KPCA are applied to these data. In this example, PCA obtains two PCs. The first PC with the largest variance is retained to calculate T^2 and the second PC is used for SPE. For this example, kernel PCs obtained by KPCA are retained such that the percentage of the accumulated variance explained by the kernel PCs is over 99%.

Table 5.1 compares the T^2 and the SPE for PCA and KPCA, respectively. The upper control limit for each monitoring statistic is defined as the 95% percentile of the monitoring statistic values obtained in the training set. An anomalous sample is detected by a monitoring statistic if the value of this statistic obtained for the sample exceeds the control limit. The results demonstrate that:

1. In PCA, T^2 detects the case where the variables exceed the healthy operating range. SPE detects the case where the sample does not follow the model of the training data. PCA needs both T^2 and SPE to detect all the three faulty samples;

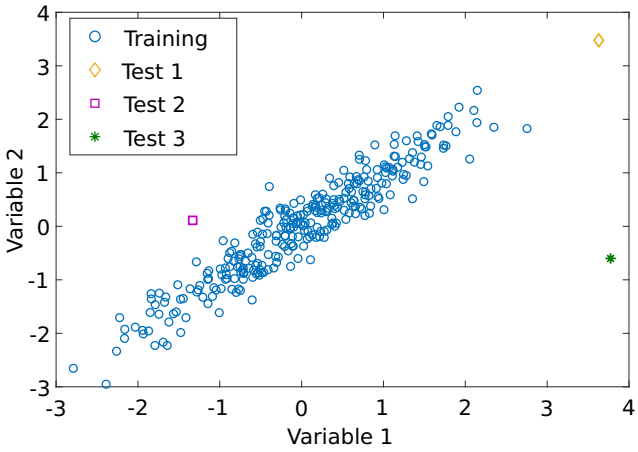


Figure 5.2: Scatter plot of data in the illustrative example

Table 5.1: Monitoring statistics of illustrative example in Fig. 5.2

	PCA		RBF-KPCA	
	T^2	SPE	T^2	SPE
Upper control limit	3.9116	0.1895	0.3228	0.3561
Test 1	12.9739	0.0119	0.0827	1.0455
Test 2	0.3822	1.0397	0.1397	0.9708
Test 3	2.5886	9.5703	0.0310	1.0931

2. T^2 for RBF-KPCA cannot detect any fault when using the upper control limit because the value of T^2 in Eqn (5.15) approaches zero when the fault is large. It is not appropriate to use the T^2 and the upper control limit for fault detection;

3. In contrast, SPE for RBF-KPCA can detect all three types of faults.

This example has demonstrated that the T^2 and the SPE for KPCA behave differently when compared with the T^2 and the SPE for PCA. These findings are explored mathematically and explained in Section 5.4.

The influence of tuning

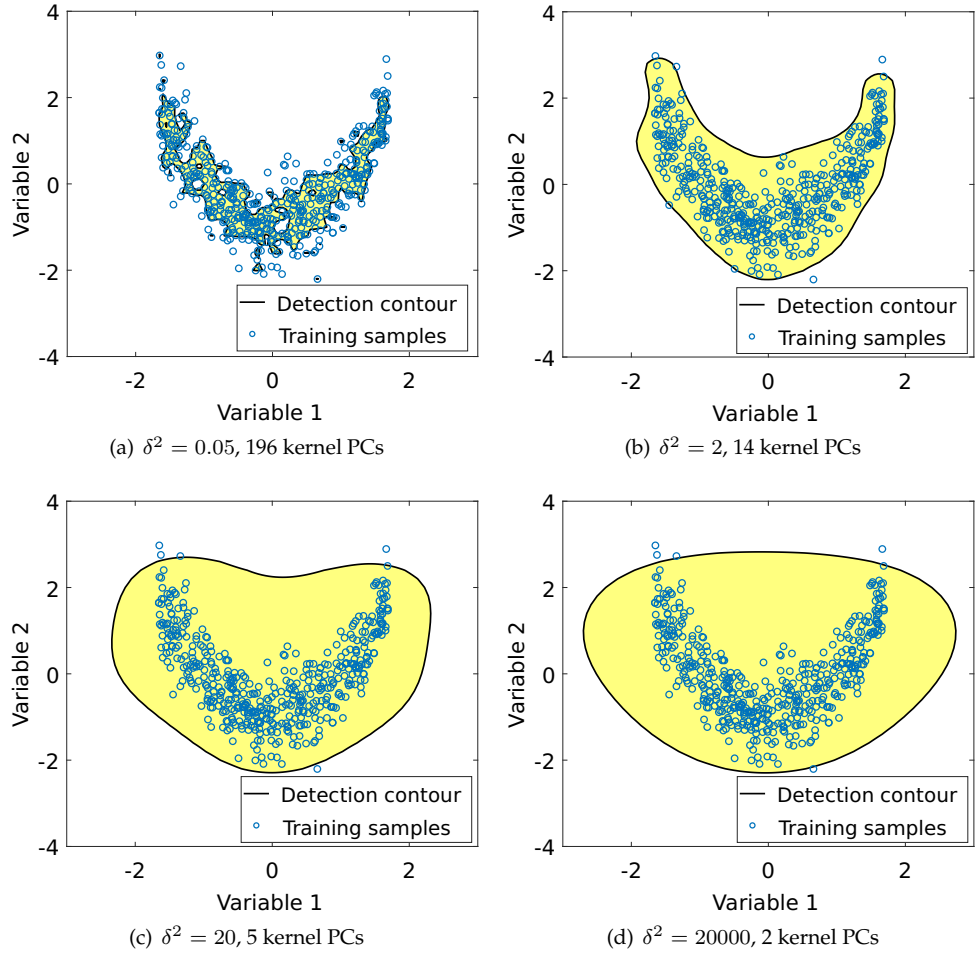
In the RBF kernel function shown in Eqn (5.7), the kernel width δ regulates the behaviour of the kernel function. Such behaviour will further influence the KPCA-based modelling and fault detection. The following nonlinear algebraic example is used to demonstrate the

influence of δ:

$$\begin{aligned} x_1 &\sim U[-1.5, 1.5], \\ x_2 &= x_1^2 - 1 + e, \\ \text{where} \quad e &\sim N(0, 0.3) \end{aligned} \tag{5.16}$$

The first variable x_1 follows a uniform distribution denoted by U. The relationship between x_1 and x_2 is quadratic, resulting in a nonlinear dataset. The second variable x_2 is contaminated by a Gaussian noise e with zero-mean and a variance of 0.3. N denotes the Gaussian distribution. The randomness in x_1 and x_2 represents the randomness in real-life data. The uniform distribution of x_1 simulates process variables with certain operating ranges (as indicated by Eqn (2.4a)). The Gaussian noise in x_2 simulates measurement noise, which usually has a smaller variance than the magnitude of the measured value. The training dataset contains 500 samples randomly generated using Eqn (5.16). Various δ values are used to train the KPCA model on the dataset. The SPE of these samples is then calculated for fault detection. The upper control limit of SPE is used such that a sample with the SPE value exceeding the control limit is detected as a fault. For this two-dimensional problem, it is possible to visualize the detection contours obtained by selecting the control limit of the SPE as the 1% percentile of the SPE values obtained on the training data and connecting the points at which the SPE reaches its control limit for each KPCA model. The shaded area in each figure shows the healthy range and samples in the white area are faulty. Fig. 5.3(a) shows that a small δ value yields an over-fitted model. Larger δ values will result in relaxed detection contours. However, when δ is too large, Fig. 5.3(d) shows that the detection contour loses its ability to capture the nonlinear profile of the data.

To summarize, the tuning of δ influences the performance of RBF-KPCA and the SPE for fault detection. The following sections will investigate the influence of δ and the behaviour of monitoring statistics through asymptotic analysis. The sections will also analyse and explain why SPE (Eqn (5.15)) and not T^2 (Eqn (5.11)) should be used as the monitoring statistic for RBF-KPCA applications.



(a) $\delta^2 = 0.05$, 196 kernel PCs (b) $\delta^2 = 2$, 14 kernel PCs

(c) $\delta^2 = 20$, 5 kernel PCs (d) $\delta^2 = 20000$, 2 kernel PCs

Figure 5.3: Detection contours by SPE and RBF-KPCA with various kernel widths. Yellow-shaded areas: range of values classified as healthy.

5.3 Asymptotic behaviour of RBF kernels

This section investigates the asymptotic behaviour of the RBF kernels, both when the δ value tends to an exceedingly large value and conversely when it tends to an exceedingly small value.

5.3.1 Exceedingly large kernel width ($\delta \to \infty$)

This section will show that when δ is large, the centered RBF kernel matrix is a scaled version of the centered kernel matrix obtained by a linear kernel.

Training the monitoring model

The i, j-th entry of the kernel matrix K is:

$$\begin{aligned} K_{i,j} &= \lim_{\delta^2 \to \infty} \exp\left(-\frac{(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)}{\delta^2}\right) \\ &= \lim_{\delta^2 \to \infty} \left[1 - \frac{(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)}{\delta^2} + o\left(\frac{(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)}{\delta^2}\right)\right] \\ &\approx 1 - \frac{1}{\delta^2}\left(\boldsymbol{x}_i^\top \boldsymbol{x}_i + \boldsymbol{x}_j^\top \boldsymbol{x}_j - 2\boldsymbol{x}_i^\top \boldsymbol{x}_j\right) \end{aligned} \tag{5.17}$$

δ is considered exceedingly large when $o\left((\boldsymbol{x}_i - \boldsymbol{x}_j)^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)/\delta^2\right) \approx 0$ for all $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in the training dataset. The kernel matrix K will be centred using Eqn (5.8). The i, j-th entry of $\tilde{K}$ is therefore:

$$\tilde{K}_{i,j} = K_{i,j} - \bar{K}_{i,\text{row}} - \bar{K}_{\text{col},j} + \bar{K} \tag{5.18}$$

In Eqn (5.17), $\bar{K}_{i,\text{row}}$ and $\bar{K}_{\text{col},j}$ are the means of i-th row and j-th column of K, respectively.

$$\begin{aligned} \bar{K}_{i,\text{row}} &= \frac{1}{n}\sum_{j=1}^{n} K_{i,j} \\ &= \frac{1}{n}\sum_{j=1}^{n}\left[1 - \frac{1}{\delta^2}\left(\boldsymbol{x}_i^\top \boldsymbol{x}_i + \boldsymbol{x}_j^\top \boldsymbol{x}_j - 2\boldsymbol{x}_i^\top \boldsymbol{x}_j\right)\right] \\ &= 1 - \frac{1}{\delta^2}\boldsymbol{x}_i^\top \boldsymbol{x}_i - \frac{1}{n\delta^2}\sum_{j=1}^{n} \boldsymbol{x}_j^\top \boldsymbol{x}_j + \frac{2}{n\delta^2}\sum_{j=1}^{n} \boldsymbol{x}_i^\top \boldsymbol{x}_j \end{aligned} \tag{5.19}$$

$$\bar{K}_{\text{col},j} = 1 - \frac{1}{\delta^2}\boldsymbol{x}_j^\top \boldsymbol{x}_j - \frac{1}{n\delta^2}\sum_{i=1}^{n} \boldsymbol{x}_i^\top \boldsymbol{x}_i + \frac{2}{n\delta^2}\sum_{i=1}^{n} \boldsymbol{x}_i^\top \boldsymbol{x}_j \tag{5.20}$$

$\bar{K}$ is the mean of all entries of K:

$$\bar{K} = 1 - \frac{2}{n\delta^2}\sum_{i=1}^{n} \boldsymbol{x}_i^\top \boldsymbol{x}_i + \frac{2}{n^2\delta^2}\sum_{i=1}^{n}\sum_{j=1}^{n} \boldsymbol{x}_i^\top \boldsymbol{x}_j. \tag{5.21}$$

Hence the centered kernel matrix $\tilde{K}$ has the following entry:

$$\tilde{K}_{i,j} = \frac{2}{\delta^2}\left[\boldsymbol{x}_i^\top \boldsymbol{x}_j - \frac{1}{n}\left(\sum_{j=1}^{n} \boldsymbol{x}_i^\top \boldsymbol{x}_j + \sum_{i=1}^{n} \boldsymbol{x}_i^\top \boldsymbol{x}_j\right) + \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} \boldsymbol{x}_i^\top \boldsymbol{x}_j\right]. \tag{5.22}$$

As shown in Schölkopf et al. (1997), KPCA with a linear kernel, defined as $K_{\text{lin}}(i, j) = \boldsymbol{x}_i^\top \boldsymbol{x}_j$, will reduce to the ordinary linear PCA. The centered linear kernel matrix is defined

as:

$$\tilde{K}_{\text{lin},i,j} = \boldsymbol{x}_i^\top \boldsymbol{x}_j - \frac{1}{n}\left(\sum_{i=1}^{n} \boldsymbol{x}_i^\top \boldsymbol{x}_j + \sum_{j=1}^{n} \boldsymbol{x}_i^\top \boldsymbol{x}_j\right) + \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} \boldsymbol{x}_i^\top \boldsymbol{x}_j. \tag{5.23}$$

A comparison of Eqns (5.22) and (5.23) shows that $\tilde{K}_{i,j} = 2\delta^{-2}\tilde{K}_{\text{lin},i,j}$. Thus, when δ is large, the RBF kernel will generate a centered kernel matrix whose entries are proportional to the entries of the centered kernel matrix obtained by the linear kernel. Hence the eigenvectors and eigenvalues of the kernel matrix obtained by the RBF kernel will be proportional to those of the linear kernel matrix. Moreover, when δ is exceedingly large, the number of kernel PCs of the illustrative example in Fig. 5.3 has already reduced to two, which is the same as the linear PCA result. This explains the behaviour in Fig. 5.3(d).

For a test sample $\boldsymbol{x}_{\text{test}}$

The test kernel vector $K_{i,\text{test}}$ of $\boldsymbol{x}_{\text{test}}$ is:

$$\begin{aligned} K_{i,\text{test}} &= \lim_{\delta^2\to\infty} \exp\left(-\frac{(\boldsymbol{x}_i - \boldsymbol{x}_{\text{test}})^\top(\boldsymbol{x}_i - \boldsymbol{x}_{\text{test}})}{\delta^2}\right) \\ &\approx 1 - \frac{1}{\delta^2}\left(\boldsymbol{x}_i^\top \boldsymbol{x}_i + \boldsymbol{x}_{\text{test}}^\top \boldsymbol{x}_{\text{test}} - 2\boldsymbol{x}_i^\top \boldsymbol{x}_{\text{test}}\right). \end{aligned} \tag{5.24}$$

The centered value $\tilde{K}_{i,\text{test}}$ is:

$$\tilde{K}_{i,\text{test}} = \frac{2}{\delta^2}\left(\boldsymbol{x}_i^\top \boldsymbol{x}_{\text{test}} - \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i^\top x_{\text{test}} - \frac{1}{n}\sum_{j=1}^{n} \boldsymbol{x}_i^\top \boldsymbol{x}_j^\top + \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} \boldsymbol{x}_i^\top \boldsymbol{x}_j\right). \tag{5.25}$$

For the linear kernel,

$$K_{\text{lin},i,\text{test}} = \boldsymbol{x}_i^\top \boldsymbol{x}_{\text{test}}. \tag{5.26}$$

The centered kernel vector $\tilde{K}_{\text{lin},i,\text{test}}$ is:

$$\tilde{K}_{\text{lin},i,\text{test}} = \boldsymbol{x}_i^\top \boldsymbol{x}_{\text{test}} - \frac{1}{n}\left(\sum_{i=1}^{n} \boldsymbol{x}_i^\top \boldsymbol{x}_{\text{test}} + \sum_{j=1}^{n} \boldsymbol{x}_i^\top \boldsymbol{x}_j\right) + \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} \boldsymbol{x}_i^\top \boldsymbol{x}_j. \tag{5.27}$$

Therefore, $\tilde{K}_{i,\text{test}} = 2\delta^{-2}\tilde{K}_{\text{lin},i,\text{test}}$. To conclude, the RBF kernel will result in centered kernel matrices for training data and centered kernel vectors for test data that are proportional to the equivalent centered kernel matrices and centered kernel vectors for a linear kernel when δ is exceedingly large relative to the training dataset X_{train} and the test sample $\boldsymbol{x}_{\text{test}}$. The behaviour of RBF-KPCA when $\boldsymbol{x}_{\text{test}}$ also approaches infinity will be investigated later.

5.3.2 Exceedingly small kernel width ($\delta \to 0$)

Training the monitoring model

On the other hand, the value of δ is considered exceedingly small when the kernel function $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ will reduce to the Kronecker Delta function:

$$\begin{aligned} K_{i,j} &= \lim_{\delta^2 \to 0} \exp\left(-\frac{(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)}{\delta^2}\right) \\ &= \begin{cases} 1, & i = j \\ 0, & i \neq j. \end{cases} \end{aligned} \tag{5.28}$$

This results in the kernel matrix K being an $n \times n$ identity matrix $I_{n\times n}$. The centered kernel matrix $\tilde{K}$ then becomes:

$$\tilde{K} = \begin{bmatrix} 1-1/n & -1/n & -1/n & \dots & -1/n \\ -1/n & 1-1/n & -1/n & \dots & -1/n \\ \dots & & & & \\ -1/n & -1/n & -1/n & \dots & 1-1/n \end{bmatrix}. \tag{5.29}$$

which has $n-1$ eigenvalues $\lambda_1 = \lambda_2 = \cdots = \lambda_{n-1} = 1$ and one eigenvalue $\lambda_n = 0$. The first $n-1$ normalized eigenvectors satisfy the following condition:

$$\sum_{i=1}^{n} \tilde{\alpha}_i^{(l)} = 0 \quad \text{for } l = [1, 2, \cdots, n-1]. \tag{5.30}$$

For a new sample $\boldsymbol{x}_{\text{test}}$

According to Eqn (5.28), $K_{i,\text{test}} = 0$ if $\boldsymbol{x}_{\text{test}} \notin X_{\text{train}}$ for all i. The centered value $\tilde{K}_{i,\text{test}}$ is:

$$\tilde{K}_{i,\text{test}} = K_{i,\text{test}} - \bar{K}_{i,\text{test}} - \bar{K}_{i,\text{row}} + \bar{K} = 0 \tag{5.31}$$

where $\bar{K}_{i,\text{row}} = \bar{K} = 1/n$. Therefore, both the kernel vector $K_{i,\text{test}}$ and centered kernel vector $\tilde{K}_{i,\text{test}}$ are zero vectors. Fig. 5.3(a) is an example of an over-fitted model caused by

δ being set too small. In the extreme case of over-fitting, the detection contour will shrink into a Dirac measure of the training set X_{train} in the variable space. In other words, any test sample that is not identical to a sample in the training set will be detected as a fault.

5.4 Behaviour of monitoring statistics in RBF-KPCA

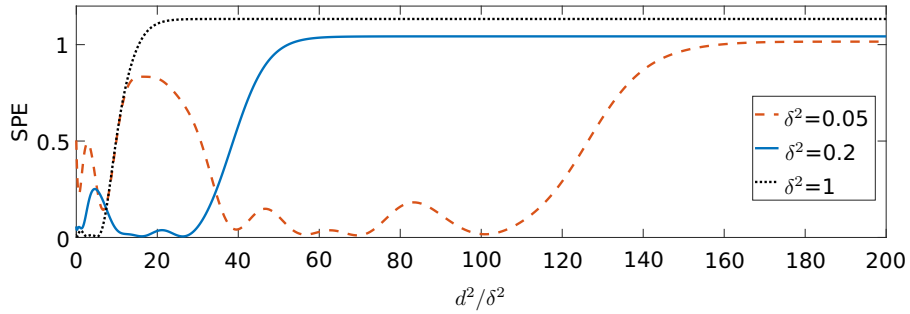
This section demonstrates why the SPE defined by Eqn (5.15) is a good choice for a general-purpose single monitoring statistic for RBF-KPCA. Other than for the over-fitted case when δ is too small, the SPE for RBF-KPCA increases monotonically with respect to the severity of faults. Therefore, an anomalous sample can be detected using the SPE for RBF-KPCA and its upper control limit. Nevertheless, T^2 has been widely used in the literature as a monitoring statistic in RBF-KPCA. Section 5.4.2 will analyse and explain the properties of T^2. Its non-monotonic behaviour explains the unsatisfactory detection performance of T^2 for KPCA. Moreover, some of the adjustments that previous authors have made to adapt T^2 as a monitoring statistic for RBF-KPCA, such as the need for both upper and lower control limits (Choi et al., 2005a), can also be explained.

5.4.1 Behaviour of SPE for RBF-KPCA

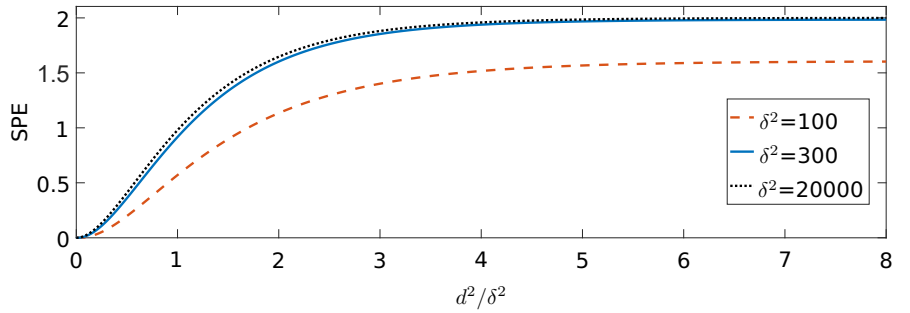
Eqn (5.12) defined the SPE as the difference between the kernel features $\tilde{\Phi}(\tilde{\boldsymbol{x}})$ and the reconstructed $\hat{\tilde{\Phi}}(\tilde{\boldsymbol{x}})$ after apply PCA in the Φ space. When using the RBF kernel, the following limit of SPE exists when $\boldsymbol{x}_{\text{test}} \to \infty$:

$$\begin{aligned} \text{SPE}_{\text{test,lim}} &= 1 - 2\bar{K}^{(\text{lim})}_{i,\text{test}} + \bar{K} - \boldsymbol{z}^{(\text{lim})\top}_{\text{test}} \boldsymbol{z}^{(\text{lim})}_{\text{test}} \\ &= 1 + \bar{K} - \boldsymbol{z}^{(\text{lim})\top}_{\text{test}} \boldsymbol{z}^{(\text{lim})}_{\text{test}} \\ \text{s.t.} \quad \boldsymbol{z}^{(\text{lim})\top}_{\text{test}} \boldsymbol{z}^{(\text{lim})}_{\text{test}} &= \sum_{l=1}^{L} \left(\sum_{i=1}^{n} \tilde{\alpha}_i^{(l)} \left[\bar{K} - \bar{K}_{i,\text{row}} \right] \right)^2 \end{aligned} \tag{5.32}$$

where $\bar{K}^{(\text{lim})}_{i,\text{test}} = 0$. Since SPE converges to a non-zero finite value when $\boldsymbol{x}_{\text{test}}$ approaches infinity, SPE cannot be χ^2 distributed because a χ^2-distributed random variable ranges from zero to infinity. Therefore, unlike the ordinary PCA, the control limits for T^2 and SPE should not be set according to the χ^2 distribution.



(a) Small δ values



(b) Large δ values

Figure 5.4: SPE with respect to d^2/δ^2. $\delta^2 = 0.05$: over-fitted model. $\delta^2 = 20000$: linear model

The illustrative example presented in Fig. 5.2 and Table 5.1 has demonstrated that the SPE in RBF-KPCA can detect both the violation of healthy ranges and the model-mismatch. The reason is that, when RBF kernel is used, the higher-dimensional kernel PCs are supposed to be a comprehensive description of the training data in the original variable space since these kernel PCs are obtained such that the reconstruction error is minimized for the training data. Therefore, the process model and the feasible range are learned simultaneously.

The SPE can distinguish between the case where the test sample is located close to the center of the training data and the case where the test sample is located far away from the training data. To demonstrate the behaviour, Fig. 5.4 shows the trends of SPE with respect to d^2/δ^2 given δ values, where $d = \sqrt{\|\boldsymbol{x}\|_2}$ is the Euclidean distance between a data sample and the origin in the variable space. The quantity d represents the distance between a data sample and the normalized training dataset. When $\delta = 0.05$, the RBF-KPCA model is over-fitted to the data in the training set, leading to the contour in Fig. 5.3(a). In this situation, as may be expected, the SPE has non-monotonic behaviour because any new data point that is in between the training samples is considered as faulty. For larger values of δ, the SPE increases monotonically as d increases, indicating the sample $\boldsymbol{x}$ deviates from the training

data. In particular, when δ^2 is small, e.g. $\delta = 0.2$, Fig. 5.4(a) shows that SPE already increases monotonically as d^2/δ^2 is around 30, which indicates a d^2 value of 6.

On the other hand, Fig. 5.4(b) is presented only to show that SPE does not suffer from the non-monotonic behaviour of T^2 when d^2/δ^2 is large. In reality, any smaller d^2 values will also result in a monotonically increasing SPE. It is necessary to notice that, although the SPE still increases monotonically when δ is extremely large, e.g. in Fig. 5.4(b), the δ values may lead to under-fitted models that cannot capture the data nonlinearity and, as a result, such δ values should be avoided.

The criterion for fault detection using SPE is:

$$\text{SPE}_{\text{test}} > \text{SPE}_{\text{UCL}} \tag{5.33}$$

where SPE_{UCL} is the upper control limit of SPE. This value may be set according to the training data. Section 5.5.2 will discuss the setting of the control limit for SPE.

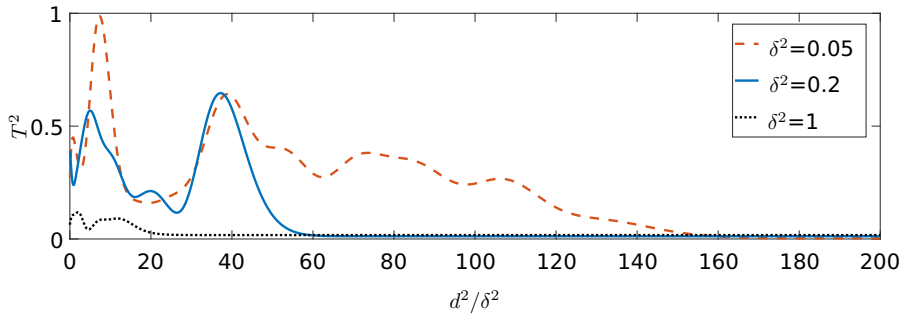
5.4.2 Behaviour of T^2 for RBF-KPCA

The T^2 statistic defined by Eqn (5.11) is suitable for PCA-based fault detection because it increases as a test data sample moves away from the training set and a fault is detected if T^2 exceeds its upper control limit. However, in RBF-KPCA T^2 may not be monotonic. This section investigates the behaviour of T^2 with respect to both δ and $\boldsymbol{x}_{\text{test}}$. Fig. 5.5 shows the trends of T^2 with respect to d^2/δ^2 given δ values for the illustrative example of Fig. 5.3. It is evident that the T^2 statistic does not increase monotonically as d increases for any choice of δ. Such behaviour can be explained through mathematical analysis.

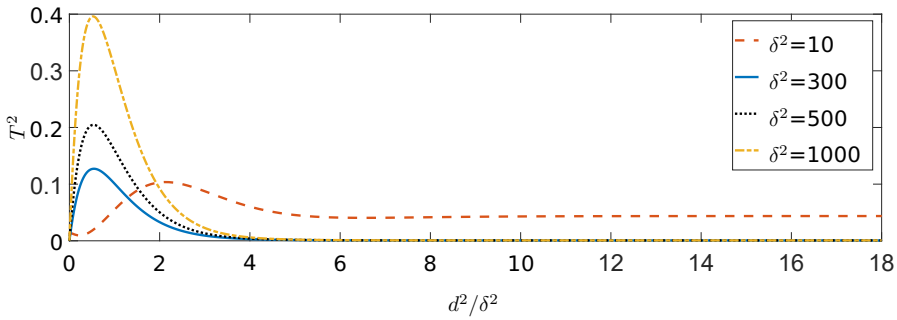
Upper bound

For an arbitrary test sample $\boldsymbol{x}_{\text{test}} \in \mathbb{R}^{r\times 1}$, the monitoring statistic T^2 is calculated as follows using the centered kernel vector $\tilde{K}_{\text{test}}$ and the eigenvectors obtained from the training data.

$$\begin{aligned} T^2_{\text{test}} &= \sum_{l=1}^{L} \lambda_l^{-1} \left(\sum_{i=1}^{n} \tilde{\alpha}_i^{(l)} \tilde{K}_{i,\text{test}} \right)^2 \\ &\leq \sum_{l=1}^{L} \lambda_l^{-1} \left[\sum_{i=1}^{n} \tilde{\alpha}_i^{(l)2} \tilde{K}^2_{i,\text{test}} + \sum_{i\neq j}^{n} \left| \tilde{\alpha}_i^{(l)} \tilde{\alpha}_j^{(l)} \right| \left| \tilde{K}_{i,\text{test}} \tilde{K}_{j,\text{test}} \right| \right] \end{aligned} \tag{5.34}$$



(a) Small δ values



(b) Large δ values

Figure 5.5: T^2 with respect to d^2/δ^2. $\delta^2 = 0.05$: over-fitted model. $\delta^2 = 10000$: linear model

where $\tilde{K}_{i,\text{test}}$ and $\tilde{K}_{j,\text{test}}$ are the i-th and j-th entry of the centered kernel vector $\tilde{K}_{\text{test}}$, respectively. $\tilde{\alpha}_i^{(l)}$ and $\tilde{\alpha}_i^{(l)}$ are i-th and j-th entry of the l-th normalized eigenvector $\tilde{\alpha}^{(l)}$, respectively.

The following inequalities hold when using the RBF kernel:

$$0 \leq K_{i,\text{test}} \leq 1, \quad 0 \leq \bar{K}_{i,\text{row}} \leq 1,$$
$$0 \leq \bar{K}_{i,\text{test}} \leq 1, \quad 0 \leq \bar{K} \leq 1.$$

As a result, the range of $\left|\tilde{K}_{i,\text{test}}\right|$ may be given as:

$$0 \leq \left|\tilde{K}_{i,\text{test}}\right| = \left|K_{i,\text{test}} - \bar{K}_{i,\text{row}} - \bar{K}_{i,\text{test}} + \bar{K}\right| \leq 2. \tag{5.35}$$

The upper bound of T^2_{test} is:

$$\begin{aligned} T^2_{\text{test}} &\leq \sum_{l=1}^{L} \lambda_l^{-1} \left[\sum_{i=1}^{n} 4\tilde{\alpha}_i^{(l)2} + \sum_{i \neq k}^{n} 4 \left|\tilde{\alpha}_i^{(l)} \tilde{\alpha}_k^{(l)}\right| \right] \\ &= 4 \sum_{l=1}^{L} \lambda_l^{-1} \left[\sum_{i=1}^{n} \sum_{k=1}^{n} \left|\tilde{\alpha}_i^{(l)} \tilde{\alpha}_i^{(k)}\right| \right] \end{aligned} \tag{5.36}$$

which is dependent only on the $\tilde{\boldsymbol{\alpha}}$s and λs obtained in the training procedure. Eqn (5.36) shows that the monitoring statistic T^2 of all possible samples has an upper bound when the training data and the kernel width are both fixed.

Large $\boldsymbol{x}_{\text{test}}$

This section examines the extreme case of faults, i.e. the $\boldsymbol{x}_{\text{test}}$ deviates significantly from the training data. Assuming a test sample $\boldsymbol{x}_{\text{test}}^{(\text{lim})}$ has sufficiently large distances from all training samples such that:

$$\begin{aligned} K_{i,\text{test}}^{(\text{lim})} &= k(\boldsymbol{x}_{\text{test}}^{(\text{lim})}, \boldsymbol{x}_i) \\ &= \exp\left(-\frac{(\boldsymbol{x}_i - \boldsymbol{x}_{\text{test}}^{(\text{lim})})^\top (\boldsymbol{x}_i - \boldsymbol{x}_{\text{test}}^{(\text{lim})})}{\delta^2}\right) = 0 \\ \text{for } i &= [1, 2, \ldots, n], \end{aligned} \tag{5.37}$$

the centered kernel vector of this test sample is:

$$\begin{aligned} \tilde{K}_{i,\text{test}}^{(\text{lim})} &= K_{i,\text{test}}^{(\text{lim})} - \bar{K}_{i,\text{tow}}^{(\text{lim})} - \bar{K}_{i,\text{test}} + \bar{K} \\ &= \bar{K} - \bar{K}_{i,\text{row}} \end{aligned} \tag{5.38}$$

where $K_{i,\text{test}}^{(\text{lim})} = \bar{K}_{i,\text{test}}^{(\text{lim})} = 0$.

The T^2 statistic in this case becomes:

$$\begin{aligned} T_{\text{test,lim}}^2 &= \sum_{l=1}^{L} \lambda_l^{-1} \left[\sum_{i=1}^{n} \tilde{\alpha}_i^{(l)} \tilde{K}_{i,\text{test}}^{(\text{lim})}\right]^2 \\ &= \sum_{l=1}^{L} \lambda_l^{-1} \left(\sum_{i=1}^{n} \tilde{\alpha}_i^{(l)} \left[\bar{K} - \bar{K}_{i,\text{row}}\right]\right)^2 \end{aligned} \tag{5.39}$$

which is a constant when the kernel matrix K of the training data is known. It can be seen that the monitoring statistic T^2 will converge to this constant value when the test sample deviates significantly from the training samples.

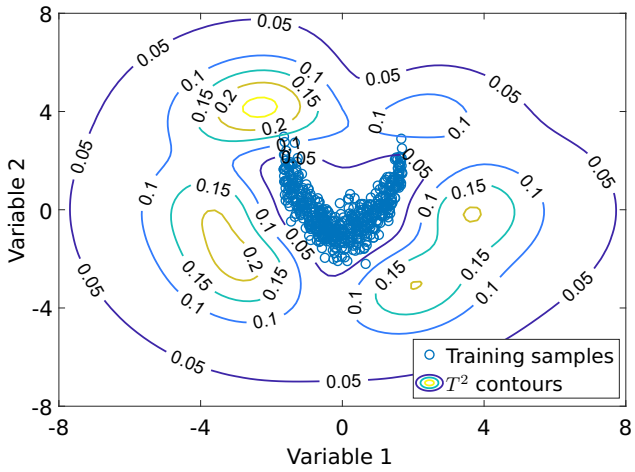


Figure 5.6: T^2 contour for the illustrative example when $\delta^2 = 1$

Large δ and large $\boldsymbol{x}_{\text{test}}$

Fig. 5.6 shows the contour plot of T^2 when zooming out Fig. 5.3(c) to a larger scale. It shows that T^2 is non-monotonic in all directions as d increases. Fig. 5.5(b) further suggests that a common turning point of T^2 exists when the δ is exceedingly large.

For a given large δ value, the common turning point of

$$\hat{d}_{\text{test}} = \delta/\sqrt{2} \tag{5.40}$$

exists for T^2 statistic when the nonlinear part in RBF kernels becomes dominant. A detailed derivation of the turning point is given in the Appendix A for the case where $\boldsymbol{x}_{\text{test}}$ is one-dimensional and $d_{\text{test}} = |x_{\text{test}}|$. The turning point is in accordance with the observation in Fig. 5.5(b): for δ^2 =300, 500 and 1000, the turning points of T^2 are all at $d^2/\delta^2 = 0.5$.

Limitation of T^2 as a monitoring statistic

The main issue of T^2 as a monitoring statistic in RBF-KPCA is that it is non-monotonic. When the δ value is too large, T^2 will firstly increase then will decrease as $\boldsymbol{x}_{\text{test}}$ moves away from the training data according to Fig. 5.5(b). The non-monotonic behaviour indicates that, if the upper control limit of T^2 is selected such that a fault is detected if $T^2 > T^2_{\text{UCL}}$, more severe faults may be missed.

A value of δ may exist such that T^2 is monotonically decreasing when the test sample moves away from the training data. A lower control limit of T^2 could be used and a fault is

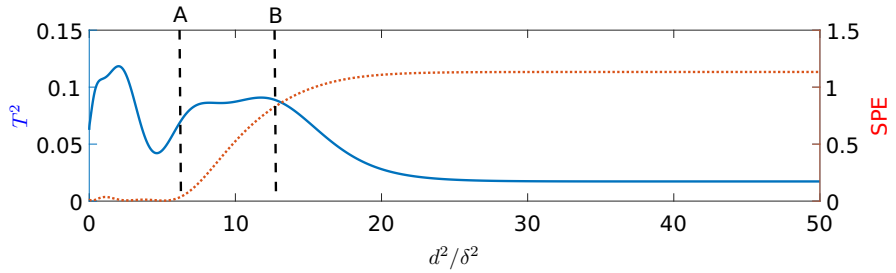


Figure 5.7: Asymptotic behaviour of SPE and T^2 when $\delta^2 = 1$. A: SPE begins to monotonically increase. B: T^2 begins to monotonically decrease.

detected when $T^2 < T^2_{\text{LCL}}$. However, since the T^2 can also be small when the test samples are close to the center of the training data (left side of the curves in Fig. 5.5(a)), the lower control limit will identify these test samples as faulty samples while they are within the training dataset, leading to the small detection contours in Fig. 5.5(b) and increased false alarms. In addition to having clear practical implications, such false alarms can lead to misleading results when tuning the kernel width δ using empirical approaches.

Fig. 5.7 compares the T^2 and the SPE with respect to d^2/δ^2 when $\delta^2 = 1$. It may be observed that T^2 has low values both when a test sample is located close to the center of the training data ($d^2 \to 0$) and when a test sample is located far from the training data ($d^2 \to \infty$). The value of T^2 in the latter case can be calculated using Eqn (5.39). In contrast the SPE is low when the test sample is close to the training data ($d^2 \to 0$) and rises as the test sample moves away from the training data ($d^2 \to \infty$). The reason is that, when the RBF kernel is tuned properly, the RBF-KPCA model is capable of capturing both the relationship between multiple variables and the healthy ranges of these variables. Therefore, the SPE, which measures the mismatch between a sample and the model, can detect both the violation of the relationship between variables and the violation of the healthy ranges of individual variables. This is different from the interpretation of linear monitoring models and T^2 and SPE in the linear models.

5.5 Tuning strategy for RBF-KPCA

Previous sections have demonstrated that the tuning of the kernel width δ influences the performance of RBF-KPCA. When training the RBF-KPCA model for fault detection, the dataset used for training is usually assumed to be from healthy operations, containing no samples that may be considered as faulty. A cross-validation approach for tuning δ divides

the data from healthy operations into training and validation sets. The RBF-KPCA model with various initial guesses of δ is trained on the training set and the δ which achieves a low false alarm rate on the cross-validation set is chosen as the appropriate δ_{opt}.

However, on further inspection, this approach is found not sufficient. For instance in Fig. 5.3(d) presented at the end of Section 5.2.3, the small number of training samples lying outside the detection contour indicates that, even when δ is inappropriately large, there could only be a small number of alarms triggered on the original dataset because of the mismatch between model and data. Hence the cross-validation approach may not tune the δ correctly if the initial guesses of δ are in an incorrect range. Therefore, this section proposes a strategy for tuning the kernel width δ in RBF-KPCA. The strategy combines the estimation of δ based on the previous analysis and the cross-validation approach.

5.5.1 Maximum value of δ

It is important to avoid a too large δ value because large δ values may impact the ability of RBF-KPCA to capture data nonlinearity (e.g. Fig. 5.3(d)). Therefore, an upper bound of δ is important. According to Eqn (A.16) given in the Appendix A, it is possible to estimate the maximum value of δ by the following empirical equation:

$$\delta_{\text{max}} = \sqrt{2} d_{\text{train,max}} \tag{5.41}$$

where $d_{\text{train,max}}$ is the maximum distance defined from the training set, i.e. $d_{\text{train,max}} = \max \sqrt{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2}$ for $\boldsymbol{x}_i, \boldsymbol{x}_j \in X_{\text{train}}$. A criterion for a maximum value of δ is required such that the RBF kernels in Eqn (5.7) are sufficiently localized without being over-fitted. This can be achieved if the values of $\boldsymbol{z}_{\text{test}}$ from Eqn (5.10) decrease monotonically when the test sample $\boldsymbol{x}_{\text{test}}$ is located outside the training data set. This is achieved for the same value of d/δ as in Eqn (5.40). Setting δ_{max} such that the largest distance between the training samples ($d_{\text{train,max}}$) can be accounted for leads to Eqn (5.41).

5.5.2 The tuning strategy

After δ_{max} is estimated, the appropriate δ value will be determined by the cross-validation performance. Eqn (5.32) shows that the kernel PCs converge to finite values, indicating that

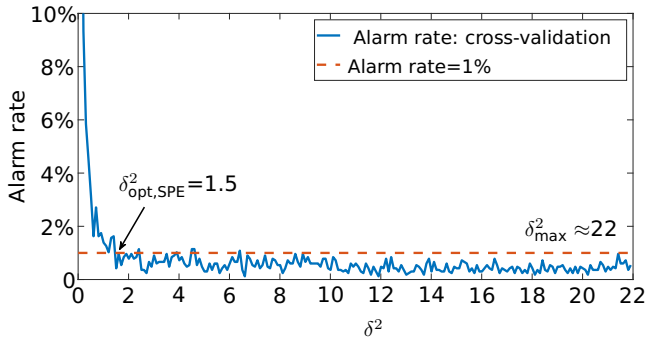


Figure 5.8: Alarm rates and $\delta^2_{\text{opt,SPE}}$ for the illustrative example

the kernel PCs cannot have a Gaussian distribution. Thus the analytic form for the distribution of SPE for RBF-KPCA is not known. Therefore, it is recommended to use a percentile of the SPEs of the training data as the control limit of SPE with a certain confidence level.

The strategy for tuning the kernel width in RBF-KPCA is:

1. Estimating the upper limit of δ by $\delta_{\text{max}} = \sqrt{2}d_{\text{train,max}}$;
2. Enumerating between $\delta = 0$ and $\delta = \delta_{\text{max}}$ to get the alarm rates on the cross-validation set. In this step a provisional control limit of a monitoring statistic is set so that the alarm rate is expected to be minimized (usually the maximum value of the monitoring statistic of the training set);
3. Setting the smallest δ that leads to an acceptable level of alarm rates on the cross-validation set as δ_{opt};
4. Specifying the final control limit of the monitoring statistic for fault detection.

For the illustrative example in Fig. 5.3, Fig. 5.8 shows the alarm rates with respect to δ^2 and the optimal δ value estimated when using the SPE as the monitoring statistic. When the acceptable level of alarm rates is set as 1%, the optimal kernel width value $\delta^2_{\text{opt,SPE}}$ should be 1.5 because it is the smallest kernel width that gives an alarm rate that is below 1%.

5.6 Examples of the influence of tuning and the behaviour of monitoring statistics

This section presents two examples. The first example is a synthetic dataset and this example demonstrates the influence of tuning on the performance of fault detection and the

behaviour of monitoring statistic SPE with respect to the tuning parameters. The second example uses the PRONTO dataset to validate that the findings discussed in previous sections are also valid for high-dimensional, real-life data.

5.6.1 Performance of RBF-KPCA on a synthetic dataset

The first example is based on the illustrative example described in Section 5.2.3. This section compares the fault detection performance of RBF-KPCA with various δ values using this example. Set 1 is the healthy set with 500 samples in the illustrative example. It is randomly divided into training and cross-validation sets with 250 samples in each set. The δ value is tuned using the training and the cross-validation sets by the strategy proposed in Section 5.5.2, as shown in Fig. 5.8. The control limit is set such that one percent of the values of the monitoring statistic obtained on the training set exceeds this control limit. The RBF-KPCA fault detection model is trained accordingly using Set 1.

Set 2 comprises another 500 healthy samples generated using Eqn (5.16) with different values of the random variables than for Set 1. The performance of the RBF-KPCA model on Set 2 is used to evaluate the robustness of the RBF-KPCA approach. Set 3 is an anomalous data set used for validating the fault detection performance. The blue circles and the red crosses in Fig. 5.9 represent Set 2 and Set 3, respectively. A fault detection approach should be able to identify the samples in Set 2 as healthy data and detect the samples in Set 3 as faulty samples.

Various δ values are used to demonstrate the influence of kernel widths. The values of δ^2 are chosen to be 0.2, 1.5, 5, 10 and 100. SPE is chosen as the monitoring statistic. Fig. 5.9 compares the detection contours generated by the upper control limits of SPE. The contour obtained by PCA is also visualized (denoted as "Linear" in Fig. 5.9). Fig. 5.9 shows that the optimal value $\delta^2_{\text{opt,SPE}} = 1.5$ can generate a good detection contour while the contour is over-fitted when $\delta^2 = 0.2$, which is smaller that the optimal value, and the contour becomes loose when δ increases ($\delta^2 = 10$).

For quantitative comparison, the False Alarm Rate (FAR) for Set 2 and the Missed Alarm Rate (MAR) for Set 3 are defined as:

$$\text{FAR} = \frac{n_{\text{AD}}}{n_{\text{Set2}}}, \quad \text{MAR} = \frac{n_{\text{ND}}}{n_{\text{Set3}}} \tag{5.42}$$

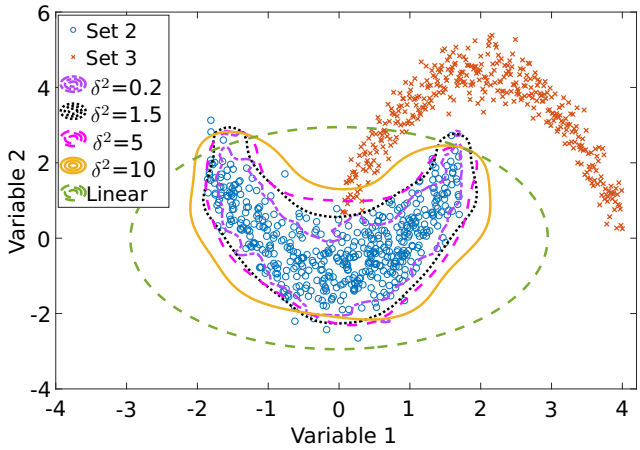


Figure 5.9: Detection contours obtained by RBF-KPCA and PCA

where n_{AD} is the number of faulty samples detected in Set 2 and n_{Set2} is the number of samples in Set 2. n_{ND} denotes the number of samples which are detected as healthy samples in Set 3 and n_{Set3} is the number of samples in Set 3. The FAR represents the robustness of the monitoring model to random variations in the healthy data. Since the confidence level of the control limit is 1%, the FAR should be close to 1%. The MAR represents the sensitivity of the monitoring model to faults. By inspecting Set 2 and Set 3 in Fig. 5.9, a good monitoring model should have no missed alarms, i.e. MAR being zero, because the two sets do not overlap.

Table 5.2 compares the quantitative performance, i.e. the FAR on Set 2 and the MAR on Set 3, of the RBF-KPCA approach with various δ values and the linear PCA approach. In particular, Fig. 5.8 in Section 5.5.2 shows that $\delta^2_{\text{opt,SPE}}$, the optimal kernel width when using SPE, should be 1.5. It can be observed that, relative to other combinations, the SPE with $\delta^2_{\text{opt,SPE}} = 1.5$ can achieve a MAR equal to zero with an FAR close to 1%. The δ value smaller than the $\delta_{\text{opt,SPE}}$ results in an over-fitted model which also achieves a MAR of zero, but with a high FAR. This indicates that the model is not robust to the randomness in the healthy data. Moreover, larger δ values (e.g. $\delta^2 = 100$ and 20000) may also achieve low FARs as the detection contours become relaxed. However, since the contours achieved by these δ values do not match the profile of the healthy data well, the monitoring model cannot differentiate properly between the healthy data and the faulty data. Thus the MAR increases as δ increases.

An extreme case occurs when linear PCA is applied. In this case, T^2 is used as the monitoring statistic in linear PCA because two PCs are retained. A data sample can always be reconstructed by the PCA model with two PCs in this bivariate example. Therefore, SPE

Table 5.2: Quantitative performance for various δ values

	SPE					Linear
δ^2	0.2	1.5	5	10	100	N/A
FAR (%)	17.4	2.2	2	1.4	1	0.8
MAR (%)	0	0	1.52	3.3	10.91	16.24

is not applicable as a monitoring statistic when linear PCA is applied. The FAR is low while the MAR is high since the contour in Fig. 5.9 achieved by linear PCA is different from the profile of the healthy data. This further indicates that, although various δ values may achieve similar FARs in cross-validation, their performance in fault detection can be different. A cross-validation strategy that purely minimizes false alarms is insufficient for tuning δ.

5.6.2 Performance of RBF-KPCA on the PRONTO dataset

Dataset description

This example uses the process data from the PRONTO benchmark case study introduced in Chapter 4. The data used for training were recorded when the facility was operating in the healthy mode with 120 $m^3\,h^{-1}$ inlet air and 0.1 $kg\,s^{-1}$ inlet water. As introduced in Section 4.3.2, the blockage fault was manually induced in this mode by closing the valve V11. The valve opening sequence for inducing this fault is shown in Fig. 5.10. Fig. 5.11 plots the time trends and the tags of the process variables used in the example. It can be observed that the deviation of process measurements becomes visible as the fault severity increases. Therefore, this faulty dataset includes a fault with low and high levels of severity, making it suitable for demonstrating the performance of fault detection.

Results and discussions

When applying the RBF-KPCA approach to this dataset, the δ is tuned based on the strategy proposed in Section 5.5.2. The results for the optimum value of δ^2 are shown in Fig. 5.12. A fault is detected when the monitoring statistic exceeds its control limit for a continuous sequence of 50 samples. The reason for setting this criterion for fault detection is to

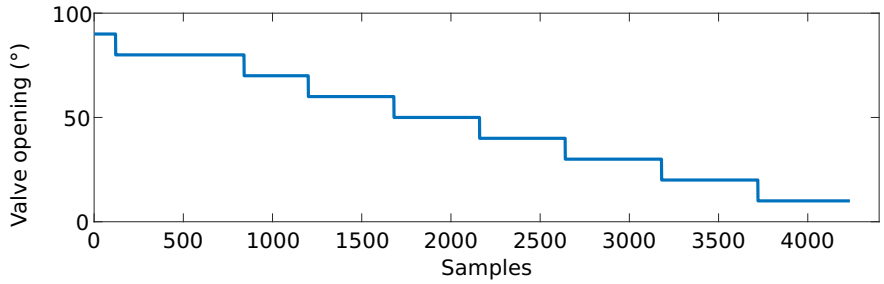


Figure 5.10: Operating sequence for inducing air blockage

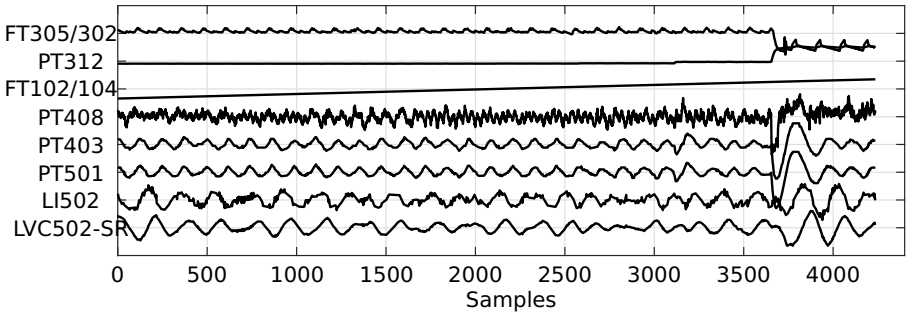


Figure 5.11: High density plot of the measurements under faulty condition

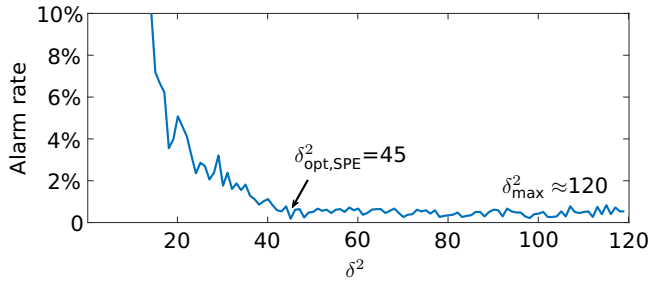
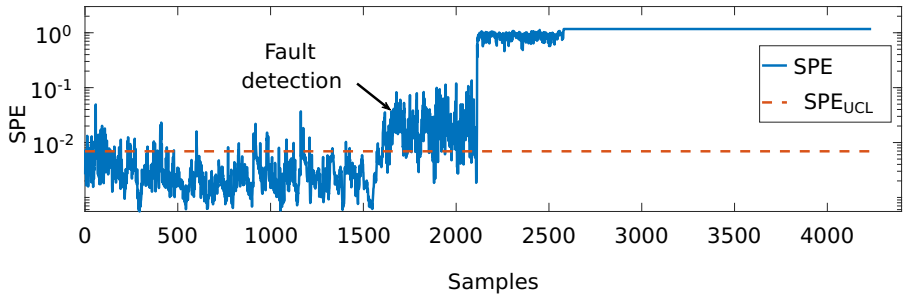


Figure 5.12: Alarm rates and $\delta^2_{\text{opt,SPE}}$ for the experimental dataset

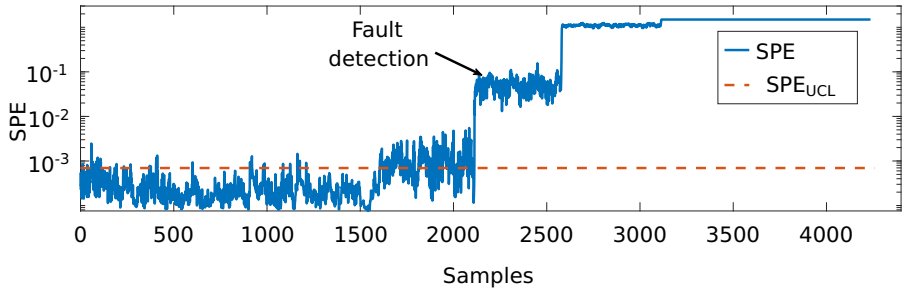
minimize false alarms. The noise in process measurements may trigger false alarms. Nevertheless, the likelihood of observing a sequence of false alarms is low. On the other hand, faults in the process are often persistent and may result in a sequence of abnormal process data. Therefore, a fault is only detected when the monitoring statistic exceeds its control limit for a sequence of samples.

Figs 5.13(a) to 5.13(c) show the performance of SPE. The SPE obtained by RBF-KPCA with $\delta^2_{\text{opt,SPE}} = 45$ (Fig. 5.13(a)) can detect the blockage fault earlier than the cases when δ is exceedingly large (Fig. 5.13(b)) or when the SPE is calculated from linear PCA (Fig. 5.13(c)).

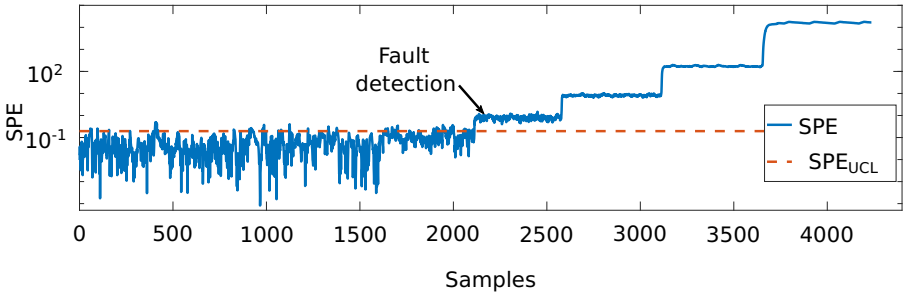
Figs 5.14 to 5.17 compare the performance of T^2 obtained by RBF-KPCA with $\delta^2 = 2, 45$ and 1500 and by PCA, respectively. When δ is small, Fig. 5.14 shows that T^2 decreases when a fault occurs. This observation explains the decision in Choi et al. (2005a) to use a



(a) Trend plot of SPE for RBF-KPCA with $\delta^2_{\text{opt}} = 45$



(b) Trend plot of SPE for RBF-KPCA with $\delta^2 = 500$



(c) Trend plot of SPE for PCA

Figure 5.13: Trend plots of SPE obtained by various methods

lower control limit for T^2. In Fig. 5.15, where $\delta^2 = 45$, T^2 first increases then reduces with respect to the fault development. When δ is inappropriately tuned (Fig. 5.16) and the upper control limit is used, T^2 can detect the fault when it is less severe while more severe faults will be missed due to the non-monotonicity of T^2. The non-monotonicity issue of T^2 does not exist when linear PCA is applied (Fig. 5.17). However, linear PCA with T^2 has a later detection when compared with the result in Fig. 5.13(a) because linear PCA cannot capture data nonlinearity.

This example shows that the behaviour of T^2 can be misleading in RBF-KPCA when δ is exceedingly large. In such a situation T^2 might be increasing when the fault is less severe and the upper control limit can detect the fault. However, since T^2 is non-monotonic, it

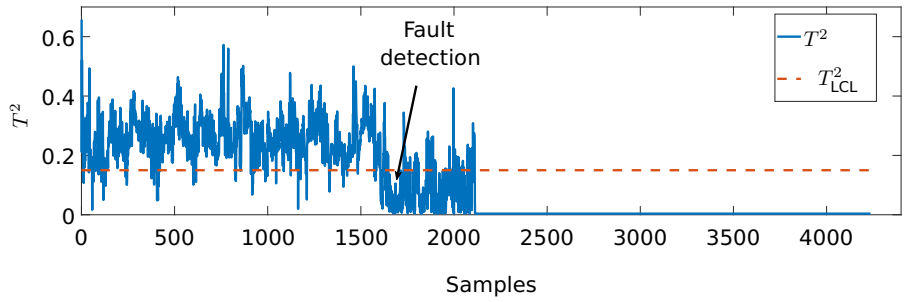


Figure 5.14: Trend plot of T^2 for RBF-KPCA with $\delta^2 = 2$

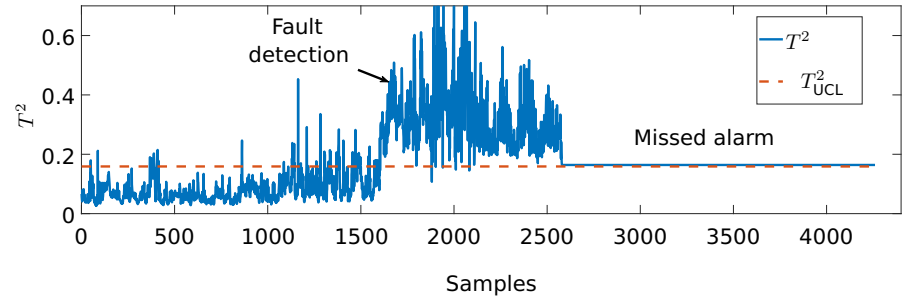


Figure 5.15: Trend plot of T^2 for RBF-KPCA with $\delta^2 = 45$

drops below the upper control limit when the fault gets more severe, leading to the RBF-KPCA approach failing to detect the severe fault. On the other hand, the performance of SPE for RBF-KPCA is not influenced by the non-monotonicity issue. RBF-KPCA with properly tuned δ values and SPE as the monitoring statistic can detect both mild and severe faults. Compared to the numerical example with two variables, this example with eight variables also shows that the findings on the SPE and T^2 for RBF-KPCA can be generalized to higher dimensional problems.

The behaviour of T^2 is unpredictable as δ changes in RBF-KPCA. In particular, it is advised against using T^2 and large δ values because RBF-KPCA with such tuning will fail to detect severe faults, which may quickly develop into significant failures.

5.7 Chapter summary

This chapter investigated the tuning of RBF kernels and the behaviour of monitoring statistics in kernel methods. Excessively large kernel widths result in under-fitted monitoring models while excessively small kernel widths result in over-fitted models. The tuning strategy proposed in this chapter can avoid inappropriate kernel widths.

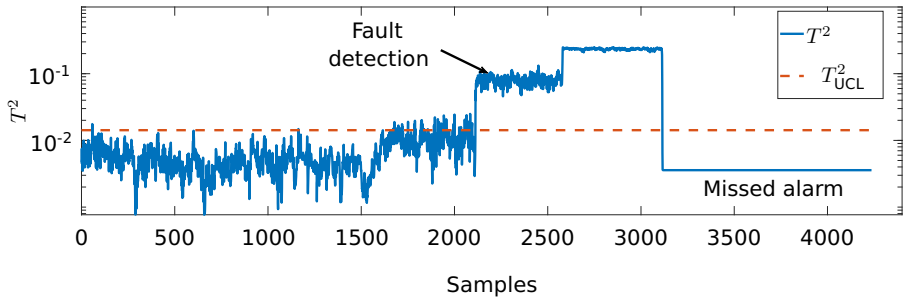


Figure 5.16: Trend plot of T^2 for RBF-KPCA with $\delta^2 = 1500$

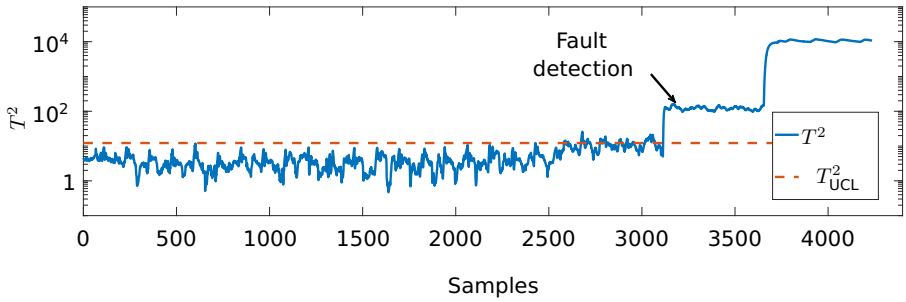


Figure 5.17: Trend plot of T^2 for PCA

The results prove that the behaviour of SPE and T^2 as monitoring statistics in RBF-KPCA is different from the behaviour of SPE and T^2 in linear methods. The results for RBF-KPCA show that T^2 is non-monotonic with respect to the severity of the fault, making it inappropriate as a monitoring statistic. Under the RBF-KPCA framework, the SPE is a better monitoring statistic because it can detect both faulty samples that exceed the healthy range of variables and faulty samples which do not have the same behaviour as the healthy data. The SPE as formulated for RBF-KPCA can detect faulty samples that would require both T^2 and SPE in a linear method. Moreover, the SPE for RBF-KPCA increases monotonically as the fault becomes more severe, making it possible to set an upper control limit for fault detection, which cannot be adopted for T^2 due to its non-monotonicity.

Both numerical simulation and the PRONTO dataset verified the findings about kernel widths and monitoring statistics. These findings, including the tuning strategy, will help achieving the desirable behaviours of kernel-based monitoring approaches.

Chapter 6

Non-stationary discrete convolution kernel for multimodal process monitoring

In this chapter a novel non-stationary discrete convolution kernel is introduced. This kernel addresses the challenges posed by multiple operating modes in process monitoring. This chapter addresses the second open question in Section 3.5.1. The question is about new kernels that are suitable for building monitoring models for multimodal processes. The chapter will first demonstrate the limitation of applying RBF-KPCA to multimodal data.

The new kernel is non-stationary because the value of the kernel function depends on the distance between two samples, and also on the values of these two samples. Therefore, the new kernel can describe the covariance structure of each operating mode. In contrast, the RBF kernel previously investigated in Chapters 5 is a stationary kernel because the kernel only depends on the distance between two samples. A discussion on the monitoring statistics and the parameter tuning for this new kernel facilitates the application of the new kernel to process monitoring.

Section 2.4 described that monitoring methods should be robust to new healthy operating modes whilst being able to detect faults in these new modes. This chapter proposes an on-line monitoring framework to account for new operating modes appearing during process operations. This on-line framework combines a data-driven clustering method with Kernel

Principal Component Analysis (KPCA) using the new kernel. The on-line framework enables on-line model update and makes it possible for the proposed method to detect faults even in new operating modes. The chapter compares the off-line performance in fault detection of the new kernel with the performance of the RBF kernel and other kernel-based methods proposed for multimodal process monitoring. Two examples using the PRONTO dataset will demonstrate the performance of the proposed new kernel in off-line fault detection and on-line model update.

6.1 Background

This section discusses the approach taken to achieve the desirable behaviour of monitoring systems presented in Chapter 2. The section gives the background for the investigations in the rest of this chapter.

Section 3.4.1 discussed the various structures of monitoring methods for multimodal processes. Chapter 3 also concluded that the structure with a single monitoring model can achieve monitoring statistics that are easily interpretable for end-users whilst being robust to changes in operating modes. Therefore, the thesis proceeds with the structure with a single monitoring model in order to achieve the desirable behaviour described in Section 2.4 and to address the challenges presented in Section 2.6. However, it is shown that RBF kernels have limitations in accounting for multimodal data regardless of the tuning. Therefore, the kernel should be specified properly so as to make KPCA a good candidate for building a single monitoring model. In this chapter, a new kernel called the Non-stationary Discrete Convolution (NSDC) kernel is developed such that this new kernel meets these requirements.

It was noted in Chapter 2 that monitoring systems should generate interpretable monitoring results, such as a monitoring statistic that is useful for process operators. In particular, operators need to detect the presence of faults with false and missed alarms minimized. This requires a monitoring statistic that has the same interpretation regardless of the mode in which the process is operating. The new NSDC kernel also aims to fulfil this requirement.

The chapter will first demonstrate the limitation of RBF kernels in accounting for multimodal data via an illustrative example. Then a new kernel for capturing multimodal

data will be proposed. The tuning issue of this new kernel will be discussed in a similar way to the analysis of RBF kernels in Chapter 5. An on-line framework using clustering approaches and the new kernel will enable the on-line model update to account for new operating modes. Several examples using numerical simulations validate the ability of the new kernel to handle multimodal data. The on-line monitoring framework is validated using the PRONTO dataset. The work in this chapter has generated the following papers. Tan et al. (2020) presents the new kernel. A conference paper (Tan et al., 2019) and a journal paper (Tan et al., 2020a) describe the on-line framework and implementation considerations associated. The work in this chapter was done in collaboration with Tian Cong from AGH University of Science and Technology in Krakow, Poland. Tian and the author of the thesis collaborated to develop the on-line monitoring framework.

6.2 Limitation of RBF kernels in multimodal process monitoring

This section uses a simulated example with multiple operating modes to illustrate the performance of RBF-KPCA in multimodal process monitoring. The bivariate example has four operating modes and one mode has a nonlinear model. The algebraic models for the four modes are as follows. A hundred samples are drawn randomly from each to formulate the training set.

Mode 1:

$$\begin{aligned} x_1 &= e_{11} \\ x_2 &= 1.5x_1 + e_{12} \end{aligned} \tag{6.1}$$

where $e_{11} \sim N(0,1)$ and $e_{12} \sim N(0,9)$.

Mode 2:

$$\begin{aligned} x_1 &= e_{21} + 8 \\ x_2 &= -0.2x_1 + 5 + e_{22} \end{aligned} \tag{6.2}$$

where $e_{21} \sim N(0,2.25)$ and $e_{22} \sim N(0,0.25)$.

Mode 3:

$$\begin{aligned} x_1 &= e_{31} + 15 \\ x_2 &= -x_1 + 20 + e_{32} \end{aligned} \tag{6.3}$$

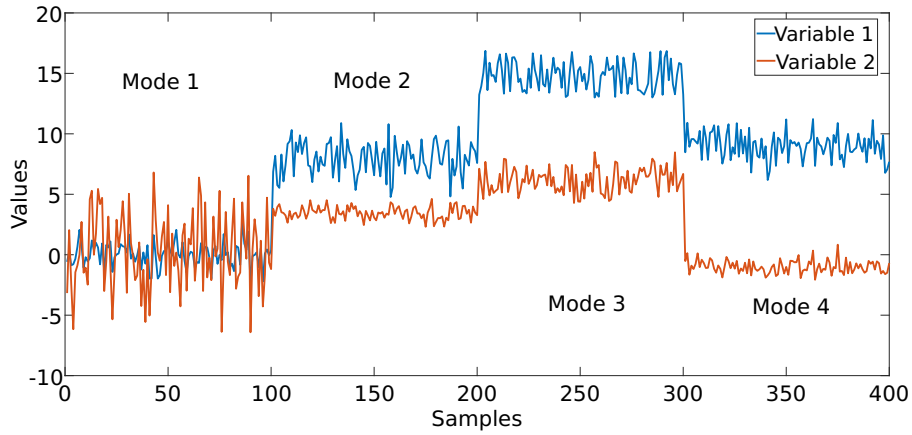


Figure 6.1: Trend plot of illustrative example

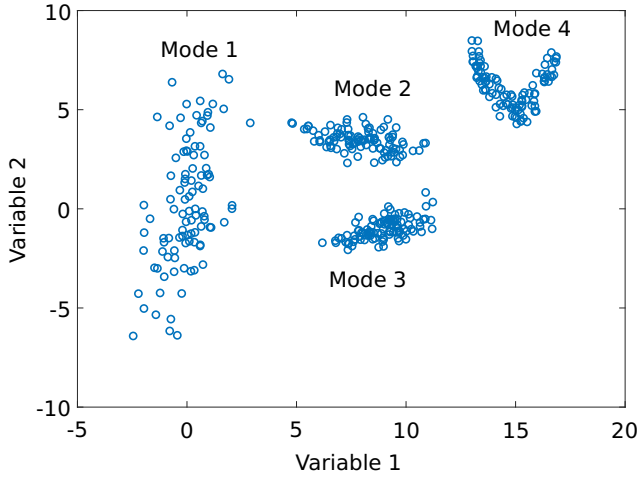


Figure 6.2: Scatter plot of illustrative example

where $e_{31} \sim N(0, 0.25)$ and $e_{32} \sim N(0, 0.09)$.

Mode 4:

$$\begin{aligned} x_1 &= e_{31} + 15 \\ x_2 &= (x_1 - 15)^2 + 5 + e_{32} \end{aligned} \tag{6.4}$$

where $e_{31} \sim U[-2, 2]$ and $e_{32} \sim N(0, 0.09)$. Figs 6.1 and 6.2 visualizes the trend plot and the scatter plot of process variables achieved by the multimode model. It may be observed from the trend plot in Fig. 6.1 and from the scatter plot in Fig. 6.2 that the variance of the dataset from Mode 1 is larger than the variances of the other three sets. In practice, such differences in variance may exist due to the nonlinearity in process variables or measurement instruments. For example, a flow measurement might have higher measurement variability if air is entrained in the process fluid.

The performance of RBF-KPCA on this dataset demonstrates a limitation of RBF kernels. As suggested in Chapter 5, this example uses SPE as the monitoring statistic. In order to

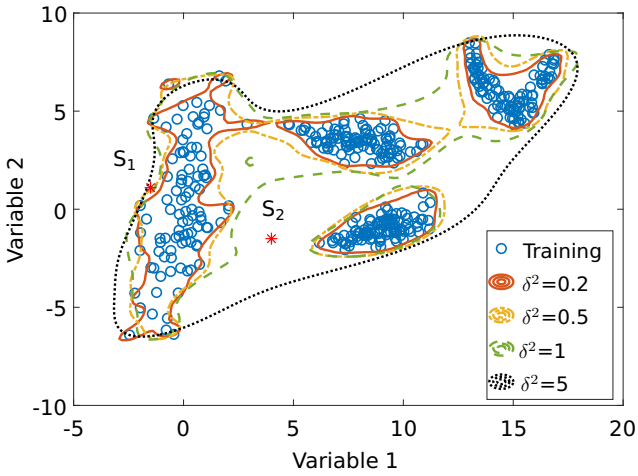


Figure 6.3: Detection contours by RBF-KPCA for illustrative example

show the boundaries for fault detection, 99% control limits of SPE obtained after KPCA with several values of the kernel width δ are visualized alongside the original samples in Fig. 6.3.

The results demonstrate the issue of RBF-KPCA. S_1 is a sample which is likely to be healthy. However, the contours obtained by RBF-KPCA when $\delta^2 = 0.2, 0.5$ and 1 are over-fitted and, as a result, S_1 is identified as faulty. On the other hand, sample S_2 does not belong to any of the four modes. Thus S_2 should be identified as faulty. When $\delta^2 = 5$, S_1 is located within the contour and, as a result, is correctly classified as a healthy sample. However, this contour is under-fitted and sample S_2 is located also within this contour. Hence S_2 will be erroneously considered as healthy,

The illustrative examples in this section demonstrate that, regardless of the tuning of the kernel width, the RBF-KPCA method is not able to fully account for multimodal data. Therefore, it is necessary to consider other kernels that can build accurate monitoring models for multimodal data.

6.3 Non-stationary Discrete Convolution Kernel

This section derives the formulation of the novel Non-stationary Discrete Convolution (NSDC) kernel by extending the formulation of the RBF kernel.

6.3.1 The NSDC kernel as a covariance function

A kernel function is defined for two samples, for example $\boldsymbol{x}$ and $\boldsymbol{x}^*$, when applied to process data. One way to construct a kernel function is to define it as a covariance function. If y is defined as Eqn (6.5), then the covariance of y can be used as a kernel function of $\boldsymbol{x}$. Eqn (6.5) uses p basis functions $\phi_i(\boldsymbol{x})$, where $i = 1, 2, \cdots, p$:

$$y = \sum_{i=1}^{p} w_i \phi_i(\boldsymbol{x}) \tag{6.5}$$

where $w_i \sim N\left(0, \sigma_w^2\right)$ are the regression coefficients with independent and identical Gaussian distributions corresponding to basis functions $\phi_i(\boldsymbol{x})$ for $i = 1, 2, \cdots, p$.

The covariance of two new output samples, y and y^*, can be calculated as a function of input samples, namely $\boldsymbol{x}$ and $\boldsymbol{x}^*$. This is known as the kernel function:

$$\begin{aligned} \operatorname{cov}(y, y^*) &= k(\boldsymbol{x}, \boldsymbol{x}^*) \\ &= \mathbb{E}\left[\sum_{i=1}^{p} w_i \phi_i(\boldsymbol{x}) \sum_{i=1}^{p} w_i \phi_i(\boldsymbol{x}^*)\right] \\ &= \Phi^\top(\boldsymbol{x}) \mathbb{E}\left[\boldsymbol{w}^\top \boldsymbol{w}\right] \Phi(\boldsymbol{x}^*) \end{aligned} \tag{6.6}$$

where $\Phi(\boldsymbol{x}) = \{\phi_1(\boldsymbol{x}), \phi_2(\boldsymbol{x}), \cdots, \phi_p(\boldsymbol{x})\}$ contains the features constructed by the kernel function, using $\boldsymbol{x}$. $\boldsymbol{w} = \{w_1, w_2, \cdots, w_p\}$ is the coefficient vector and $\mathbb{E}\left[\boldsymbol{w}^\top \boldsymbol{w}\right]$ denotes the expectation of $\boldsymbol{w}^\top \boldsymbol{w}$. Since w_i are independently and identically distributed following a Gaussian distribution with variance σ_w^2, Eqn (6.6) can be written as:

$$\operatorname{cov}(y, y^*) = \sigma_w^2 \sum_{i=1}^{p} \phi_i(\boldsymbol{x}) \phi_i(\boldsymbol{x}^*). \tag{6.7}$$

When the radial basis function is adopted in Eqn (6.5) with $\boldsymbol{c}^{(i)}$ as its center, the covariance of y and y^* is:

$$\operatorname{cov}(y, y^*) = \sigma_w^2 \sum_{i=1}^{p} \left[\exp\left(-\frac{(\boldsymbol{x} - \boldsymbol{c}^{(i)})^\top (\boldsymbol{x} - \boldsymbol{c}^{(i)})}{l^2}\right) \exp\left(-\frac{(\boldsymbol{x}^* - \boldsymbol{c}^{(i)})^\top (\boldsymbol{x}^* - \boldsymbol{c}^{(i)})}{l^2}\right)\right] \tag{6.8}$$

where $\boldsymbol{c}^{(i)}$ is the center of the i-th radial basis function.

The formulation of RBF kernels assumes an infinite number of basis functions, $p \to \infty$. $\boldsymbol{c}^{(i)}$ are allocated evenly from $-\infty$ to ∞ and $\mathrm{d}\boldsymbol{c} = \boldsymbol{c}^{(i+1)} - \boldsymbol{c}^{(i)}$. Eqn (6.8) becomes:

$$\begin{aligned}\operatorname{cov}(y, y^*) &= \sigma_w^2 \lim_{p\to\infty} \sum_{i=1}^{p} \phi_{\boldsymbol{c}^{(i)}}(\boldsymbol{x})\phi_{\boldsymbol{c}^{(i)}}(\boldsymbol{x}^*) \\ &= \sigma_0^2 \int_{-\infty}^{\infty} [\phi_{\boldsymbol{c}}(\boldsymbol{x})\phi_{\boldsymbol{c}}(\boldsymbol{x}^*)]\,\mathrm{d}\boldsymbol{c} \\ &= \sqrt{2\pi}\delta\sigma_0^2 \exp\left(-\frac{(\boldsymbol{x}-\boldsymbol{x}^*)^\top(\boldsymbol{x}-\boldsymbol{x}^*)}{\delta^2}\right)\end{aligned} \tag{6.9}$$

where σ_w^2 is selected as σ_0^2/p and σ_0 is a finite constant. $\delta^2 = 2l^2$ is the kernel width of the RBF kernel. Such selection avoids the covariance value approaching to infinity when $p \to \infty$ by setting the coefficient σ_w^2 as a function of p. Appendix B gives a detailed reasoning regarding why such selection of σ_w^2 can guarantee the covariance value in Eqn (6.9) is bounded. In RBF-KPCA, the infinite number of basis functions result in the infinite dimensions of the feature space Φ, which was discussed in Section 5.2.2.

Eqn (6.9) is the convolution of two Gaussian functions with variables $\boldsymbol{x}$ and $\boldsymbol{x}^*$. This convolution formulation leads to a scaled and multivariate formulation of the RBF kernel function presented in Eqn (3.13) of Chapter 3. The covariance matrix K in kernel-based methods can be calculated accordingly.

Like the RBF kernel (Eqn (6.9)), the NSDC kernel also derives from Eqn (6.8). In the formulation of the NSDC kernel, the kernel is made data-dependent by selecting only the training samples as $\boldsymbol{c}^{(i)}$, the centers of the basis functions in Eqn (6.8). Assuming that P clusters of healthy training samples obtained from P operating modes exist in $C \in \mathbb{R}^{m\times n}$, i.e. $C = C_1 \bigcup \cdots \bigcup C_P$, the kernel function can be defined using each and every sample in C as the centers of the basis function:

$$k_{\text{NSDC}}(\boldsymbol{x}, \boldsymbol{x}^*) = \operatorname{cov}(y, y^*) = \frac{\sigma_0^2}{n} \sum_{i=1}^{n} \phi_{\boldsymbol{c}^{(i)}}(\boldsymbol{x})\phi_{\boldsymbol{c}^{(i)}}(\boldsymbol{x}^*) \tag{6.10}$$

where $\boldsymbol{c}^{(i)} \in C$. It is important to note that the number of basis functions is equal to n, the number of samples from healthy operating modes. Since the centers $\boldsymbol{c}^{(i)}$ are discrete, Eqn (6.10) results in a calculation of discrete convolution. The univariate and multivariate solutions to the discrete convolution structure in Eqn (6.10) will yield the new NSDC kernel function. The next section will demonstrate how the covariance matrix of each mode can

be incorporated into the kernel function. For conciseness, k denotes k_{NSDC} for the rest of this chapter.

6.3.2 Univariate formulation

For simplicity, $\boldsymbol{x}$ is first assumed to be univariate. By using radial basis functions in Eqn (6.10), the NSDC kernel can be derived as:

$$\begin{aligned} k(x, x^*) &= \frac{\sigma_0^2}{n} \sum_{i=1}^{n} \left[\exp\left(-\frac{(x - c^{(i)})^2}{l^2} \right) \exp\left(-\frac{(x^* - c^{(i)})^2}{l^2} \right) \right] \\ &= \sigma_0^2 \exp\left(-\frac{d^2}{\delta^2} \right) \frac{1}{n} \sum_{i=1}^{n} \exp\left(-\frac{\left(c^{(i)} - \frac{1}{2}(x + x^*) \right)^2}{(\delta/2)^2} \right) \end{aligned} \tag{6.11}$$

where $d = x - x^*$ is the distance between x and x^*. Similarly to the formulation of RBF kernels, $\delta = \sqrt{2}l$ is assumed to be the kernel widths. This new kernel has a similar formulation to the RBF kernel. However, given $c^{(i)} \in C$, the weighting coefficient

$$\frac{1}{n} \sum_{i=1}^{n} \left[\exp\left(-\frac{\left(c^{(i)} - \frac{1}{2}(x + x^*) \right)^2}{(\delta/2)^2} \right) \right]$$

is proportional to the kernel density estimation of the conditional likelihood

$$P_{\text{kde}}\left(\frac{x + x^*}{2} | C \right),$$

which represents observing a sample with the value $(x + x^*)/2$ given the training set C. Therefore, the extra weighting coefficient makes this new kernel dependent on the training set C. In addition, this kernel is non-stationary as it is dependent not only on the distance d between two input samples, but also on the locations of these samples.

Moreover, when considering the auto-covariance of a single sample x^*, $d = 0$:

$$k(x^*, x^*) = \frac{\sigma_0^2}{n} \sum_{i=1}^{n} \exp\left(-\frac{\left(c^{(i)} - x^* \right)^2}{(\delta/2)^2} \right) \tag{6.12}$$

The auto-covariance of x^* is therefore proportional to the conditional likelihood of x^* given the training set C.

6.3.3 Multivariate extension

For the multidimensional case, $\boldsymbol{x}$ is taken to be an m-dimensional vector. The NSDC kernel is thus given as:

$$k(\boldsymbol{x}, \boldsymbol{x}^*) = \frac{\sigma_0^2}{n} \exp\left(-\frac{\boldsymbol{d}^\top \boldsymbol{d}}{\delta^2}\right) \times \sum_{i=1}^{n} \exp\left(-\frac{\left(\boldsymbol{c}^{(i)} - \frac{1}{2}(\boldsymbol{x} + \boldsymbol{x}^*)\right)^\top \left(\boldsymbol{c}^{(i)} - \frac{1}{2}(\boldsymbol{x} + \boldsymbol{x}^*)\right)}{(\delta/2)^2}\right) \quad (6.13)$$

Similarly to the univariate case, this revised kernel function is the product of the RBF kernel with respect to the distance $\boldsymbol{d} = \boldsymbol{x} - \boldsymbol{x}^*$ and the kernel density estimation of the following likelihood

$$P_{\text{kde}}\left(\frac{\boldsymbol{x} + \boldsymbol{x}^*}{2} \middle| C\right).$$

6.4 Using the NSDC kernel for process monitoring

Similarly to the results presented in Chapter 5, the parameters for the NSDC kernel will also influence the performance of process monitoring using the NSDC kernel. Moreover, the monitoring statistics should be selected according to the behaviour of the NSDC kernel. This section examines the monitoring statistics and discusses the tuning strategy for the NSDC kernel.

6.4.1 The monitoring statistics

The monitoring statistics are defined using the kernel PCs. Section 3.4.2 discussed the general formulation in KPCA and the kernel PCs $\boldsymbol{z}$ are extracted using the kernel matrix K and Eqn (3.12). The procedure is the same in NSDC-KPCA, except that the kernel matrix K is obtained by the NSDC kernel. In NSDC-KPCA, T^2 is defined as

$$T_*^2 = \boldsymbol{z}_*^\top \Lambda_{n_1}^{-1} \boldsymbol{z}_* \quad (6.14)$$

where $\boldsymbol{z}_* \in \mathbb{R}^{n_1 \times 1}$ are the kernel PCs of $\boldsymbol{x}^*$ obtained by KPCA with the NSDC kernel, Λ_{n_1} is an $n_1 \times N_1$ diagonal matrix with the first n_1 eigenvalues being its diagonal elements. The

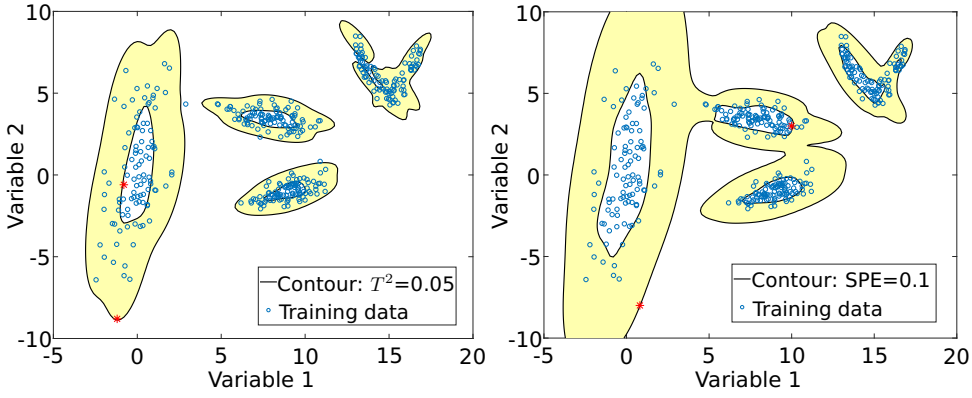
SPE formulation for the RBF-KPCA method was defined in Eqn (5.12) as

$$\begin{aligned} \text{SPE}_* &= \|\Phi(\tilde{\boldsymbol{x}}^*)\|_2 - 2\Phi^\top(\tilde{\boldsymbol{x}}^*)\cdot\Phi_0 + \|\Phi_0\|_2 - \boldsymbol{z}_*^\top\boldsymbol{z}_* \\ &= K_{**} - 2\bar{K}_* + \bar{K} - \boldsymbol{z}_*^\top\boldsymbol{z}_* \end{aligned} \tag{6.15}$$

where $\tilde{\boldsymbol{x}}^*$ is the centered $\boldsymbol{x}^*$, $\Phi(\tilde{\boldsymbol{x}}^*)$ are the unknown projections of $\boldsymbol{x}^*$ to the kernel space before applying PCA-based dimension reduction, and Φ_0 is the center of $\Phi(\tilde{\boldsymbol{x}}_i)$ for $i = 1, \cdots, n$. Although $\Phi(\tilde{\boldsymbol{x}}^*)$ and Φ_0 cannot be obtained directly, their second order norms, $\|\Phi(\tilde{\boldsymbol{x}}^*)\|_2$ and $\|\Phi_0\|_2$, can be calculated using the kernel matrix. $K_{**} = k(\boldsymbol{x}_*, \boldsymbol{x}_*)$ is the variance of $\boldsymbol{x}_*$ in the kernel space and $\bar{K}_*$ is the mean of the kernel vector $K_* = [k(\boldsymbol{x}_1, \boldsymbol{x}_*), \cdots, k(\boldsymbol{x}_n, \boldsymbol{x}_*)]$. It should be noted that, unlike the RBF kernel, $k(\boldsymbol{x}_*, \boldsymbol{x}_*)$ is not necessarily equal to 1 when using the NSDC kernel. The auto-covariance result in Eqn (6.12) illustrates this point.

The previous chapter showed that SPE is superior to T^2 as a monitoring statistic when the RBF kernel is used since $\Phi(\tilde{\boldsymbol{x}})$ has infinite dimensions while $\boldsymbol{z}_*$ only has finite dimension. As a result, SPE_* increases monotonically as $\boldsymbol{x}^*$ gradually deviates away from the training data. However, when using NSDC-KPCA neither T^2 nor SPE are ideal for anomaly detection as the indicators do not increase monotonically with the level of fault. More specifically, multiple samples that are faulty to different extents may have the same value of T^2 or SPE. For the numerical example in Fig. 6.4(a), although the two samples highlighted (red asterisks) are different because one is located within the training data cluster while the other is far away from the training data and is likely to be an anomaly, they have the same T^2 value. The shaded region defined by the contour is not suitable for fault detection because false alarms may be triggered when test samples are inside the training cluster. The same behaviour may be observed when considering SPE as shown in Fig. 6.4(b). The white "hole" in such contours are due to the non-monotonic behaviour of T^2 and SPE when NSDC kernel is used.

The reason is that, in the NSDC formulation, the dimension of $\Phi(\tilde{\boldsymbol{x}})$ is also finite ($N \times 1$) due to the number of basis functions in Eqn (6.10) being finite. As a result it is insufficient to only monitor the model-data mismatch in the NSDC-KPCA model. Therefore, a monitoring statistic should be able to monitor the variations that cause violation of the NSDC-KPCA model and the variations that cannot be modelled by the finite number of basis functions in the NSDC-KPCA model.



(a) T^2 (b) SPE

Figure 6.4: Non-monotonic behaviour of T^2 and SPE obtained by NSDC-KPCA. Yellow-shaded region: healthy region. White region: faulty region

A revised SPE is proposed to address this issue. The revised SPE is defined as the sum of the SPE defined in Eqn (6.15), which quantifies the violation of the NSDC-KPCA model, and an additional term $\|\Phi_{n+1:\infty}(\tilde{\boldsymbol{x}}^*)\|_2$ that includes the variations which are not modelled by NSDC-KPCA:

$$\begin{aligned}\text{SPE}_* &= \|\Phi_{1:n}(\tilde{\boldsymbol{x}}^*)\|_2 - 2\Phi_{1:n}^\top(\tilde{\boldsymbol{x}}^*)\cdot\Phi_0 + \|\Phi_0\|_2 - \boldsymbol{z}_*^\top\boldsymbol{z}_* + \|\Phi_{n+1:\infty}(\tilde{\boldsymbol{x}}^*)\|_2 \\ &= 1 - 2\Phi^\top(\tilde{\boldsymbol{x}}^*)\cdot\Phi_0 + \|\Phi_0\|_2 - \boldsymbol{z}_*^\top\boldsymbol{z}_* \\ &= 1 - 2\bar{K}_* + \bar{K} - \boldsymbol{z}_*^\top\boldsymbol{z}_*\end{aligned} \tag{6.16}$$

given that in the RBF kernel

$$k_{\text{RBF}}(\boldsymbol{x}^*,\boldsymbol{x}^*) = \|\Phi_{1:\infty}(\tilde{\boldsymbol{x}}^*)\|_2 = \|\Phi_{1:n}(\tilde{\boldsymbol{x}}^*)\|_2 + \|\Phi_{n+1:\infty}(\tilde{\boldsymbol{x}}^*)\|_2 = 1. \tag{6.17}$$

$\boldsymbol{x}^*$ is detected as an anomaly if the following detection criterion holds:

$$\text{SPE}_* > \text{SPE}_{\text{UCL}} \tag{6.18}$$

where SPE_{UCL} is the upper control limit of the revised SPE estimated from the training data. The distribution of SPE in KPCA does not have an analytic form because of the kernel transformation. In practice, Box et al. (1978) defined SPE_{UCL} by the uses of the empirical reference distribution of SPE values for healthy training data. This chapter will use the $(100-\eta)\%$ percentile of the SPEs obtained in the training data as the SPE_{UCL} with confidence level $\eta\%$.

6.4.2 Tuning the kernel parameters

The covariance of the two projections of two samples decreases with respect to the distance between the original samples. The kernel parameters regulate the behaviour of the kernel methods. Due to the multimodal nature of the training set (e.g. the covariance and the dynamic relationships of process variables), the rate of decrease depends on the underlying mechanisms of each operating mode and the variances of individual variables in the same operating mode. Therefore, it is necessary to specify the parameters properly. The parameters for the NSDC kernel can be estimated by the covariance matrix of process variables and a global scaling factor δ. This factor δ is the kernel width for the NSDC kernel. This section will present the covariance matrices and the next section discusses the tuning of the kernel width δ.

Kernel parameters for individual variables

By introducing the $m \times m$ covariance matrix $\Lambda = \mathrm{cov}(\boldsymbol{x})$, the basis function $\phi_{\boldsymbol{c}^{(i)}}(\boldsymbol{x})$ will be revised:

$$\phi_{\boldsymbol{c}^{(i)}}(\boldsymbol{x}) = \exp\left(-\frac{1}{l^2}\left(\boldsymbol{x}-\boldsymbol{c}^{(i)}\right)^{\top}\Lambda^{-1}\left(\boldsymbol{x}-\boldsymbol{c}^{(i)}\right)\right). \tag{6.19}$$

The NSDC kernel is derived accordingly:

$$\begin{aligned} k(\boldsymbol{x},\boldsymbol{x}^*) =& \frac{\sigma_0^2}{n}\sum_{i=1}^{n}\left[\exp\left(-\frac{1}{\delta^2}\boldsymbol{d}^{\top}\Lambda^{-1}\boldsymbol{d}\right)\right. \\ & \left.\exp\left(-\frac{4}{\delta^2}\left(\boldsymbol{c}^{(i)}-\frac{\boldsymbol{x}+\boldsymbol{x}^*}{2}\right)^{\top}\Lambda^{-1}\left(\boldsymbol{c}^{(i)}-\frac{\boldsymbol{x}+\boldsymbol{x}^*}{2}\right)\right)\right]. \end{aligned} \tag{6.20}$$

where $\delta^2 = 2l^2$. In order to take the covariance structure between variables into consideration, the Mahalanobis distance is used in the kernel function by inserting the covariance matrix Λ. Λ can be estimated using the sample covariance of the training set. The kernel width δ is the only parameter to be specified.

Kernel widths for operating modes

When a-priori information about data clusters with respect to operating modes and transition periods is available, it is possible to assign an individual covariance matrix Λ_p for the p-th cluster C_p with n_p samples in order to represent each operating mode. The basis

function for the p-th cluster is:

$$\phi_{\boldsymbol{c}_p^{(i)}}(\boldsymbol{x}) = \exp\left(-\frac{1}{l^2}\left(\boldsymbol{x}-\boldsymbol{c}_p^{(i)}\right)^\top \Lambda_p^{-1}\left(\boldsymbol{x}-\boldsymbol{c}_p^{(i)}\right)\right) \tag{6.21}$$

where $\boldsymbol{c}_p^{(i)} \in C_p$. The kernel function can be constructed accordingly:

$$\begin{aligned} k(\boldsymbol{x},\boldsymbol{x}^*) =& \frac{\sigma_0^2}{n}\sum_{p=1}^{P}\left[\exp\left(-\frac{1}{\delta^2}\boldsymbol{d}^\top\Lambda_p^{-1}\boldsymbol{d}\right)\right. \\ & \left.\sum_{i=1}^{n_p}\exp\left(-\frac{4}{\delta^2}\left(\boldsymbol{c}_p^{(i)}-\frac{\boldsymbol{x}+\boldsymbol{x}^*}{2}\right)^\top\Lambda_p^{-1}\left(\boldsymbol{c}_p^{(i)}-\frac{\boldsymbol{x}+\boldsymbol{x}^*}{2}\right)\right)\right]. \end{aligned} \tag{6.22}$$

where $\Lambda_p = \mathrm{cov}(\boldsymbol{x})$ such that $\boldsymbol{x} \in C_p$ can be estimated by the sample covariance of the p-th data cluster and δ. By using Λ_p, the covariance matrix of the p-th operating mode, the NSDC kernel uses the Mahalanobis distance that takes the localized covariance structure into account. When the data clustering information is not available, the NSDC kernel can still be implemented by assuming $P = 1$, yielding Eqn (6.20).

To summarize, the new NSDC kernel adopts the sample covariance matrices of each data cluster in its formulation, if they are known. In this formulation, the scaling δ regulates the overall behaviour of the NSDC kernel while the varying sample covariances capture the non-stationary covariance structure. Consequently, compared to the RBF kernel, the NSDC kernel can handle the non-stationary behaviour caused by multiple operating modes without introducing additional parameters. Even when P is set to be one, the NSDC kernel can still handle the non-stationary behaviour to a certain extend due to the localized term using the training samples, as shown in Eqn (6.13).

6.4.3 Tuning the kernel width δ

This section analyses the behaviour of the NSDC kernel with respect to the kernel width δ and proposes a strategy for tuning δ.

Asymptotic analysis of the NSDC kernel

Asymptotic analysis of the NSDC kernel investigates the extreme cases of large δ values and small δ values. The analysis also investigates the extreme case when a test sample is

very far away from the training data. The aim is to show that inappropriate kernel widths will impact the performance of the NSDC kernel.

In Eqn (6.22), when the kernel width δ is excessively large relative to the scale of the data, the p, q-th entry of the kernel matrix K obtained by the NSDC kernel of two sample $\boldsymbol{x}_p$ and $\boldsymbol{x}_q$ is calculated by expanding the exponential functions and omitting higher order terms. For simplicity, the covariance matrix $\hat{\Sigma}_j^{-1}$ is assumed to be an identity matrix because the process data are often normalized.

$$\begin{aligned} K_{p,q} = \lim_{\delta\to\infty} \sigma_w^2 \sum_{j=1}^{J} \Bigg[& \exp\left(-\frac{1}{\delta^2}(\boldsymbol{x}_p - \boldsymbol{x}_q)^\top (\boldsymbol{x}_p - \boldsymbol{x}_q)\right) \\ & \times \sum_{i=1}^{n_j} \exp\left(-\frac{4}{\delta^2}\left(\boldsymbol{c}^{(i)} - \frac{\boldsymbol{x}_p + \boldsymbol{x}_q}{2}\right)^\top \left(\boldsymbol{c}^{(i)} - \frac{\boldsymbol{x}_p + \boldsymbol{x}_q}{2}\right)\right)\Bigg] \\ & \approx N - \mu_2 - \frac{1}{\delta^2}\left[(2J - \mu_2)(\boldsymbol{x}_p^\top \boldsymbol{x}_p + \boldsymbol{x}_q^\top \boldsymbol{x}_q) - 2\mu_2 \boldsymbol{x}_1^\top \boldsymbol{x}_2\right] \end{aligned} \tag{6.23}$$

where $\mu_2 = \sum_{j=1}^{J} \sum_{i=1}^{n_j} \boldsymbol{c}^{(i)\top} \boldsymbol{c}^{(i)}$. Given that $\sum_{p=1}^{n} \boldsymbol{x}_p = 0$, $\bar{K}_{p,\text{row}}$, the mean of the p-th row, and $\bar{K}$, the mean of all entries in K, are calculated respectively:

$$\bar{K}_{p,\text{row}} = N - \mu_2 - \frac{1}{\delta^2}\left[(2J - \mu_2)\left(\boldsymbol{x}_p^\top \boldsymbol{x}_p + \frac{1}{n}\sum_{l=1}^{n} \boldsymbol{x}_p^\top \boldsymbol{x}_l\right) - \frac{2\mu_2}{n}\sum_{l=1}^{n} \boldsymbol{x}_p^\top \boldsymbol{x}_l\right] \tag{6.24}$$

which also holds for the p-th column since K is symmetric.

$$\bar{K} = N - \mu_2 - \frac{1}{\delta^2}\left[(2J - \mu_2)\left(\frac{2}{n}\sum_{l=1}^{n} \boldsymbol{x}_l^\top \boldsymbol{x}_l - \frac{2\mu_2}{n^2}\sum_{l=1}^{n}\sum_{h\neq l}^{n} \boldsymbol{x}_l^\top \boldsymbol{x}_h\right)\right]. \tag{6.25}$$

Therefore the p, q-th entry of the centered kernel matrix $\tilde{K}$ is

$$\begin{aligned} \tilde{K}_{p,q} &= K_{p,q} - \bar{K}_{p,\text{row}} - \bar{K}_{q,\text{col}} + \bar{K} \\ &= \frac{2\mu_2}{\delta^2}\left(\boldsymbol{x}_p^\top \boldsymbol{x}_q - \frac{1}{n}\sum_{l=1}^{n} \boldsymbol{x}_l^\top \boldsymbol{x}_l - \frac{1}{n}\sum_{h=1}^{n} \boldsymbol{x}_h^\top \boldsymbol{x}_h + \frac{1}{n^2}\sum_{l=1}^{n}\sum_{h\neq l}^{n} \boldsymbol{x}_l^\top \boldsymbol{x}_h\right) \end{aligned} \tag{6.26}$$

Given μ_2 and δ are constant, Eqn (6.26) is proportional to the entry of the centered kernel matrix obtained by the linear kernel $k_{\text{linear}}(\boldsymbol{x}_p, \boldsymbol{x}_q) = \boldsymbol{x}_p^\top \boldsymbol{x}_q$. Similar derivations hold for the test samples. Hence the NSDC kernel with excessively large kernel widths will reduce to a linear kernel, losing its ability to build a nonlinear monitoring model for multimode and nonlinear data.

On the other hand, the NSDC kernel will result in an over-fitted model when δ is excessively small. The kernel function becomes

$$\begin{aligned} K_{p,q} &= \lim_{\delta \to 0} \sigma_w^2 \sum_{j=1}^{J} \left[\exp\left(-\frac{1}{\delta^2} (\boldsymbol{x}_p - \boldsymbol{x}_q)^\top (\boldsymbol{x}_p - \boldsymbol{x}_q) \right) \right. \\ &\quad \left. \sum_{i=1}^{n_j} \exp\left(-\frac{4}{\delta^2} \left(\boldsymbol{c}^{(i)} - \frac{\boldsymbol{x}_p + \boldsymbol{x}_q}{2} \right)^\top \left(\boldsymbol{c}^{(i)} - \frac{\boldsymbol{x}_p + \boldsymbol{x}_q}{2} \right) \right) \right] \\ &= \begin{cases} 1, & p = q, \\ 0, & p \neq q. \end{cases} \end{aligned} \tag{6.27}$$

For a test sample $\boldsymbol{x}_{\text{test}}$,

$$\begin{aligned} K_{p,\text{test}} &= \lim_{\delta \to 0} \sigma_w^2 \sum_{j=1}^{J} \left[\exp\left(-\frac{1}{\delta^2} (\boldsymbol{x}_p - \boldsymbol{x}_{\text{test}})^\top (\boldsymbol{x}_p - \boldsymbol{x}_{\text{test}}) \right) \right. \\ &\quad \left. \sum_{i=1}^{n_j} \exp\left(-\frac{1}{2\delta^2} \left(\boldsymbol{c}^{(i)} - \frac{\boldsymbol{x}_p + \boldsymbol{x}_{\text{test}}}{2} \right)^\top \left(\boldsymbol{c}^{(i)} - \frac{\boldsymbol{x}_p + \boldsymbol{x}_{\text{test}}}{2} \right) \right) \right] \\ &= \begin{cases} K_{p,p}, & \boldsymbol{x}_{\text{test}} = \boldsymbol{x}_p, \\ 0, & \boldsymbol{x}_{\text{test}} \neq \boldsymbol{x}_p. \end{cases} \end{aligned} \tag{6.28}$$

Therefore, the monitoring model is extremely over-fitted to the training data because it will always label a test sample as anomalous if this sample does not belong to the training data.

Tuning strategy

This chapter adopts the tuning strategy proposed in Chapter 5 for RBF kernels. The strategy examines several values of the kernel width δ to train the monitoring model. The false alarm rates are calculated by applying the trained monitoring models to a cross-validation dataset generated using the model presented at the beginning of Section 6.2. Fig. 6.5 plots the false alarm rates achieved for various δ values. The optimal δ value is considered as 5.4 because $\delta = 5.4$ can achieve a reasonably low false alarm rate that is close to 1%.

Fig. 6.6 compares the monitoring models built by NSDC-KPCA with various values of kernel width by visualizing the detection contours. The detection contours for the numerical example are achieved by connecting the samples in the variable space with SPE values equal to SPE_{UCL}. Any sample located outside the detection contour will be detected as an

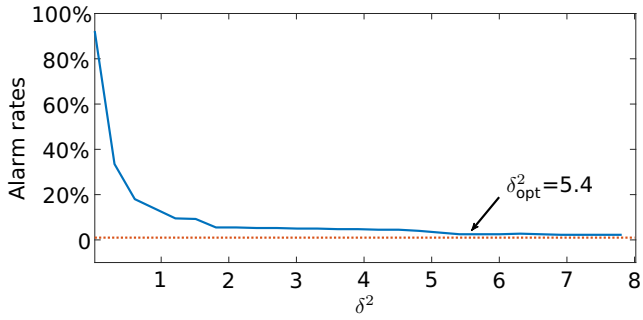
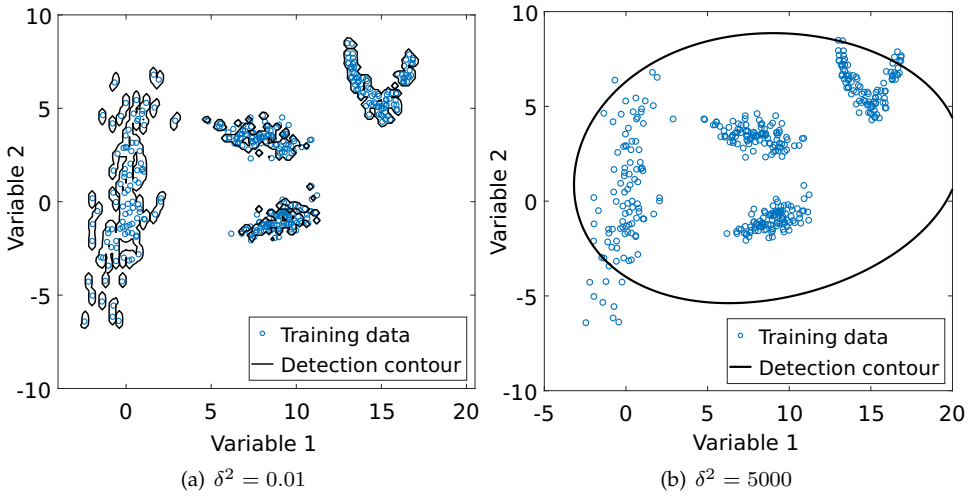
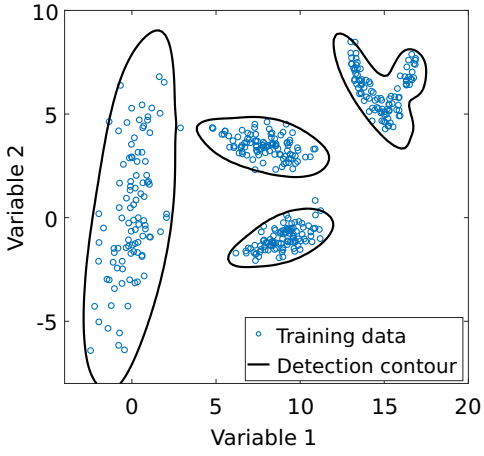


Figure 6.5: False alarm rates and δ tuning for NSDC-KPCA



(a) $\delta^2 = 0.01$ (b) $\delta^2 = 5000$



(c) $\delta^2 = 5.4$

Figure 6.6: Performance of NSDC-KPCA with various δ^2 values

anomaly. The contour achieved when $\delta = 0.01$ in Fig. 6.6(a) is over-fitted and will lead to increased false alarms. The contour achieved when $\delta = 1000$ in Fig. 6.6(b) is under-fitted and cannot describe the dataset sufficiently, leading to increased missed alarms. The optimal kernel width $\delta = 5.4$ results in the contour in Fig. 6.6(c), which suits the multimode data well.

To conclude, when using NSDC-KPCA for building monitoring models, excessively large kernel widths may lead to under-fitted models which are incapable of describing the data nonlinearity. The model data mismatch will impact the monitoring performance and will result in increased missed alarms. Excessively small kernel widths may lead to over-fitted models where small variations in the healthy data are not tolerated and will lead to more false alarms. In order to improve the accuracy of monitoring models, one should apply the tuning strategy introduced in this section to properly tune the kernel width δ for NSDC kernels.

6.5 Comparison with other methods

This section compares the performance in process monitoring of NSDC-KPCA against the performance of RBF-KPCA and other methods proposed for multimodal process monitoring. Several examples with numerical simulations are used for the comparison.

6.5.1 NSDC kernel and RBF kernel

Detection contours

This section first presents the detection contours generated by NSDC-KPCA for the illustrative examples in Section 6.2. The first r PCs with 99% accumulated variability are chosen as the kernel features $\boldsymbol{z}$. r is selected such that:

$$\frac{\sum_{i=1}^{r}\lambda_i}{\sum_{i=1}^{q}\lambda_i} \geq 99\% \tag{6.29}$$

where λ_i is the i-th element of q eigenvalues corresponding to $\boldsymbol{z}$ in descending order. The monitoring statistic is SPE and various values of the kernel widths are used. The confidence level of the SPE is set as 99%.

Fig. 6.7 shows the results for the illustrative example. It can be seen that the contours generated by NSDC-KPCA can acknowledge S_1 as a healthy sample and can detect S_2 as a faulty sample. By comparing Fig. 6.3 and Fig. 6.7, it is possible to conclude that NSDC-KPCA gives better descriptions of the multimodal training dataset and will significantly improve the fault detection performance. It may be observed from Fig. 6.3 that for higher values

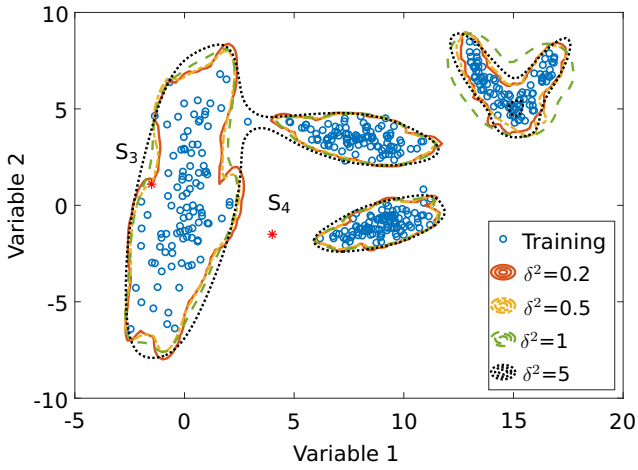
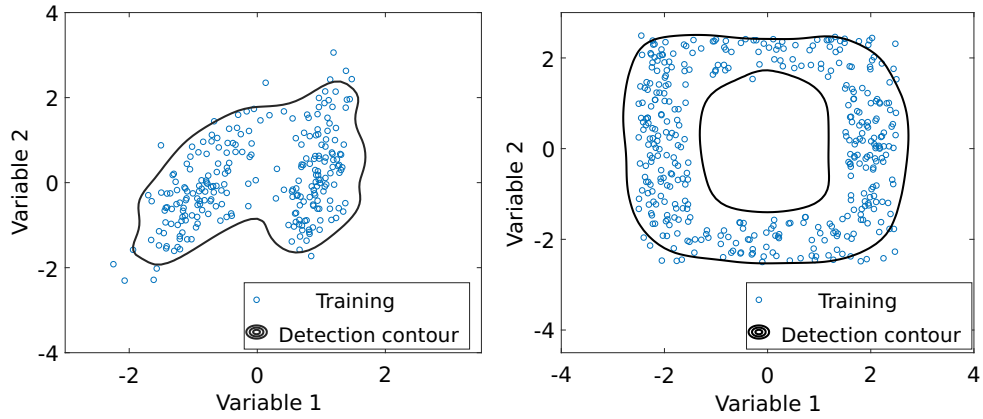


Figure 6.7: Detection contours by NSDC-KPCA for the illustrative example

of δ the control limit of RBF-KPCA may be excessively relaxed, leading to missed alarms, as samples such as S_2 are not detected as faulty. On the other hand, the RBF kernel suffers from over-fitting issues when δ reduces resulting in more false alarms such as sample S_1. In contrast, the NSDC kernel is capable of generating satisfactory detection contours for all operating modes (Fig. 6.7).

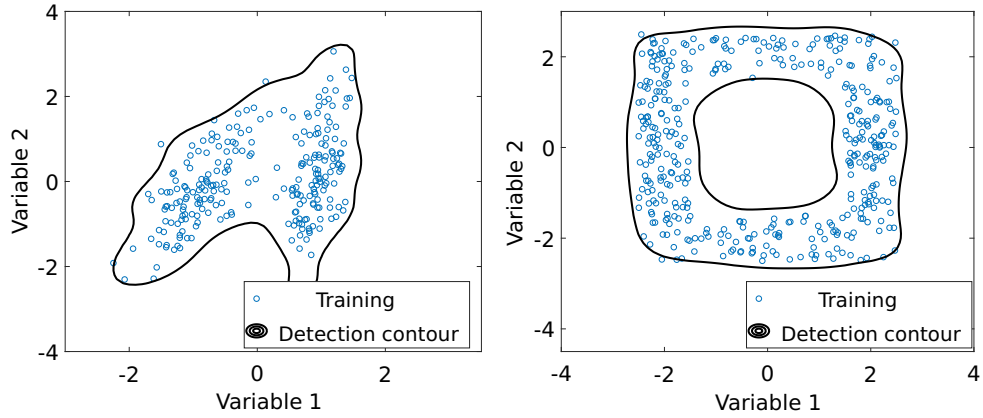
Fig. 6.8 shows the monitoring contours generated by the NSDC kernel for another two nonlinear examples. Fig. 6.9 shows the corresponding contours achieved by RBF-KPCA. The data clusters are not explicit in these nonlinear examples. RBF-KPCA can achieve good contours because the RBF kernel can cope with nonlinearity in the data. For NSDC-KPCA, the data are not clustered in advance. By setting the cluster number $P = 1$, one can implement the NSDC kernel defined by Eqn (6.20) and obtain proper monitoring contours. The contours also demonstrate that, when the data clusters are not explicit, the NSDC kernel can cope with other types of nonlinearity without considering the varying covariance structures of each data cluster.

The results in this section indicate that the NSDC kernel will yield a KPCA model which generates a better control limit than the RBF kernel for fault detection of multimodal data. The NSDC kernel also suffers less from the issues associated with over or under-fitting. Even when there is no data clustering information available, the performance of the NSDC kernel will not be significantly compromised. It also indicates that the NSDC kernel can handle other types of data nonlinearity in addition to the multiple operating modes.



(a) Nonlinear example 1: NSDC-KPCA (b) Nonlinear example 2: NSDC-KPCA

Figure 6.8: Detection contours obtained by NSDC-KPCA for other nonlinear examples



(a) Nonlinear example 1: RBF-KPCA (b) Nonlinear example 2: RBF-KPCA

Figure 6.9: Detection contours obtained by RBF-KPCA for other nonlinear examples

Fault detection performance

This section further tests the ability of fault detection of the proposed NSDC kernel. In particular, the test considers the behaviour of the monitoring statistic SPE achieved by NSDC-KPCA.

Two test sequences, which represent how the process variables deviate from the healthy operating modes, are used to test the fault detection performance in the illustrative example presented in Section 6.2. Fig. 6.10 plots the training data and the two test sequences. Test sequence 1 (marked with yellow "+" in Fig. 6.10) has data samples moving from Mode 1 to Mode 4. Noise, in the form of Gaussian white noise, was added to the data. Test sequence 2 (marked with purple "$\times$" in Fig. 6.10) has data samples moving from Mode 1 via Mode 2

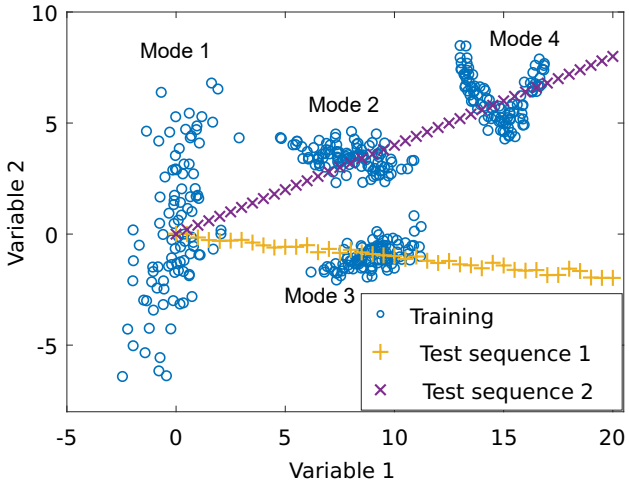
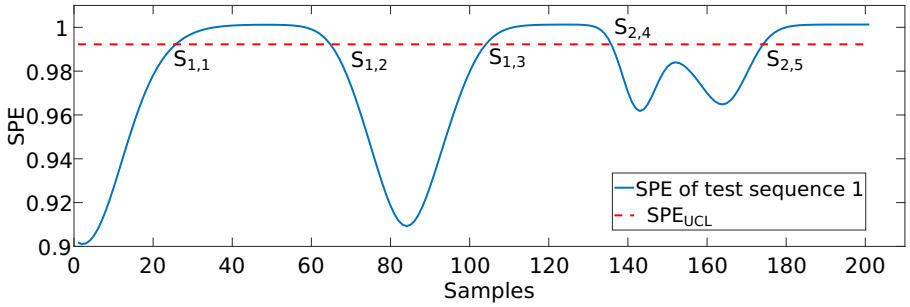


Figure 6.10: Scatter plots of the training data and test fault sequences in illustrative example

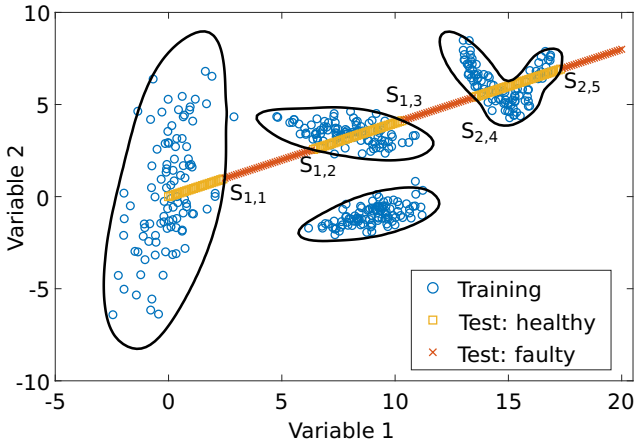
to Mode 3, which is a nonlinear mode. For clarity, every fifth sample of Test sequences 1-2 is plotted in Fig. 6.10.

The fault detection performance of the proposed approach for the two test sequences is shown in Figs 6.11 and 6.12. The kernel width δ used for NSDC-KPCA is 5 and the confidence level of SPE_{UCL} is 1%. In order to relate the results in the trend plots of SPE to the scatter plots of the data, several samples are marked. In Fig. 6.11(a), the samples between $\text{S}_{1,1}$ and $\text{S}_{1,2}$ do not belong to any known modes, and the SPE exceeds its control limit in Fig. 6.11(b), resulting in these samples being identified as faulty. Similar behaviour is observed for the samples after $\text{S}_{1,3}$. On the other hand, it can been seen that the rest of the samples in the test sequence, e.g. the samples between $\text{S}_{1,2}$ and $\text{S}_{1,3}$, belong to known modes and that the associated SPE does not exceed its control limit. For Test sequence 2, the samples between $\text{S}_{2,1}$ and $\text{S}_{2,2}$, the samples between $\text{S}_{2,3}$ and $\text{S}_{2,4}$, and the samples after $\text{S}_{2,5}$ are identified as faulty, as shown in Fig. 6.12(b). It can also be observed that the control limit SPE_{UCL} is the same in different modes.

For comparison, Fig. 6.13 visualizes the results obtained by RBF-KPCA for Test sequence 2. Fig. 6.13(a) shows that a false alarm will be triggered starting from the sample $\text{S}'_{2,1}$ when the kernel width is small ($\delta^2 = 0.2$). Fig. 6.13(b) shows that missed alarms appear between the sample $\text{S}'_{2,2}$ and the sample $\text{S}'_{2,3}$ when the kernel width slightly increases ($\delta^2 = 0.7$). This indicates that RBF-KPCA will not be able to achieve a small number of false alarms and a small number missed alarms simultaneously.



(a) Trend plot of SPE



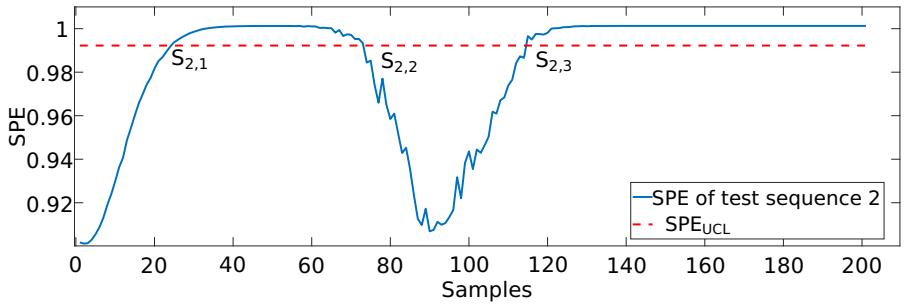
(b) Scatter plot of training and testing data

Figure 6.11: NSDC-KPCA performance for the simulated example: Test sequence 1

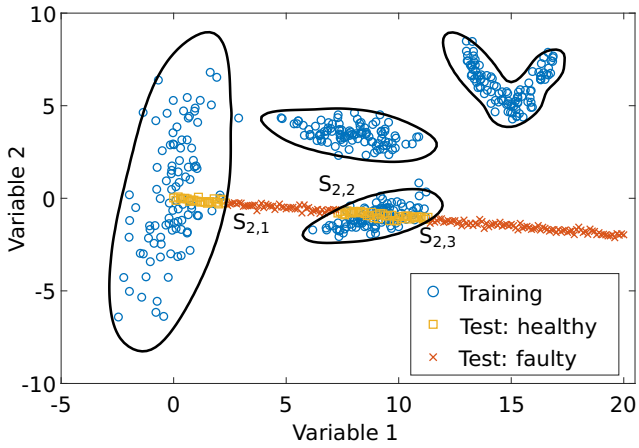
When compared with RBF-KPCA, NSDC-KPCA can build better monitoring models and can generate better monitoring contours for multimodal data. The results also demonstrate that the magnitude of the SPE obtained using NSDC-KPCA is sensitive to the mode changes and the same control limit can be used for anomaly detection in several operating modes. In addition, the SPE increases monotonically as the test sample deviates from the known modes. This behaviour may also be useful for estimating the development of a fault.

6.5.2 NSDC kernel and other methods in literature

Comparisons with other methods proposed for multimodal process monitoring are challenging because it is difficult to implement and tune each method in a rigorous manner that ensures a fair comparison. Instead, it is better to apply NSDC-KPCA on datasets that have been used by other authors to demonstrate their monitoring methods. One can assume that the authors have fully optimized their own methods. In order to provide a fair evaluation of the performance of the NSDC kernel relative to existing methods, this section

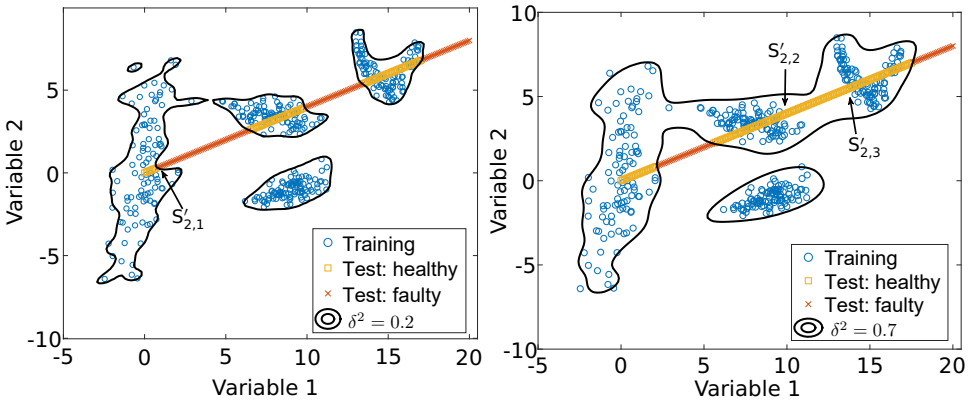


(a) Trend plot of SPE



(b) Scatter plot of data

Figure 6.12: NSDC-KPCA performance for the simulated example: Test sequence 1



(a) $\delta^2 = 0.2$ (b) $\delta^2 = 0.7$

Figure 6.13: RBF-KPCA performance for the simulated example: Test sequence 2

compares the NSDC-KPCA method with the results reported in recently published papers Deng et al. (2017), Li et al. (2017) and Yu et al. (2018), which adapted kernel-based methods for multimodal process monitoring. The simulated datasets described in each of these papers form the basis of the comparison. The False Alarm Rate (FAR) defined in Eqn (6.30) and the Missed Alarm Rate (MAR) defined in Eqn (6.31) are used to evaluate the anomaly

Table 6.1: Performance comparison of NSDC-KPCA with methods in literature

Tested examples	Confidence level (%)	FAR (%)			MAR (%)		
		NSDC	RBF	Other	NSDC	RBF	Other
Example 1[1]	5	4.13	4.85	5	8.08	11.58	14.2
Example 2[2]	5	3.7	4.42	4	1.15	5.68	18.2
Example 3[3]	1	0.83	1	0.73	0.27	0.3	0.46

[1] Fault D2 from Section V.A in Deng et al. (2017)
[2] Case 4 from Section 4.1 in Li et al. (2017)
[3] Case 2 from Section 5.1 in Yu et al. (2018)

detection performance

$$\text{FAR} = \frac{n_{\text{FA}}}{n_{\text{ft}}} \tag{6.30}$$

$$\text{MAR} = \frac{n_{\text{MA}}}{n_{\text{ft}}} \tag{6.31}$$

where n_{MA} denotes the number of anomalous samples not being detected as anomalies and n_{ft} denotes the total number of faulty samples. The MAR is the rate of missed detections in a test set that includes faulty samples. The MAR should ideally be zero.

The kernel widths of the RBF and the NSDC kernels and the control limits of SPE are tuned using the strategy proposed in Section 6.4.3 according to the confidence levels. The confidence levels of monitoring statistics are used as the expected FARs for both the cross-validation and the test sets. Therefore the FAR obtained by NSDC and RBF kernels for the test sets should be similar to the confidence levels if the monitoring model can describe the normal data well. The column headed as *Other* shows the results reported in Deng et al. (2017), Li et al. (2017), and Yu et al. (2018) for the respective multimodal process monitoring methods described in each paper. It can be seen that the NSDC-kernel PCA approach achieves lower MARs than both the RBF-KPCA and the methods presented in Deng et al. (2017), Li et al. (2017) and Yu et al. (2018). For Example 2 in Table 6.1, the NSDC also achieves a smaller FAR than the target value of 5% because of the artificial outliers placed into the training set by Li et al. (2017).

6.5.3 Qualitative comparison

Previous examples have shown that the new NSDC kernel can achieve good models for monitoring multimodal processes. Table 6.2 discusses the qualitative comparison of the NSDC kernel and the RBF kernel. The main advantage of the NSDC kernel over the RBF

Table 6.2: Performance comparison of NSDC kernel and RBF kernel

	NSDC	RBF
Convolution formulation?	✓	✓
Number of basis functions	Finite	Infinite
Centers of basis functions	Training samples	$(-\infty, \infty)$
Parameters	Kernel width δ	Kernel width δ
Non-stationary?	✓	×
Optimization-free?	✓	✓
Prior knowledge needed?	✓	×

kernel is that, due to the new assumptions in the convolution kernel formulation, the NSDC kernel is non-stationary and data-dependent. Hence the NSDC kernel can handle the non-stationary covariance caused by multimodality.

Recent works on multivariate approaches for multimodal process monitoring deal with the multimodality in an ad-hoc way, including the locally-weighted approach (Song and Shi, 2018; Deng and Wang, 2018; Li et al., 2017) and using the local statistics matrix (Deng et al., 2017) or the residuals obtained by kernel regression (Yu et al., 2018) instead of original measurements. For those methods using data-dependent kernels, Chen et al. (2008) suggested that the parameters of these kernels need to be optimized. This chapter proposes a new and systematic way of formulating the non-stationary kernel function via convolution and derives the closed-form solution, i.e. the NSDC kernel. Since the NSDC kernel only requires one parameter, i.e. the kernel width, to be tuned, it also avoids additional parametrization of the kernel function and makes the training procedure and on-line monitoring easier.

6.6 An on-line framework for process monitoring

A framework for on-line process monitoring using the proposed NSDC kernel is introduced in this section. The framework includes off-line training of monitoring models and on-line updating of the models. The section first discusses an unsupervised clustering method which can prepare the off-line training data such that NSDC-KPCA can train the monitoring model using these data, followed by how the new data from operation can be incorporated to update the monitoring model.

6.6.1 Data clustering and on-line updating

The NSDC-KPCA approach requires the training data to be clustered a priori to find the operating modes. When clustering the historical data used for training, the number of operating modes existing in the data may not be known. Data-driven clustering methods such as K-means (Lloyd, 1982) and expectation maximization method (Dempster et al., 1977), typically need to know the number of clusters a-priori (Xu and Wunsch, 2005). In contrast, Teh (2011) presented the Dirichlet Process (DP), which is a non-parametric Bayesian method, to automatically determine the number of clusters when clustering the data. DP organizes the data so that they could be used in NSDC-KPCA. The formulation of DP-based clustering was devised in collaboration with Tian Cong from AGH University, and executed using clustering codes written by Tian.

Another expectation of monitoring methods given by Section 2.4 is to account for new healthy operating modes and to detect faults in new modes. Such an expectation requires the on-line update of the monitoring models using healthy data from the new healthy modes. Therefore, an on-line monitoring framework is proposed in this chapter. The off-line training step of the proposed framework assumes that all training data are from healthy operating modes. This step applies DP to automatically determine the number of modes in the training data and to label the training samples accordingly. The step is semi-supervised such that it does not require the information about which mode each data sample belongs to whilst requiring all the training data to be labelled as healthy data. The training step then employs NSDC-KPCA in order to reduce false and missed alarms. Unlike the multiple model approaches such as Zhao et al. (2006), Zhu et al. (2012) and Song et al. (2016), the NSDC-KPCA approach does not require the classification of a test sample or the fusion of the monitoring results. Moreover, the magnitude of a monitoring statistic achieved by NSDC-KPCA reflects the level of anomalies in the data with respect to the known healthy operating modes regardless of which mode the process is running in. In the on-line update step, new data collected from new healthy operating modes can be incorporated via the DP-NSDC-KPCA approach allowing the monitoring model to be updated on-line.

6.6.2 Data clustering using Dirichlet process Gaussian mixture models

Section 6.4.2 presented the NSDC kernel parametrized with the covariance matrices of the data from each mode. Therefore, in order to build monitoring models it is necessary to cluster the training data. This section discusses a clustering method which can be applied to the training data.

The Dirichlet Process Gaussian Mixture Model (DP-GMM) is an unsupervised clustering algorithm that does not require prior knowledge of the number of clusters. It uses Gaussian Mixture Models (GMMs) (Reynolds, 2015) to describe data and the Dirichlet Process (DP) for generating clusters.

Gaussian Mixture Models

In a GMM, an m-dimensional random variable $\boldsymbol{x}$ follows a Gaussian mixture model with J components:

$$\begin{aligned} &\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}_j, \Sigma_j) \quad \text{with probability } \pi_j \\ &\text{s.t.} \quad \pi_j > 0 \,\forall\, j,\ \sum_{j=1}^{J} \pi_j = 1 \end{aligned} \tag{6.32}$$

where $\sim$ denotes that the random variable on the left side follows the probability distribution on the right hand side. $\boldsymbol{\mu}_j, \Sigma_j$ are the mean vector and the covariance matrix of the j-th Gaussian component, respectively. The mixture proportion π_j is the probability of drawing from the j-th component.

DP-GMMs

In DP-GMMs, the sample $\boldsymbol{x}$ is drawn from a GMM. The Gaussian components in this mixture have parameters (e.g. $\boldsymbol{\mu}_j$, Σ_j and π_j) and these parameters follow the distributions generated by the Dirichlet Process in a Bayesian way. According to the conclusions in Escobar (1988) and Escobar (1994), when the vector of mixture proportions $\boldsymbol{\pi} = \{\pi_1, \cdots, \pi_J\}$ follows the Dirichlet distribution (Forbes et al., 2011), a discrete distribution $G(\cdot)$ of Gaussian means and variances is generated from a Dirichlet Process specified by the base distribution G_0 and the concentration parameter α. Hence, a Dirichlet mixture of Gaussian distributions can be written as:

$$\begin{aligned} G(\cdot) &\sim DP(\alpha, G_0) \\ \boldsymbol{\mu}_j, \Sigma_j &\sim G(\cdot) \quad \text{for } j = 1, \cdots, J \\ \boldsymbol{\pi} &\sim Dir(\underbrace{\frac{\alpha}{J}, \cdots, \frac{\alpha}{J}}_{J}) \end{aligned} \tag{6.33}$$

where $\boldsymbol{\mu}_j$ and Σ_j are mean and covariance of the j-th component, J is the number of components in this mixture, and DP and Dir respectively stand for the Dirichlet Process and the Dirichlet Distribution. In this thesis, the concentration parameter α is kept constant.

Given the prior distributions of these parameters, the key Bayesian step is to obtain the posterior distributions using data. The prior distributions placed over $\boldsymbol{\mu}_j$ and Σ_j are taken as a Gaussian distribution and an Inverse Wishart distribution, respectively:

$$\begin{aligned} \Sigma_j &\sim IW(\nu_0, \Lambda_0) \\ \boldsymbol{\mu}_j | \Sigma_j &\sim \mathcal{N}\left(\boldsymbol{u}_0, \frac{\Sigma_j}{\kappa_0}\right) \end{aligned} \tag{6.34}$$

where ν_0 and κ_0 are positive values. Eqn. (6.34) leads to the joint distribution of $\boldsymbol{\mu}_j$ and Σ_j being a Normal Inverse Wishart (NIW) distribution parametrized with $\Theta_0 = \{\boldsymbol{u}_0, \kappa_0, \nu_0, \Lambda_0\}$ (Murphy, 2007):

$$(\boldsymbol{\mu}_j, \Sigma_j) \sim NIW(\Theta_0). \tag{6.35}$$

This NIW distribution is considered as the base distribution G_0 in $DP(\alpha, G_0)$. The parameters Φ for $G(\cdot)$ will be updated using samples of $\boldsymbol{x}$.

Inference procedure for DP-GMMs

When a set of samples $X = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n\} \in \mathbb{R}^{m \times n}$ is available, the inference step employs a vector $\boldsymbol{I}$, where in this chapter $\boldsymbol{I} = \{I_1, \cdots, I_n\} \in \mathbb{R}^{n \times 1}$ is the cluster indices. For the i-th sample $\boldsymbol{x}_i$, I_i takes a value between 1 and J, indicating the Gaussian component which $\boldsymbol{x}_i$ belongs to. Therefore, the parameters for the DP-GMM are $\boldsymbol{\mu}_j, \Sigma_j$ and $\boldsymbol{I}$ along with the hyperparameter Θ for the NIW distribution. These parameters are to be inferred from the data. Görür and Rasmussen (2010) proposed the following updating steps based on Gibbs Sampling:

a. Updating the parameters of each Gaussian component $(\boldsymbol{\mu}_j, \Sigma_j)$ for $j = 1, \cdots, J$;

b. Updating the vector $\boldsymbol{I} \in \mathbb{R}^n$;

c. Updating the hyperparameter Θ of the NIW distribution.

When initiating the parameters, the parameters Θ_0 and $\boldsymbol{I}_0$ are assigned and $\boldsymbol{\mu}_j, \Sigma_j$ are sampled from G_0 to generate an initial group of clusters.

The updating of $\boldsymbol{\mu}_j$ and Σ_j can be achieved by sampling $\boldsymbol{\mu}_j$ and Σ_j from the updated posterior NIW distribution $NIW(\Theta^{(j)})$.

The updating of $\boldsymbol{I}$ is equivalent to clustering the samples. The cluster index I^* is sampled given the existing clusters by assigning a sample $\boldsymbol{x}^*$ to a cluster with respect to the current clustering result of all other samples. In DP, countably infinite clusters are assumed and the probability of a sample $\boldsymbol{x}^*$ being assigned to one of the known clusters is:

$$\begin{aligned} &p(I^* = j|\boldsymbol{I}^{'}, \boldsymbol{\mu}_1, \Sigma_1, \cdots, \boldsymbol{\mu}_J, \Sigma_J, \alpha) \\ &\qquad = \frac{n_j p(\boldsymbol{x}^*|\boldsymbol{\mu}_j, \Sigma_j)}{\alpha G_0(\boldsymbol{x}^*) + \sum_{j=1}^{J} n_j p(\boldsymbol{x}^*|\boldsymbol{\mu}_j, \Sigma_j)}. \end{aligned} \tag{6.36}$$

A new cluster is created for $\boldsymbol{x}^*$ with the probability:

$$\begin{aligned} &p(I^* \notin \{1, \cdots, J\}|\boldsymbol{I}^{'}, \Theta_0, \alpha) \\ &\qquad = \frac{\alpha G_0(\boldsymbol{x}^*)}{\alpha G_0(\boldsymbol{x}^*) + \sum_{j=1}^{J} n_j p(\boldsymbol{x}^*|\boldsymbol{\mu}_j, \Sigma_j)} \end{aligned} \tag{6.37}$$

where $\boldsymbol{I}^{'}$ is the vector $\boldsymbol{I}$ excluding the cluster index of $\boldsymbol{x}^*$, n_j is the number of samples in the j-th cluster. Görür and Rasmussen (2010) showed that $G_0(\boldsymbol{x}^*)$ is the marginal likelihood of $\boldsymbol{x}^*$ that follows a multivariate Gaussian distribution of which the mean and covariance are a sample drawn from $NIW(\Theta_0)$. The number of clusters J will increase when a new cluster is created and will reduce if a cluster is empty and removed after updating.

The updating of Φ requires the subset X_j with n_j samples $\{\boldsymbol{x}_1^{(j)}, \boldsymbol{x}_2^{(j)}, \cdots, \boldsymbol{x}_{n_j}^{(j)}\}$ assigned to the j-th cluster. Hence, the posterior distribution of $(\boldsymbol{\mu}_j, \Sigma_j)$ is derived based on Bayes' Theorem:

$$\begin{aligned} p(\boldsymbol{\mu}_j, \Sigma_j|X_j, \Theta_0) &\propto p(X|\boldsymbol{\mu}_j, \Sigma_j) p(\boldsymbol{\mu}_j, \Sigma_j|\Theta_0) \\ &= p(\boldsymbol{\mu}_j, \Sigma_j|\Theta^{(j)}) \end{aligned} \tag{6.38}$$

where p denotes the probability function and Θ_0 denotes the parameters for the prior NIW distribution $\boldsymbol{u}_0, \kappa_0, \nu_0$ and Λ_0 and $\Theta^{(j)}$ for the posterior distribution, i.e. $\boldsymbol{u}^{(j)}, \kappa^{(j)}, \nu^{(j)}$ and $\Lambda^{(j)}$. The NIW distribution is a conjugate prior for the multivariate Gaussian distribution with unknown mean and variance. In the Bayesian formulation, when the likelihood has a multivariate Gaussian distribution with unknown mean and variance, the conjugate prior distribution guarantees that the posterior distribution is also a multivariate Gaussian distribution. Therefore $\Theta^{(j)}$ can be obtained by comparing the posterior distribution in Eqn. (6.38) with the standard NIW formulation. $\Theta^{(j)}$ is calculated by Eqn. (6.39) using Θ_0 and X_j:

$$\begin{aligned}
\boldsymbol{u}^{(j)} &= \frac{\kappa_0}{\kappa_0 + n_j}\boldsymbol{u}_0 + \frac{n_j}{\kappa_0 + n_j}\bar{X}_j \\
\kappa^{(j)} &= \kappa_0 + n_j \\
\nu^{(j)} &= \nu_0 + n_j \\
\Lambda^{(j)} &= \Lambda_0 + \sum_{i=1}^{n_j}(\boldsymbol{x}_i^{(j)} - \bar{X}_j)(\boldsymbol{x}_i^{(j)} - \bar{X}_j)^T + \\
&\quad \frac{\kappa_0 n_j}{\kappa_0 + n_j}(\bar{X}_j - \boldsymbol{u}_0)(\bar{X}_j - \boldsymbol{u}_0)^T \\
\text{where} \quad \bar{X}_j &= \frac{1}{n_j}\sum_{i=1}^{n_j}\boldsymbol{x}_i^{(j)}.
\end{aligned} \tag{6.39}$$

Escobar (1994) noted that the procedure should be carried out iteratively so that the joint distribution of samples from each group of parameters in the update stage will converge to the true joint posterior distribution consistent with the data. Therefore the most probable number of clusters will be determined automatically and the data will be clustered as the updating steps iterate.

Parameter estimation for GMMs

By applying DP for clustering analysis, multimodal data are clustered and the statistical parameters of each cluster are estimated. However, Chang et al. (2018) recommended the re-estimation of the mean and the covariance matrix of a Gaussian component. The mean and the covariance matrix are estimated using the sample mean and covariance of the cluster that is associated to each Gaussian component, as shown in Eqn. (6.40). These estimates are more likely to be accurate especially when the sample size is sufficiently large because

the initialization of NIW distributions in DP may introduce errors to the parameter estimation.

$$\begin{aligned}\hat{\boldsymbol{\mu}}_j &= \bar{X}_j \\ \hat{\Sigma}_j &= \frac{1}{n_j - 1}\sum_{i=1}^{n_j}(\boldsymbol{x}_i^{(j)} - \bar{X}_j)(\boldsymbol{x}_i^{(j)} - \bar{X}_j)^{\mathrm{T}}.\end{aligned} \tag{6.40}$$

To summarize, the number of clusters, the assignment of data to the clusters, and the estimation of the parameters of the Gaussian distributions associated with each cluster will be obtained by the DP-GMM clustering analysis.

6.6.3 On-line monitoring framework

The flowchart in Fig. 6.14 presents the on-line implementation of this algorithm for identification of new operating modes and updating the monitoring model. The monitoring model is trained off-line using historical data and then used for process monitoring for on-line fault detection. The training dataset is normalized all together using the mean and covariance of the entire dataset before applying DP for data clustering. It is not advisable to perform normalization for each individual operating mode, as it will result in the different data clusters overlapping about the origin. Moreover, as shown previously, the NSDC-KPCA method can build a good global model for the entire multimodal dataset without normalizing the data from each mode separately. Therefore, the on-line data can be normalized with respect to the whole training set. This is convenient in on-line monitoring because the current operating mode of the plant may be unknown.

In the off-line training stage, the historical data are then divided into training data and cross-validation data. The kernel width for the NSDC kernel is tuned using the strategy proposed in Section 6.4.3. The NSDC kernel function generates the kernel matrix. PCA obtains kernel PCs using the kernel matrix and builds the monitoring model. SPE is calculated using the kernel PCs. The control limit of SPE is determined accordingly.

The on-line monitoring stage uses the monitoring model and the control limit of SPE for process monitoring. There is no need to cluster the on-line data. The on-line data are first normalized using the mean and standard deviation of the training data. The monitoring model uses normalized on-line data to calculate the SPE for these data. In on-line implementation, an *anomaly* is a data sample that is recognized by the monitoring model to be

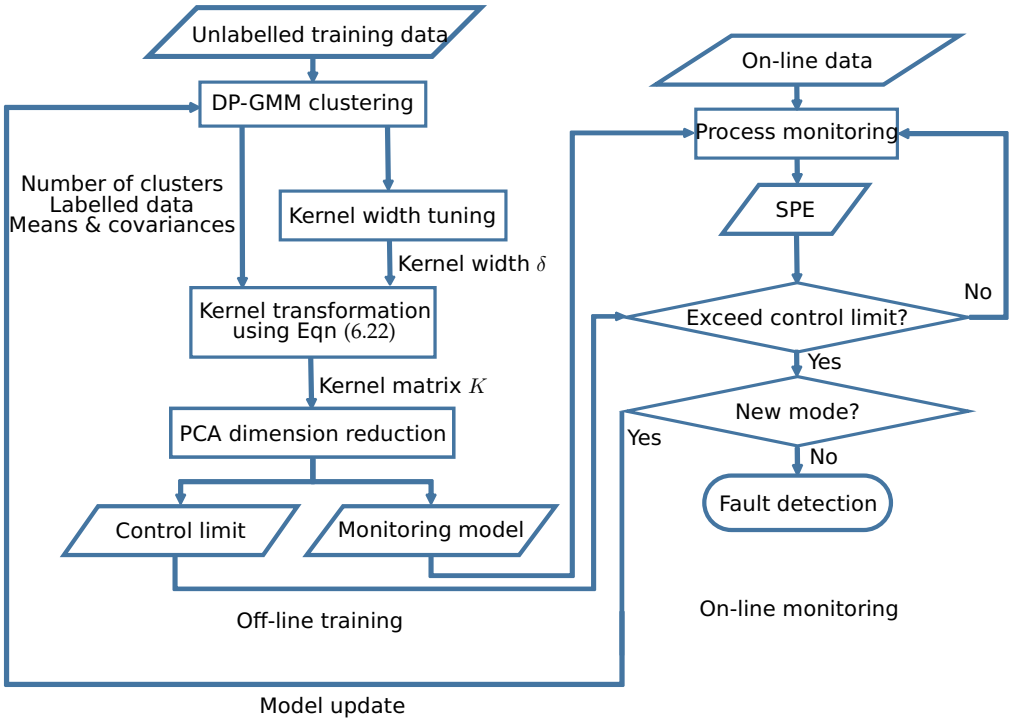


Figure 6.14: Flowchart for the on-line framework from Tan et al. (2020a)

unseen in the historical data. For example, an anomaly will be detected if the SPE of this sample exceeds its control limit.

The detection of an anomaly means that the process has moved to a new operating mode that was not present in the training set. This could be a healthy operating mode or it could be a fault. At this stage, the process operator has to intervene to classify the new mode as normal or faulty. If the operator confirms that the process has moved to a new operating mode, then the data from the new operating mode will be incorporated in the clustering stage along with the historical data. DP will determine the data cluster for the new mode and the parameters associated to it. The NSDC-KPCA method will then take the new cluster into account and update the monitoring model. Otherwise a fault will be detected if the operator does not confirm a new mode of operation.

6.7 Performance on the PRONTO dataset

This section uses the data from the PRONTO benchmark dataset described in Chapter 4 to verify the performance in process monitoring of the proposed NSDC kernel using real-life

Table 6.3: Healthy operating modes in off-line example

Mode	Input air flow rate	Input water flow rate	Data length
A	120 $Sm^3\ h^{-1}$	0.1 $kg\ s^{-1}$	501
B	150 $Sm^3\ h^{-1}$	0.5 $kg\ s^{-1}$	561

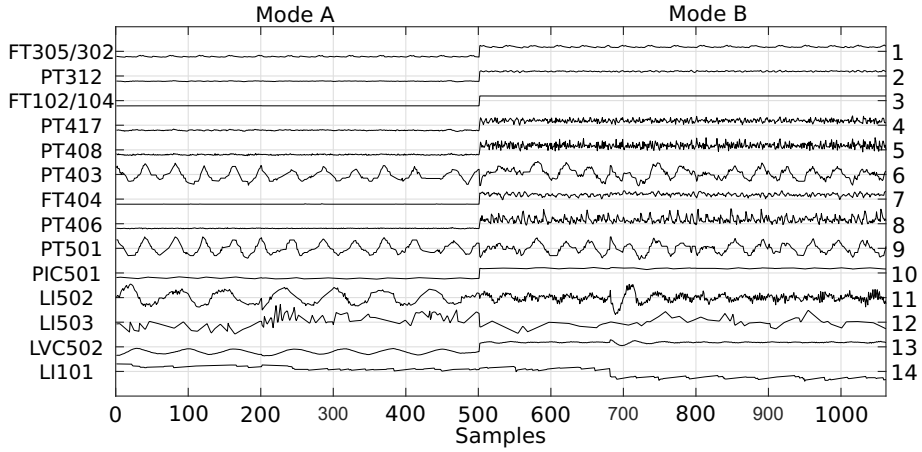


Figure 6.15: High density plot of healthy data from the PRONTO dataset used in off-line example

industrial data. Moreover, the dataset is used to create the scenario of on-line implementation where the process moves to new operating modes. The section tests the on-line monitoring framework, especially the model update step, using this dataset.

6.7.1 Off-line performance comparison

This section compares the off-line performance in process monitoring of NSDC-KPCA and the performance of RBF-KPCA using the data collected during the artificial faults.

Data description

The detailed description of the PRONTO dataset is available in Chapter 4. The example here uses the healthy data from the following two healthy operating modes, Mode A and Mode B, for training the monitoring models. Table 6.3 shows the air flow rate and the water flow rate for these two modes. The high density plot in Fig. 6.15 visualizes the normalized time trends and the tags of the process variables used for training the monitoring models.

Three artificial faults were induced in these two modes. The faults are the input air blockage, input air leakage and the diverted flow. The details for generating these faults are

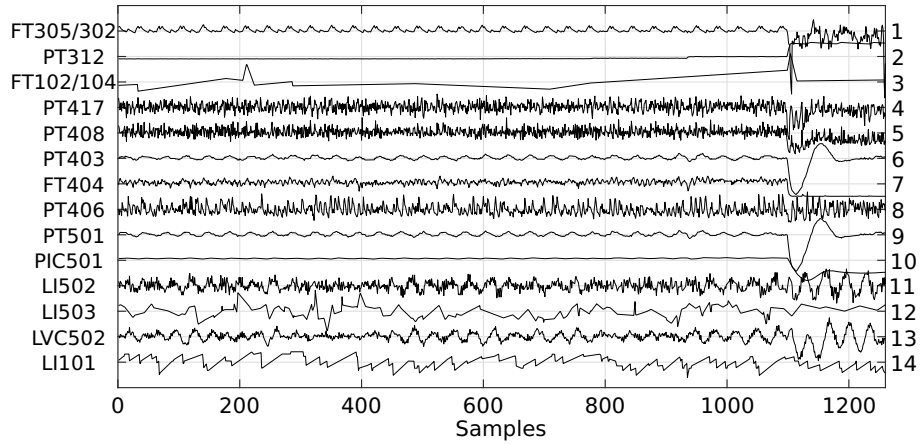


Figure 6.16: High density plot of faulty data from the PRONTO dataset used in off-line example with the blockage fault F1

Table 6.4: Information of faulty data in off-line example

Fault number	Fault	Valve for fault generation	Data length	
			Mode A	Mode B
F1	Input air leakage	V10	920	840
F2	Input air blockage	V11	1360	1260
F3	Riser base diverted flow	U39	1395	1100

given in Section 4.3.2. Starting from healthy operating condition, the valve opening is changed gradually in order to mimic the development of incipient faults in real-life process operations. As an example, Fig. 6.16 presents the high density plot of the process variables when the air blockage fault was induced in operating mode B. Table 6.4 summarizes the information about the faults and the length of the data collected when each fault was induced. The healthy data and the faulty data have the same sampling interval of three seconds.

Results on healthy data

The healthy data from Mode A and Mode B are randomly partitioned into training, cross-validation, and test sets with equivalent number of samples. The training and cross-validation sets are used for training the monitoring model and for tuning the kernel width. The monitoring model is then applied to the test dataset and the faulty data for fault detection. The data are used in batch to train the monitoring model and detect the faults.

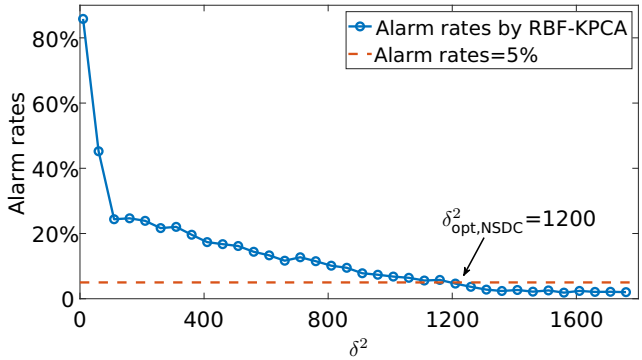
The time trends of the process variables in the PRONTO dataset have temporal correlation, whereas NSDC was designed on the basis that every sample is independent from

each other. Nevertheless, it is possible to use NSDC with the PRONTO dataset because the example uses mainly data from steady-states and the temporal correlation in the time trends is not significant. Moreover, as described in Chapter 4 and plotted in Fig. 6.16, artificial faults mainly influence the steady-state values of the process variables. Therefore, it is possible to apply NSDC-KPCA to this example. The detection criterion also requires that a fault is detected if the monitoring statistic exceeds the control limit for a sequence of at least 20 samples. The reason is that a fault is likely to result in a persistent change in the system which will be reflected in the time trends, whereas noise will lead to random erroneous alarm triggers. A future direction of the research is to extend the kernel methods to dynamic process monitoring so that the methods can monitor faults in both the steady-states and the dynamics of a process. This point is discussed further in Chapter 8, the Critical Evaluation chapter.

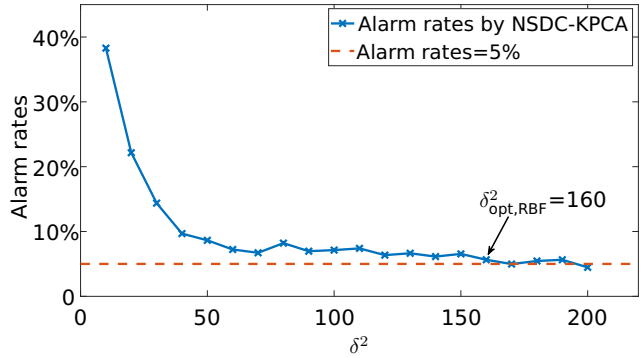
The same strategy as in Eqn (6.29) is adopted for grouping the features obtained by kernel PCA into PCs and residuals. The confidence level of control limits for SPE is set as 99%.To evaluate the performance, the FAR defined in Eqn (6.30) on the cross-validation set is compared over the kernel width δ values in Fig. 6.17. The test uses the Monte Carlo approach to reduce the uncertainty in the alarm rates due to the random partition. The healthy data are randomly partitioned 10 times. The alarm rate visualized in Fig. 6.17 for each δ value is the averaged value of the alarm rates achieved by the Monte Carlo approach. The acceptable level of the alarm rate on the cross-validation sets is set as 5%.

Fig. 6.17 shows that reduced δ values will lead to larger alarm rates on cross-validation sets, of which the samples are supposed to be healthy. This indicates that the model is over-fitted. On the other hand, smaller alarm rates imply that the monitoring model might have a higher missed detection rate and hence be less sensitive to faults when the δ value is large. Therefore, the optimal kernel width for RBF-KPCA is $\delta^2_{\text{opt,RBF}}$=1200 while the optimal kernel width for NSDC-KPCA is $\delta^2_{\text{opt,NSDC}}$=160 because these two values have achieved the acceptable level of the alarm rate. The rest of the section uses the optimal kernel widths for RBF-KPCA and NSDC-KPCA to generate the results of fault detection for comparison.

Figs 6.18-6.20 present the monitoring statistics of test dataset in healthy operations obtained by linear PCA, RBF-KPCA and NSDC-KPCA, respectively. Both T^2 and SPE are used for linear PCA. According to the recommendation in Chapter 5, SPE is used for RBF-KPCA and NSDC-KPCA. For the clarity of visualization, SPE statistics are plotted in the logarithmic scale in Fig. 6.19. In Fig. 6.18, a jump occurs to the magnitude of T^2 obtained by linear



(a) Alarm rates on cross-validation sets of healthy data by RBF-KPCA



(b) Alarm rates on cross-validation sets of healthy data by NSDC-KPCA

Figure 6.17: Alarm rates on cross-validation sets of healthy data

PCA. The SPE obtained by linear PCA has a larger variance when the process switched to Mode B. According to Fig. 6.15, the switch from mode A to Mode B happened at around the 500th sample. This is consistent with the data length presented in Table 6.3.

A similar issue exists for the SPE obtained by RBF-KPCA in 6.19. This is due to the limitation of RBF kernels in handling multimodal data. The kernel width of RBF kernels will be overly-relaxed in order to account for multiple modes in the data. The SPE obtained by NSDC-KPCA does not have such an issue. The NSDC kernel can ensure that the monitoring statistics of the two different operating modes have similar magnitudes. It is clear that the influence of multimodality in the training set can be addressed by the NSDC kernel. Moreover, the NSDC kernel is able to deliver many of the requirements for ideal monitoring statistics discussed previously in the thesis.

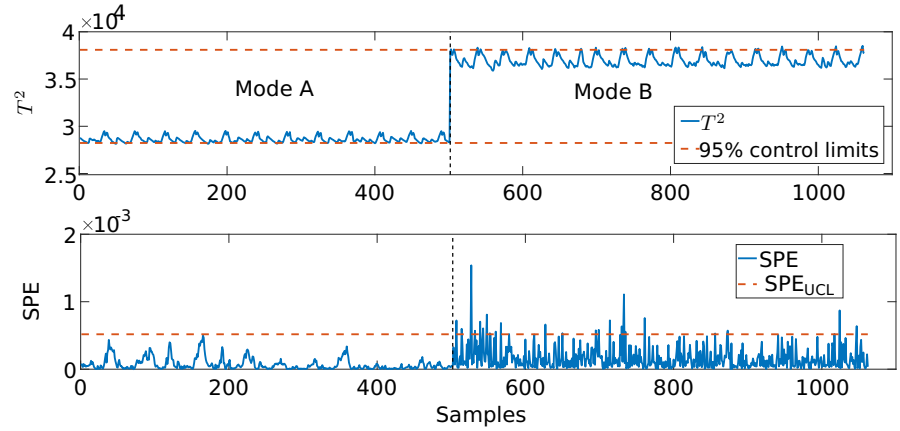


Figure 6.18: Monitoring result for healthy data: Linear PCA

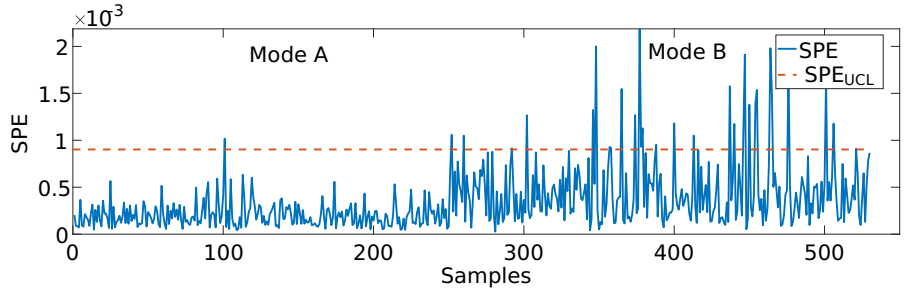


Figure 6.19: Cross-validation result for healthy data: RBF kernel

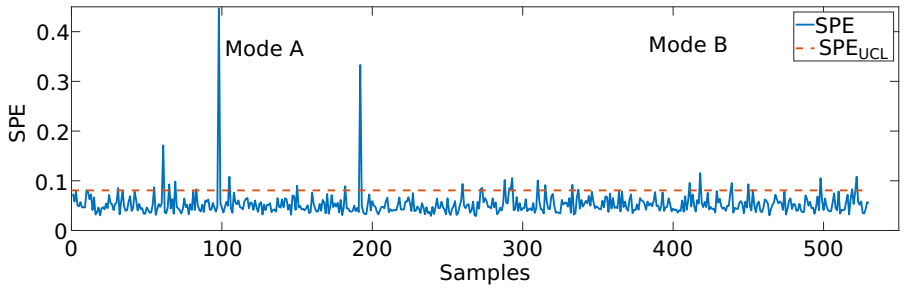


Figure 6.20: Cross-validation result result for healthy data: NSDC kernel

Results on faulty data

Eqn (6.41) defines the Detection Time (DT) for faults.

$$\text{DT} = t(\text{fault detection}) - t(\text{fault occurrence}). \tag{6.41}$$

DT provides a measure of the sensitivity of the monitoring model in this example. Since the fault severity increased gradually in the experiment (e.g. the valve opening sequence in Fig. 6.21) and the variation in process variables may not be visible in the early stage (Fig. 6.16), it is difficult to define the faulty period clearly. Therefore, the DT is used instead of the missed detection rate. Due to the persistent existence of the fault, fault detection occurs

Table 6.5: Monitoring performance of NSDC-KPCA and RBF-KPCA in off-line example

Kernel	Healthy	Fault scenario					
	FAR (%)	Mode A: DT			Mode B: DT		
		F1	F2	F3	F1	F2	F3
RBF	4.34	229	836	575	156	635	OD[1]
NSDC	5.09	229	515	OD	158	635	OD

[1] OD stands for the case when a detection occurs from the first sample.

when a consecutive sequence of 20 samples exceeds the control limits. This reduces the influence of noise in the process measurements.

The second column of Table 6.5 compares the FARs of the NSDC and the RBF kernels on the test sets which comprise normal samples that are not used for model training and parameter tuning. Since the confidence level is set to be 95%, the tuning strategy in Section 6.4.3 ensures that both the RBF and the NSDC kernels have a FAR that is close to 5% on the test set.

Table 6.5 shows that the process monitoring model built by NSDC-KPCA is capable of detecting the faults earlier than the model built by RBF-KPCA when air blockage and diverted flow occur to Mode A. The reason is the SPE behaviour in Figs 6.19 and 6.20. In Fig. 6.19, the SPE for Mode B has a larger variance than the SPE for Mode A. The control limit of SPE is set based on the SPE values in both modes. Therefore, it is more difficult for the SPE to exceed the control limit in Mode A. Fig. 6.21 visualizes the sequence of valve opening adopted for seeding Fault 2 in Mode A. In Fig. 6.22 and 6.23, the monitoring performance in Mode A is compared against this sequence. The performance indicates that, at the early stage of an incipient fault and when the deviation in principal components is not significant, small model-data mismatches existing in process measurements caused by the fault can be captured by the monitoring model using the NSDC kernel. As a result, the incipient fault can be detected and dealt with before severe performance degradation occurs in the process. In addition, the SPE statistic fulfils the expectation for ideal monitoring statistics such that the monitoring statistic should increase monotonically as the severity of the fault increases.

It may be noticed that the valve opening sequence in Fig. 6.21 has step changes while real-life faults may have fault magnitude that ramps up gradually overtime. Due to the

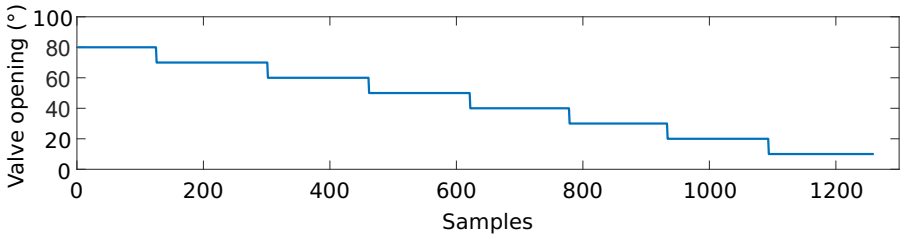


Figure 6.21: Valve opening sequence for Fault 2, Mode A

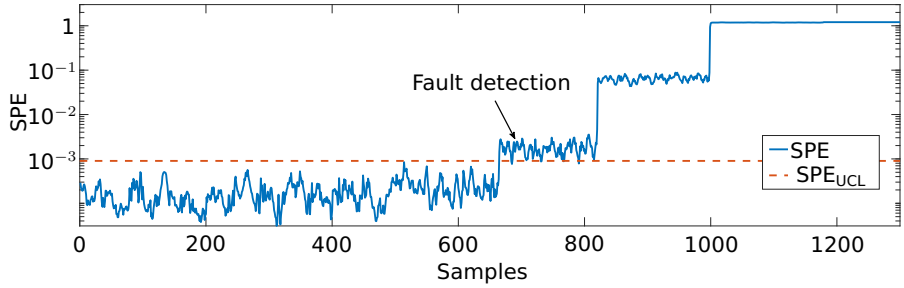


Figure 6.22: RBF monitoring performance for Fault 2, Mode A

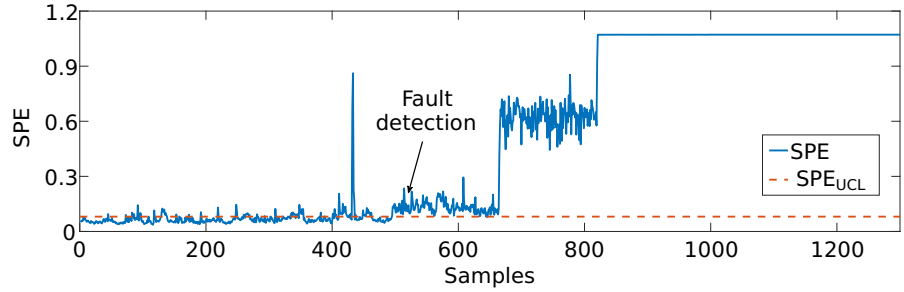


Figure 6.23: NSDC monitoring performance for Fault 2, Mode A

limitation of the experimental facility for obtaining the PRONTO dataset, it is difficult to induce such faults. However, the performance of the NSDC kernel in response to such faults can be inferred from the results presented in Section 6.5.1. The test sequences in Fig. 6.10 can be considered as samples that gradually deviate from healthy training data and NSDC-KPCA can detect the case even if the deviation is minor.

6.7.2 On-line monitoring results

This section uses the data from several operating modes in the PRONTO dataset to simulate the on-line implementation. The on-line monitoring framework described in Section 6.6 is applied to these data. The section will compare the monitoring performance when model update step is used with the performance when there is no model update.

Table 6.6: Healthy operating modes from the PRONTO dataset in on-line example

Mode number	Mode 1	Mode 2	New 1	New 2
Input water (0.1 kg s^{-1})	0.1	0.1	0.5	0.5
Input air (Sm^3 h^{-1})	120	200	150	100

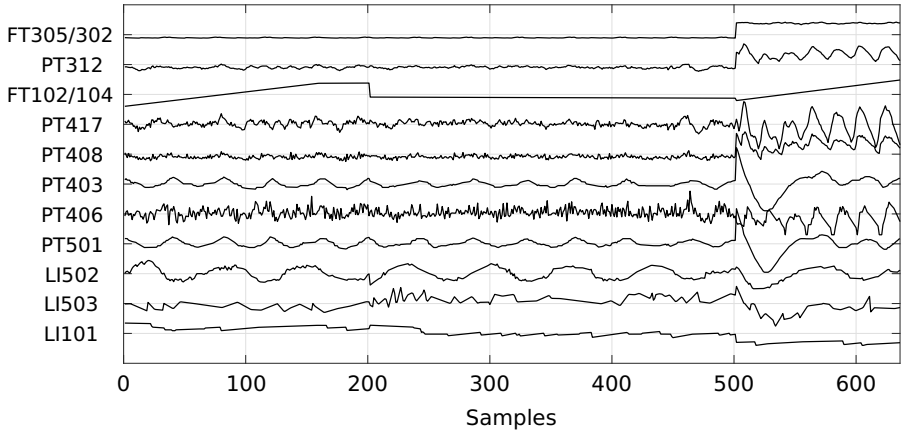


Figure 6.24: High density plot of data from the PRONTO dataset for off-line training

Data description

Table 6.6 presents the four operating modes included in constructing the training and the test datasets. The sampling interval for these measurements is three seconds. The training set includes data from Mode 1 and Mode 2. Fig. 6.24 plots the time trends and the tags of the process variables in the training set. The test set comprises of additional data from Mode 1 and Mode 2, and data from the two new modes (New 1 and New 2). Fig. 6.25 plots the trends and the tags of the process variables in the test set. After 1417 samples, the air blockage fault was induced under operating mode New 2 by gradually closing the manual valve on the input air line. The sequences of operating modes and the valve opening for introducing the fault are presented in Fig. 6.26.

Performance

In this test, the monitoring model is trained off-line using the proposed DP-NSDC algorithm. The monitoring performance of the implementations with and without model updates are compared in Fig. 6.27.

Due to the switching between existing modes and new modes, the monitoring statistic goes above the control limit at sample 501 (A_1 in Fig. 6.27(a)), returns to healthy at sample 1090

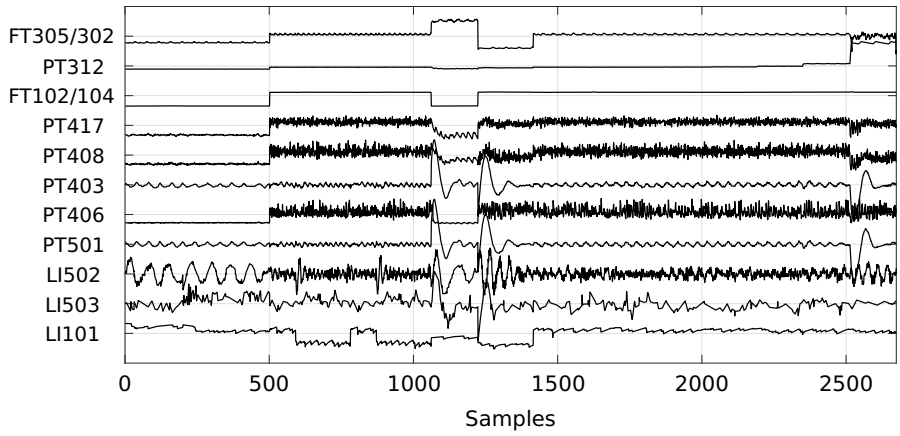


Figure 6.25: High density plot of data from the PRONTO dataset for on-line testing

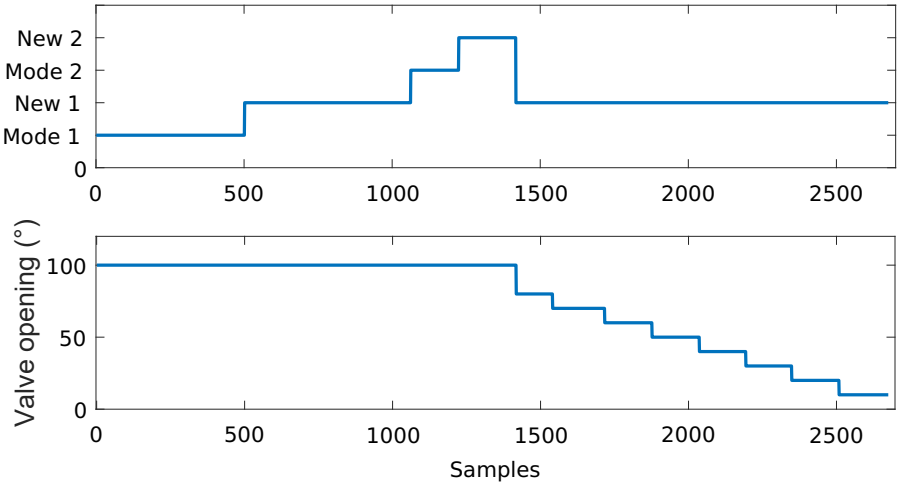
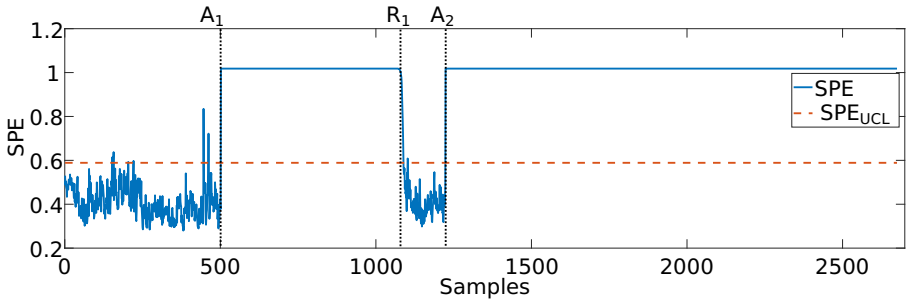


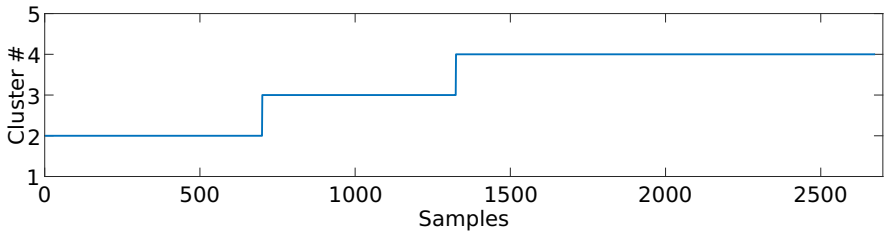
Figure 6.26: Operating sequence for the test set

(R_1) and goes above again at sample 1224 (A_2). It may be observed that the monitoring model is sensitive to mode changes, making it qualified for anomaly detection, but unable to distinguish between a new operating mode and a fault in the process.

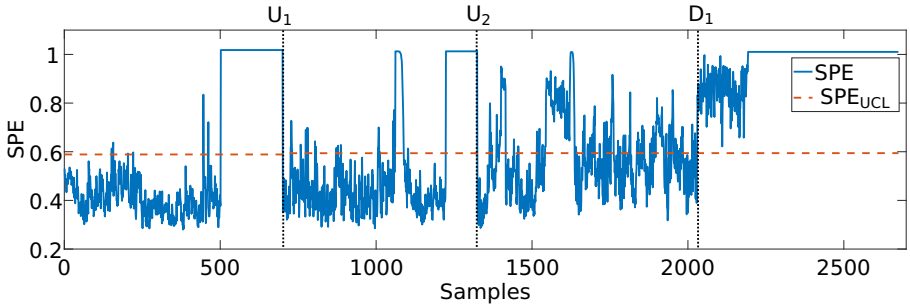
The monitoring performance in Fig. 6.27(c) has improved due to the model update. At sample 700, the anomalies are acknowledged by the operator as a new operating mode. The DP-NSDC algorithm re-trains the monitoring model using the data from this anomaly period and historical data to account for the new mode. Hence the monitoring statistic returns to its control limit as soon as the model has been updated (U_1 in Fig. 6.27(c)). The same update happens at U_2. Such updates can reduce the false alarms triggered due to the new mode, e.g. for the period from sample 700 to sample 1090. Fig. 6.27(b) shows the number of clusters which DP obtained before and after model update. Therefore, DP is capable of identifying the emerging new mode using historical and incoming data. Moreover, by incorporating the incoming data, this framework can detect faults when the process is



(a) Monitoring performance: without update



(b) Number of clusters identified by DP



(c) Monitoring performance: with update

Figure 6.27: Monitoring results for the on-line test set

running in the new mode that has appeared previously in on-line operations, but not in the historical data used for training, as shown by C_2 in Fig. 6.27(c). Since the fault was induced using the valve opening sequence in Fig. 6.26, it was developing gradually over time. Therefore, a delay exists between the fault induction and the fault detection. Nevertheless, the proposed method can detect this fault at its early stage before the variation in process variables becomes visible (Fig. 6.25).

The on-line monitoring framework enables the monitoring model to incorporate insights regarding operating conditions in real-time and update when necessary while maintaining sensitivity in fault detection. Additionally, it may be observed in Fig. 6.27(c) that there is a short period of false alarms around sample 1100 due to the transition between New 1 and Mode 2. To account for the transition and avoid false alarms is a subject of a future dynamic extension of the algorithm.

To sum up, the proposed NSDC kernel in this chapter fulfils the first two considerations of practical applications discussed in Section 2.4. The new kernel can reduce false alarms and missed alarms because it can improve the accuracy of monitoring models for multimodal data. In particular, the tuning strategy reduces false alarms and missed alarms of the monitoring models obtained using NSDC kernels. The NSDC kernel also produces SPE statistics that do not react to the change of known operating modes whilst increasing monotonically with respect to the severity of faults under the KPCA framework. Thus the interpretability of the monitoring results is improved. The on-line framework fulfils the other two considerations Section 2.4. The framework can incorporate data from unseen operating modes in the process to update the monitoring model, making the model robust to the new modes. The updated model can also detect faults occurring in these new operating modes.

6.8 Chapter summary

In this chapter, the limitation of the RBF kernel in handling multimodal data is first demonstrated. The chapter then proposed the novel NSDC kernel which is data-dependent and can describe multimodal data. The tuning issue and the monitoring statistics were discussed to use the NSDC kernel properly for fault detection. This chapter also introduced a clustering method using DP-GMM. The clustering step can prepare the training data to be used by the NSDC kernel such that the off-line training of monitoring models does not need the labels indicating in which operating mode the healthy samples were collected. The on-line framework incorporates expert knowledge from process operators to distinguish between a fault and an unseen healthy operating mode. The monitoring model is then updated using data from unseen healthy operating modes.

The results of numerical simulated examples demonstrate that the NSDC kernel can better address multimodal data when compared to the RBF kernel and other methods proposed for multimodal process monitoring. The results using the PRONTO dataset shows the ability of the NSDC kernel to build monitoring models for multimodal data. In particular, the monitoring statistic does not react to the mode changes but is sensitive to faults or unseen operating modes. The on-line implementation results show that the on-line framework can update the monitoring model to account for healthy modes that are not seen in the training data. The new kernel and the on-line framework address the considerations for practical implementations of monitoring methods discussed in Chapter 2.

Chapter 7

Contribution-based fault diagnosis for multimodal processes

This chapter investigates contribution-based fault diagnosis when kernel methods are applied. The chapter first discusses the insights of kernel methods established in the previous chapters, then revisits the existing way of defining contributions for linear methods. The concept of the *deviation contribution*, which is the contribution of a process variable to a deviation in the monitoring statistic, is proposed. A deviation contribution is an approximated measure of the contribution of process variables to the monitoring statistic. This new deviation contribution is suitable for fault diagnosis when nonlinear methods are used.

The chapter also discusses several aspects of the concept of deviation contributions, such as the relationship with the contribution in linear methods and the selection of reference samples during calculation. The formula for calculating the deviation contribution of process variables under the RBF-KPCA framework is derived. A numerical example is presented to validate the performance of the new deviation contribution when applied to fault diagnosis. In particular, this example illustrates that the deviation contribution can still provide useful insights for fault diagnosis when the SPE obtained by RBF-KPCA approaches to a constant, as can happen for large anomalies.

7.1 Background and insights from previous chapters

This section first reviews previous works in contribution-based fault diagnosis used for both linear and nonlinear process monitoring. The investigation of kernel methods in previous chapters provides insights to contribution-based fault diagnosis when kernel methods are applied and Section 7.1.2 outlines these insights.

7.1.1 Background

After a fault is detected, fault diagnosis aims to provide additional information about the nature of the fault to end-users to facilitate their decisions concerning production and maintenance. Examples of additional information includes the location of the fault, and the type of the fault. Section 3.3.3 reviewed existing methods for fault diagnosis. Table 3.7 summarized the results of fault diagnosis that can be delivered to end-users.

This chapter focuses on delivering a method for determining the contributions of variables described by Table 3.7. The contribution of a variable refers to the portion of the value of a monitoring statistic due to that variable. Section 7.2 will review the mathematical definition of such contributions. Contribution-based fault diagnosis is suitable for multivariate statistical process monitoring because it can identify process variables that are related to the occurrence of faults.

Contribution plots measure how strongly process variables are related to the fault occurrence by quantifying the contribution of these variables to the monitoring statistics (Miller et al., 1998). Contribution plots have been widely applied to various multivariate linear methods, such as principal component analysis (Chiang et al., 2000a), multiway principal component analysis (Kourti et al., 1995), independent component analysis (Lee et al., 2004b), and canonical variate analysis (Jiang et al., 2015).

Alcala and Qin (2010) pointed out that the extension of standard contribution plots is difficult when nonlinear methods, such as kernel PCA, are adopted. In order to accommodate kernel methods in nonlinear process monitoring, Cho et al. (2005) and Zhang et al. (2013) used the contribution rate, which is defined as the first order derivatives of a monitoring statistic with respect to the original variables. Alcala and Qin (2010) proposed the reconstruction-based contribution for nonlinear process monitoring. In their work, the

magnitude of a process variable is reduced by a certain amount such that the monitoring statistic is smaller than the control limit. The work defined the reconstruction-based contribution of a process variable as such a minimum amount of the magnitude of this process variable to be reduced.

In linear methods, the contribution of a variable is defined as the portion of the value of a monitoring statistic due to this variable. However, the contribution rate and the reconstruction-based contribution in nonlinear methods are not portions of the value of a monitoring statistic because the reconstruction-based contribution has the same unit of a variable and the contribution rate is a derivative of the monitoring statistic. The previous chapters have described several methods to achieve a desired behaviour of the monitoring statistic for nonlinear process monitoring so that the monitoring statistic can reflect the magnitude of faults. This makes it possible to define contributions such that the contributions are portions of monitoring statistics. The next section summarizes the insights of the previous chapters and explains why such a definition is preferred.

7.1.2 Insights from previous chapters

Section 2.6.2 compared the contribution plots obtained for the two monitoring statistics, namely T^2 and SPE, for the illustrative multimodal example. The results indicate that the following two points need to be achieved when using contributions of variables for fault diagnosis.

- For healthy data, the contributions of the variables should all be relatively low and be of the same magnitude regardless of operating modes;
- For faulty data with one particular variable being faulty, the contribution of the faulty variable should be visibly higher than the contributions of other variables that are not directly influenced by the fault.

The previous chapters on kernel-based methods for fault detection have achieved the following behaviour of the SPE when applied to multimodal processes:

1. The monitoring statistic SPE has the same magnitude for healthy data from several operating modes;

2. The monitoring statistic SPE increases monotonically as faulty samples deviate from the healthy data;

3. The derivation and the results in Section 5.4.1 show that SPE approaches to a maximum value as a faulty sample deviates from the training data. This comment is valid for KPCA with RBF kernels and NSDC kernels.

Such behaviour enables fault diagnosis that is based on contributions of variables. This is because SPE increases monotonically as a fault develops, making it reasonable to define the contribution of a process variable to SPE as the portion of the value of SPE due to this variable. If a variable has a large contribution to SPE, this variable is likely to be related to the fault occurrence.

Nevertheless, a limitation of contributions exists when the contribution to SPE is defined in a similar way to Eqn (7.6), which assumes that SPE can be written as the summation of the contributions of process variables to SPE. As the fault develops, the contributions of variables related to the fault should increase. However, the contributions will approach to constant values as the SPE approaches to a constant if Eqn (7.6) holds. Therefore, it is necessary to define the contributions differently when KPCA uses RBF kernels or NSDC kernels.

This chapter proposes a novel way to define the contributions of process variables to monitoring statistics, especially when kernel methods are applied. The new deviation contribution proposed in this chapter provides a measure of how a process variable is related to a fault occurrence. If a sample is collected when a fault occurs in the process, the monitoring statistic of this sample will be large and the contribution of the process variables that are most related to this fault will also be large. A journal paper Tan and Cao (2019) is based partially on the work in this chapter. The chapter takes the behaviour of monitoring statistics in kernel methods into account to improve the concept of deviation contribution defined in Tan and Cao (2019).

7.2 Mathematical definition of contributions

This section reviews the standard definition of contributions used in linear methods and illustrates why such a definition cannot apply directly to nonlinear methods.

Chiang et al. (2000b) defined the contribution of process variables to monitoring statistics in PCA. Assuming that $\boldsymbol{x} = \{x_1, \cdots, x_m\}$ is a vector of m process variables and $X = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n\}$ is a matrix with n samples of $\boldsymbol{x}$, the normalized data are denoted as $\tilde{X} = \{\tilde{\boldsymbol{x}}_1, \cdots, \tilde{\boldsymbol{x}}_n\}$. PCA is applied to the normalized data X and the first L principal components retained after applying PCA are denoted as $\boldsymbol{z} = \{z^{(1)}, \cdots, z^{(L)}\}$. For the j-th sample $\boldsymbol{x}_j$, the projection of normalized $\tilde{\boldsymbol{x}}_j$ to the principal components is $\boldsymbol{z}_j$.

The monitoring statistic T_j^2 of the j-th sample $\boldsymbol{x}_j$ is defined using $\boldsymbol{z}_j$ as:

$$T_j^2 = \boldsymbol{z}_j^\top \Lambda^{-1} \boldsymbol{z}_j \tag{7.1}$$

where $\Lambda = \text{diag}\{\lambda_1, \cdots, \lambda_L\}$ is the diagonal matrix with the first L eigenvalues. Since T^2 is a quadratic function of the principal components, the contribution of $x_{r,j}$, the value of the r-th variable x_r in sample $\boldsymbol{x}_j$, to T_j^2 is considered to be the summation of the contributions of $x_{r,j}$ to the value of each principal component. The contribution of the normalized value of the r-th process variable $\tilde{x}_r$ to the l-th principal component $z_j^{(l)}$ is defined as:

$$\mathscr{c}_{l,r,j} = \frac{z_j^{(l)}}{\lambda_l} \tilde{x}_{r,j} \tag{7.2}$$

where λ_l is the l-th eigenvalue corresponding to the l-th principal component. Then the total contribution of the r-th variable $x_{r,j}$ to the monitoring statistic T_j^2 is

$$\mathscr{C}_{T^2,r,j} = \sum_{l=1}^{L} \mathscr{c}_{l,r,j} \tag{7.3}$$

The monitoring statistic SPE_j is defined as the sum of squares of the difference between the original sample $\boldsymbol{x}_j$ and $\hat{\boldsymbol{x}}_j$ reconstructed by the PCA model:

$$\text{SPE}_j = \|\tilde{\boldsymbol{x}}_j - \hat{\tilde{\boldsymbol{x}}}_j\|_2 = \sum_{r=1}^{m} \left(\tilde{x}_{r,j} - \hat{\tilde{x}}_{r,j}\right)^2 \tag{7.4}$$

where m is the number of variables in $\boldsymbol{x}$. $\hat{\tilde{\boldsymbol{x}}} = P_L^\top P \tilde{\boldsymbol{x}}$ represents the reconstructed $\tilde{\boldsymbol{x}}$ according to the PCA model. P_L is the projection matrix to obtain the first L principal components. Therefore, the contribution of the r-th variable x_r to SPE is:

$$\mathscr{C}_{\text{SPE},r,j} = \left(\tilde{x}_{r,j} - \hat{\tilde{x}}_{r,j}\right)^2 \tag{7.5}$$

The following relationship holds between the contributions of individual variables and the SPE:

$$\text{SPE}_j = \sum_{r=1}^{m} \mathscr{C}_{\text{SPE},r,j} \tag{7.6}$$

The same definition may not propagate directly to KPCA because the method to calculate SPE is different from the method used in PCA. Chapter 5 defines the SPE as the quadratic function of the mismatch between the kernel features $\tilde{\Phi}(\boldsymbol{x})$ and the reconstructed $\hat{\tilde{\Phi}}(\boldsymbol{x}_j)$, such that:

$$\begin{aligned} \text{SPE}_j &= \|\tilde{\Phi}(\boldsymbol{x}_j) - \hat{\tilde{\Phi}}(\boldsymbol{x}_j)\|_2 \\ &= 1 - 2\bar{K}_j + \bar{K} - \boldsymbol{z}_j^\top \boldsymbol{z}_j \end{aligned} \tag{7.7}$$

where K is the kernel matrix, $\bar{K}_j$ is the mean of the kernel vector $K_j = \{k(\boldsymbol{x}_1, \boldsymbol{x}_j), \cdots, k(\boldsymbol{x}_n, \boldsymbol{x}_j)\}$, and $\bar{K}$ is the mean of the kernel matrix. The kernel principal components of the j-th sample $\boldsymbol{x}_j$ are denoted as $\boldsymbol{z}_j$. The rest of the work described in this chapter will focus on SPE because Chapter 5 concluded that SPE is the appropriate monitoring statistic for RBF-KPCA.

When using Eqn (7.5) it may not be possible to clearly identify the contributions of individual process variables as portions of SPE. The reason is that Eqn 7.7 cannot be written as a summation of functions of each process variable. Instead, it it necessary to write both $\Phi(\boldsymbol{x})$ and $\hat{\tilde{\Phi}}(\boldsymbol{x})$ as summations of functions of each process variable. When the RBF kernel is applied, it is also not possible to first define the contribution of variable x_r to each feature in $\Phi(\boldsymbol{x})$ because $\Phi(\boldsymbol{x})$ cannot be calculated explicitly. This also applies to $\hat{\tilde{\Phi}}(\boldsymbol{x})$. Therefore, the contribution of variables to SPE when KPCA is applied needs to be defined in a different way from the definition given by Eqn (7.5).

7.3 Deviation contributions

This section presents how the deviation contribution is derived. One issue in implementation of the concept of deviation contribution, namely the selection is reference point, will also be discussed.

7.3.1 Preliminary

Eqn (7.8) summarizes the mathematical structure of a multivariate statistical monitoring method. Given a vector of process variables $\boldsymbol{x}$, a multivariate method extracts representative features $\boldsymbol{z}$ from $\boldsymbol{x}$ by using a function $\boldsymbol{f}$. In this chapter the monitoring statistic is denoted by the symbol I. I is often a function of the representative features $\boldsymbol{z}$ and often takes a quadratic form. For example, the T^2 statistic in PCA is a quadratic function of the principal components. The SPE statistic in KPCA is a quadratic function of the mismatch between the kernel features and the reconstructed kernel features. The upper control limit of I, I_{UCL}, is specified based on its probability density function P with a confidence level α. Eqn (7.9) indicates that, for healthy data, the probability that the monitoring statistic I is smaller than I_{UCL} is α.

$$\boldsymbol{z} = \boldsymbol{f}(\boldsymbol{x});\ I = I(\boldsymbol{z}) \tag{7.8}$$

$$P(I < I_{\text{UCL}}) = \alpha \tag{7.9}$$

When applied to fault detection, the monitoring statistic I_* of a new sample $\boldsymbol{x_*}$ is calculated using Eqn (7.10) and is compared against the control limit I_{UCL}.

$$\boldsymbol{z_*} = \boldsymbol{f}(\boldsymbol{x_*});\ I_* = I(\boldsymbol{z}_*) \tag{7.10}$$

$\boldsymbol{x}_*$ is detected as faulty if the monitoring statistic I_* exceeds its upper control limit:

$$I_* > I_{\text{UCL}} \tag{7.11}$$

Section 7.2 presents several examples of definitions of contributions to monitoring statistics. When PCA is applied, the mapping $\boldsymbol{f}$ from the process variables $\boldsymbol{x}$ to the features $\boldsymbol{z}$ is a linear function. The following relationship holds between the contributions of the variables to the monitoring statistic and the monitoring statistic itself:

$$I_* = \sum_{r=1}^{m} c_{I,r,*} \tag{7.12}$$

where $c_{I,r,*}$ is the contribution of $x_{r,*}$, the r-th variable in $\boldsymbol{x}_*$, to the monitoring statistic I_*. This is valid if I is taken as T^2 or SPE when PCA is used because the two monitoring statistics are quadratic functions of $\boldsymbol{z}$, hence quadratic functions of $\boldsymbol{x}$.

However, the mapping $\boldsymbol{f}$ is nonlinear when kernel methods are used. Section 7.2 discussed the reason why Eqn (7.12) may not be valid for the SPE in KPCA. The rest of this section proposes the concept of deviation contribution for contribution-based fault diagnosis when kernel methods are used.

7.3.2 Definition of deviation contributions

This section derives the deviation contribution. The sample being diagnosed is $\boldsymbol{x}_*$ and its monitoring statistic is I_*. First, Eqn (7.13) applies the mean value theorem to $\boldsymbol{I}_*$ by selecting a reference sample $\boldsymbol{x}_{\text{ref}}$ that is close to $\boldsymbol{x}_*$:

$$I_* = I_{\text{ref}} + \sum_{r=1}^{m} \left.\frac{\partial I}{\partial x_r}\right|_{\boldsymbol{x}=\boldsymbol{x}_c} (x_{r,\text{ref}} - x_{r,*}) \tag{7.13}$$

where I_{ref} is the monitoring statistic of $\boldsymbol{x}_{\text{ref}}$ and $\boldsymbol{x}_c$ is a sample between $\boldsymbol{x}_*$ and $\boldsymbol{x}_{\text{ref}}$. The quantity $\frac{\partial I}{\partial x_r}$ is the first order derivative of I with respect to the r-th process variable x_r. The number of process variables is denoted by m. Thus the difference between I_* and I_{ref} can be calculated by a summation of terms contributed by each process variable:

$$\Delta I = I_* - I_{\text{ref}} = \sum_{r=1}^{m} \left.\frac{\partial I}{\partial x_r}\right|_{\boldsymbol{x}=\boldsymbol{x}_c} (x_{r,\text{ref}} - x_{r,*}) \tag{7.14}$$

The *deviation contribution* is then defined as the contribution of a process variable to ΔI:

$$\Delta \mathscr{C}_{I,*,r} = \left.\frac{\partial I}{\partial x_r}\right|_{\boldsymbol{x}=\boldsymbol{x}_c} (x_{r,\text{ref}} - x_{r,*}) \tag{7.15}$$

$\Delta \mathscr{C}_{I,*,r}$ denotes the deviation contribution of the r-th variable to the monitoring statistic I_* for the sample $\boldsymbol{x}_*$. Section 7.3.4 will discuss why the first order derivative is evaluated at $\boldsymbol{x} = \boldsymbol{x}_c$.

Section 7.2 demonstrated that the monitoring statistic SPE cannot be written in the form of the summation of the contributions of individual process variables. This problem can be resolved by considering the contributions of the process variables to the difference between the monitoring statistic of the test sample being diagnosed and the monitoring statistic of a reference sample. When the deviation contribution of process variables is defined as Eqn (7.15), the following relationship between the monitoring statistics I_* and I_{ref} and the

deviation contributions of process variables is valid by substituting Eqn (7.15) to Eqn 7.13:

$$I_* = I_{\text{ref}} + \sum_{r=1}^{m} \Delta\mathscr{C}_{I,*,r} \tag{7.16}$$

If the sample $\boldsymbol{x}_*$ is faulty and the monitoring statistic I_* exceeds its control limit, a large deviation contribution of a certain process variable indicates that this variable is a major reason that I_* exceeds the control limit. Hence the variable may be related to the fault.

Several aspects need to be specified when using the deviation contribution. Since the monitoring statistic I is a continuous monotone function of $\boldsymbol{x}$, mean value theorem guarantees the existence of a point $\boldsymbol{x}_c$ between $\boldsymbol{x}_*$ and $\boldsymbol{x}_{\text{ref}}$ that satisfies Eqn (7.13). However, the value of $\boldsymbol{x}_c$ is not known and hence the first order derivative of I with respect to $\boldsymbol{x}$ at $\boldsymbol{x} = \boldsymbol{x}_c$ cannot be calculated explicitly. Eqn (7.17) proposes an approach for estimating the first order derivative $\frac{\partial I}{\partial x_r}$ at $\boldsymbol{x} = \boldsymbol{x}_c$:

$$\left.\frac{\hat{\partial} I}{\partial x_r}\right|_{\boldsymbol{x}=\boldsymbol{x}_c} = \frac{1}{2}\left(\left.\frac{\partial I}{\partial x_r}\right|_{\boldsymbol{x}=\boldsymbol{x}_*} + \left.\frac{\partial I}{\partial x_r}\right|_{\boldsymbol{x}=\boldsymbol{x}_{\text{ref}}}\right) \tag{7.17}$$

This approximation was made because it includes the information from both $\boldsymbol{x}_*$ and $\boldsymbol{x}_{\text{ref}}$. This approximation is particularly useful when RBF-KPCA and SPE are used for fault detection. Section 5.4.1 demonstrated that SPE increases monotonically as a sample deviates from the healthy training data and will approach to a constant when this sample is significantly far away. Therefore, the first order derivative of SPE over a process variable will converge to zero. This can also be observed from the trends of SPE in Fig. 5.4. Hence the approximation in Eqn (7.17) avoids the case when deviation contributions converge to zero, which would happen if only

$$\left.\frac{\partial I}{\partial x_r}\right|_{\boldsymbol{x}=\boldsymbol{x}_*}$$

is used. Hence the deviation contribution becomes:

$$\Delta\mathscr{C}_{I,*,r} = \frac{1}{2}\left(\left.\frac{\partial I}{\partial x_r}\right|_{\boldsymbol{x}=\boldsymbol{x}_*} + \left.\frac{\partial I}{\partial x_r}\right|_{\boldsymbol{x}=\boldsymbol{x}_{\text{ref}}}\right)(x_{r,\text{ref}} - x_{r,*}) \tag{7.18}$$

The selection of $\boldsymbol{x}_{\text{ref}}$ influences the deviation contribution obtained by Eqn (7.18). Section 7.3.4 will discuss the proper selection of reference samples. Section 7.5.1 derives the analytical expressions for the derivatives $\frac{\partial I}{\partial \boldsymbol{x}}$, which enable the evaluation of the terms on the right hand side of Eqn (7.18), for RBF-KPCA.

7.3.3 Deviation contribution in linear methods

The proposed concept of deviation contributions applies to both linear and nonlinear methods for feature extraction. This section proves that the deviation contribution of a variable will represent the difference between the contributions of the same variable in $\boldsymbol{x}_*$ and $\boldsymbol{x}_{\text{ref}}$ when linear feature extraction methods are applied.

The monitoring statistic I is a quadratic function of the sample $\boldsymbol{x}$, i.e.

$$I = \beta_r x_r^2 \tag{7.19}$$

where m is the number of process variables and β_r denotes the coefficient of x_r in the formulation of I. The first order derivative of I with respect to x_r is

$$\frac{\partial I}{\partial x_r} = 2\beta_r x_r \tag{7.20}$$

When linear PCA is applied, Eqn (7.3) and Eqn (7.5) in Section 7.2 are two specific examples of the contributions of a process variable to the T^2 statistic and the SPE statistic, respectively.

Two samples $\boldsymbol{x}_*$ and $\boldsymbol{x}_{\text{ref}}$ have the monitoring statistics I_* and I_{ref}. According to Eqn (7.18), the deviation contribution of the r-th variable x_r to I_* can be calculated using $\boldsymbol{x}_{\text{ref}}$ as the reference sample:

$$\begin{aligned} \Delta\mathscr{C}_{I,*,r} &= \frac{1}{2}\left(2\beta_r x_{\text{ref},r} + 2\beta_r x_{*,r}\right)\left(x_{*,r} - x_{\text{ref},r}\right) \\ &= \beta_r\left(x_{*,r}^2 - x_{\text{ref},r}^2\right) \end{aligned} \tag{7.21}$$

Since the feature extraction method is linear and I is a quadratic function of $\boldsymbol{x}$, the contributions of x_r to I_{ref} and I_* are calculated respectively as follows:

$$\mathscr{C}_{I,*,r} = \beta_r x_{*,r}^2 \tag{7.22}$$

$$\mathscr{C}_{I,\text{ref},r} = \beta_r x_{\text{ref},r}^2 \tag{7.23}$$

The deviation contribution $\Delta\mathscr{C}_{I,*,r}$ is the difference between $\mathscr{C}_{I,\text{ref},r}$ and $\mathscr{C}_{I,*,r}$:

$$\Delta\mathscr{C}_{I,*,r} = \mathscr{C}_{I,*,r} - \mathscr{C}_{I,\text{ref},r} = \beta_r\left(x_{*,r}^2 - x_{\text{ref},r}^2\right) \tag{7.24}$$

According to Eqn (7.19), the linear methods will achieve $I = 0$ when $\boldsymbol{x} = \mathbf{0}$ and all contributions of the process variables, $\mathscr{C}_{I,\mathbf{0},r}$, are zero. Therefore, if selecting the origin $\boldsymbol{x} = \mathbf{0}$ as $\boldsymbol{x}_{\text{ref}}$, the deviation contribution $\Delta\mathscr{C}_{I,*,r}$ becomes

$$\Delta\mathscr{C}_{I,*,r} = \mathscr{C}_{I,*,r} - \mathscr{C}_{I,\text{ref},r} = \beta_r x_{*,r}^2 = \mathscr{C}_{I,*,r} \tag{7.25}$$

where $\mathscr{C}_{I,*,r}$ is the contribution of the r-th variable in $\boldsymbol{x}_*$ to the monitoring statistic I. This shows that the deviation contribution in linear methods is equivalent to the contribution defined in Section 7.2, if the origin $\boldsymbol{x} = \mathbf{0}$ is chosen as the reference point $\boldsymbol{x}_{\text{ref}}$. Therefore, the formulation of deviation contributions is in line with the concept of contributions developed in previous works for linear methods whilst providing a way to calculate contributions in nonlinear methods.

7.3.4 Reference sample selection

Eqn (7.13) uses the mean value theorem to calculate difference between the monitoring statistics of $\boldsymbol{x}_*$, the sample being diagnosed and a reference sample $\boldsymbol{x}_{\text{ref}}$. This approach requires the first order derivative of $\frac{\partial I}{\partial \boldsymbol{x}}$ at a point $\boldsymbol{x} = \boldsymbol{x}_c$, which is unknown. Since Eqn (7.17) provides a way of estimating $\frac{\partial I}{\partial \boldsymbol{x}}$ at $\boldsymbol{x} = \boldsymbol{x}_c$ using $\boldsymbol{x}_*$ and $\boldsymbol{x}_{\text{ref}}$, the selection of $\boldsymbol{x}_{\text{ref}}$ will therefore influence the accuracy of the approximation, the deviation contributions of variables, and the diagnosis result thereafter. The first rule-of-thumb is that the reference sample $\boldsymbol{x}_{\text{ref}}$ should be a healthy sample so that its monitoring statistic I_{ref} is small. The reason is that the deviation contribution is defined for the difference between the monitoring statistic I_* of $\boldsymbol{x}_*$ and the monitoring statistic I_{ref} of $\boldsymbol{x}_{\text{ref}}$. If a reference sample is anomalous or not representative of the overall healthy dataset and I_{ref} is large, the deviation contribution will not be an accurate reflection of the contributions of variables to the fault existing in the process when $\boldsymbol{x}_*$ was sampled. Therefore, $\boldsymbol{x}_{\text{ref}}$ should be a healthy sample.

In linear methods, it is possible to use the origin, i.e. $\boldsymbol{x} = \mathbf{0}$, as the reference sample. The previous section proved that, when $\boldsymbol{x} = \mathbf{0}$ is used, the deviation contribution is equivalent to the contribution used by other researchers for linear methods. However, this may not be feasible for nonlinear methods because the origin does not necessarily belong to the healthy dataset when the dataset is nonlinear, especially if it is multimodal. This is also true even if the dataset has been normalized. The arc-shaped bivariate dataset in Fig. 5.3, Section 5.2.3, is an example. On the other hand, when the monitoring statistic I is monotonically

increasing, the accuracy of estimating the derivative $\frac{\partial I}{\partial x_r}$ at $\boldsymbol{x} = \boldsymbol{x}_c$ influences the accuracy of the approximation in Eqn (7.13). The accuracy of the approximation depends on the distance between $\boldsymbol{x}_*$ and $\boldsymbol{x}_{\text{ref}}$. The deviation contribution can approximate the difference between I_* and I_{ref} if $\boldsymbol{x}_*$ and $\boldsymbol{x}_{\text{ref}}$ are close to each other. Therefore, it is reasonable to select the nearest neighbour of $\boldsymbol{x}_*$ in the healthy dataset as the reference sample $\boldsymbol{x}_{\text{ref}}$:

$$\boldsymbol{x}_{\text{ref}} = \arg\min_{\boldsymbol{x}} \|\boldsymbol{x} - \boldsymbol{x}_*\|_2, \quad \text{s.t. } \boldsymbol{x} \in X \tag{7.26}$$

where X is the healthy dataset.

7.4 Deviation contribution for RBF-KPCA

This section provides the formula for calculating deviation contributions under the KPCA framework using RBF kernels. The formulation of RBF-KPCA can be found in Section 5.2.2. This section uses the notations used by Section 5.2.2 and derives the deviation contribution for the SPE obtained by RBF-KPCA because Chapter 5 has demonstrated that SPE is an appropriate monitoring statistic for RBF-KPCA.

Assuming a test sample is $\boldsymbol{x}_*$, the entry of the kernel vector K_* is as follows when the RBF kernel is used:

$$\begin{aligned} K_{i,*} &= \exp\left(\frac{1}{\delta^2}\left(\boldsymbol{x}_i - \boldsymbol{x}_*\right)^\top \left(\boldsymbol{x}_i - \boldsymbol{x}_*\right)\right) \\ &= \exp\left(\frac{1}{\delta^2}\sum_{r=1}^{m}\left(x_{i,r} - x_{*,r}\right)^2\right) \end{aligned} \tag{7.27}$$

where $\boldsymbol{x}_i$ is the i-th training sample. $x_{i,r}$ and $x_{*,r}$ are the r-th variable in $\boldsymbol{x}_i$ and $\boldsymbol{x}_*$, respectively.

Assuming L kernel PCs are retained after RBF-KPCA, the vector of kernel PCs obtained for $\boldsymbol{x}_*$ is denoted as $\boldsymbol{z}_* = \{z_*^{(1)}, \cdots, z_*^{(L)}\}$, the l-th kernel PC $z_*^{(l)}$ of $\boldsymbol{z}_*$ is calculated as

$$z_*^{(l)} = \sum_{i=1}^{n} \tilde{\alpha}_i^{(l)} \tilde{K}_{i,*} \tag{7.28}$$

where $\tilde{K}_{i,*}$ is the centered entry of the kernel vector K_* and $\tilde{\alpha}_i^{(l)}$ is the normalized entry of the l-th eigenvector $\boldsymbol{\alpha}$.

The SPE of $\boldsymbol{x}_*$ obtained by RBF-KPCA is

$$\text{SPE}_* = 1 - 2\bar{K}_* + \bar{K} - \boldsymbol{z}_*^\top \boldsymbol{z}_*. \tag{7.29}$$

Section 5.4.1 has proved that, when the test sample $\boldsymbol{x}_* \to \infty$, SPE will converge to a non-zero value $\text{SPE}_{*,\text{lim}}$.

Based on Eqn (7.18), the deviation contribution of the r-th process variable $x_{*,r}$ to SPE_* is defined as:

$$\Delta\mathscr{C}_{\text{SPE},r,*} = \frac{1}{2}\left(\left.\frac{\partial \text{SPE}_*}{\partial x_r}\right|_{\boldsymbol{x}=\boldsymbol{x}_*} + \left.\frac{\partial \text{SPE}_*}{\partial x_r}\right|_{\boldsymbol{x}=\boldsymbol{x}_{\text{ref}}}\right)(x_{*,r} - x_{\text{ref},r}) \tag{7.30}$$

The first order derivative can be calculated as:

$$\begin{aligned}\frac{\partial \text{SPE}_*}{\partial x_r} &= \frac{\partial\left(1 - 2\bar{K}_* + \bar{K} - \boldsymbol{z}_*^\top \boldsymbol{z}_*\right)}{\partial x_r} \\ &= -2\frac{\partial \bar{K}_*}{\partial x_r} - \frac{\partial \boldsymbol{z}_*^\top \boldsymbol{z}_*}{\partial x_r}\end{aligned} \tag{7.31}$$

where $\bar{K}_* = 1/n\sum_{i=1}^n K_{i,*}$ is the mean of the kernel vector $K_* = \{K_{1,*}, \cdots, K_{n,*}\}$. $\boldsymbol{z}_*^\top \boldsymbol{z}_* = \sum_{l=1}^L {z_*^{(l)}}^2$ is the sum of squares of the kernel PCs. $\bar{K}$ is the mean of all elements in the kernel matrix K obtained from the training data; thus $\bar{K}$ is a constant. The derivatives are then calculated individually. Following from Eqn (7.27), the derivative of $\bar{K}_*$:

$$\begin{aligned}\frac{\partial \bar{K}_*}{\partial x_r} &= \frac{1}{n}\sum_{i=1}^n \frac{\partial K_{i,*}}{\partial x_r} \\ &= -\frac{2}{n\delta^2}\sum_{i=1}^n K_{i,*}\left(x_{*,r} - x_{i,r}\right)\end{aligned} \tag{7.32}$$

From Eqn (7.28), the derivative of $\boldsymbol{z}_*^\top \boldsymbol{z}_*$ is:

$$\begin{aligned}\frac{\partial \boldsymbol{z}_*^\top \boldsymbol{z}_*}{\partial x_r} &= 2\sum_{l=1}^L z_*^{(l)} \frac{\partial z_*^{(l)}}{\partial x_r} \\ &= 2\sum_{l=1}^L z_*^{(l)} \sum_{i=1}^n \tilde{\alpha}_i^{(l)} \frac{\partial \tilde{K}_{i,*}}{\partial x_r} \\ &= 2\sum_{l=1}^L z_*^{(l)} \sum_{i=1}^n \tilde{\alpha}_i^{(l)} \left(\frac{\partial \tilde{K}_{i,*}}{\partial x_r} - \frac{1}{n}\sum_{i=1}^n \frac{\tilde{K}_{i,*}}{\partial x_r}\right) \\ &= 2\sum_{l=1}^L z_*^{(l)} \sum_{i=1}^n \frac{\tilde{\alpha}_i^{(l)}}{\delta^2} \left(K_{i,*}\left(x_{*,r} - x_{i,r}\right) + \frac{1}{n}\sum_{i=1}^n K_{i,*}\left(x_{*,r} - x_{i,r}\right)\right)\end{aligned} \tag{7.33}$$

The derivative of SPE_* with respect to x_r at $\boldsymbol{x}_*$ is

$$\left.\frac{\partial \mathrm{SPE}_*}{\partial x_r}\right|_{\boldsymbol{x}=\boldsymbol{x}_*} = \frac{4}{n\delta^2}\sum_{i=1}^{n} K_{i,*}\left(x_{*,r}-x_{i,r}\right) - \sum_{l=1}^{L} z_*^{(l)} \sum_{i=1}^{n} \frac{\tilde{\alpha}_i^{(l)}}{\delta^2}\left(K_{i,*}\left(x_{*,r}-x_{i,r}\right) + \frac{1}{n}\sum_{i=1}^{n} K_{i,*}\left(x_{*,r}-x_{i,r}\right)\right) \tag{7.34}$$

The derivative of SPE_* with respect to x_r at $\boldsymbol{x}_{\mathrm{ref}}$ is

$$\left.\frac{\partial \mathrm{SPE}_*}{\partial x_r}\right|_{\boldsymbol{x}=\boldsymbol{x}_{\mathrm{ref}}} = \frac{4}{n\delta^2}\sum_{i=1}^{n} K_{i,\mathrm{ref}}\left(x_{\mathrm{ref},r}-x_{i,r}\right) - \sum_{l=1}^{L} z_{\mathrm{ref}}^{(l)} \sum_{i=1}^{n} \frac{\tilde{\alpha}_i^{(l)}}{\delta^2}\left(K_{i,\mathrm{ref}}\left(x_{\mathrm{ref},r}-x_{i,r}\right) + \frac{1}{n}\sum_{i=1}^{n} K_{i,\mathrm{ref}}\left(x_{\mathrm{ref},r}-x_{i,r}\right)\right) \tag{7.35}$$

Therefore, the deviation contribution of the r-th process variable x_r to SPE_* is obtained by substituting Eqns (7.34) and (7.35) to Eqn (7.30).

7.5 Examples of fault diagnosis based on deviation contribution

This section presents two examples of using deviation contribution plots for fault diagnosis when RBF-KPCA is applied. The first example is a simulated example introduced by Dong and McAvoy (1996) and the second example uses the data from the PRONTO dataset when the air blockage fault was induced.

7.5.1 Fault diagnosis of a numerical example

The deviation contribution based fault diagnosis for KPCA is investigated in this section. The corresponding fault identification performance is compared with the performance obtained by contribution rate of a numerical simulation.

Simulation model

Eqn (7.36) shows a numerical model adopted from Dong and McAvoy (1996). This model is used here because it has been used by other researchers for testing the performance of

Table 7.1: Fault specification for numerical example

	Description	Type	Fault-related variable
Fault 1	Model mismatch	$x_3 = -u^3+4u^2+e_3$	x_3
Fault 2	Constant bias	$x_1 = u + e_1 + 0.5$, $x_2 = u^2-3u+e_2+1$	x_1, x_2
Fault 3	Excessive random variation	$e_2 \sim N(0, 0.1)$	x_2
Fault 4	Developing fault	$x_3 = -u^3+\beta u^2+e_3$	x_3

contribution based fault diagnosis using nonlinear methods (Lee et al., 2004a; Choi et al., 2005a).

$$\begin{aligned} x_1 &= u + e_1 \\ x_2 &= u^2 - 3u + e_2 \\ x_3 &= -u^3 + 3u^2 + e_3 \end{aligned} \tag{7.36}$$

where $u \in [0.01, 1]$ is a random variable with a uniform distribution and x_1, x_2 and x_2 are three measured variables. The white Gaussian noise e_i has a distribution $N(0, 0.01)$ for $i = 1, 2, 3$.

The training dataset contains 500 samples from the model specified by Eqn (7.36). Another 500 samples from this model are generated as the validation dataset. Table 7.1 summarizes the three types of faults that are introduced by changing the equations. Fault 1 to Fault 3 are persistent faults. In particular, Fault 2 results in changes in two variables x_1 and x_2. Fault 4 is a developing fault where the data deviates gradually from the model described by Eqn (7.36). This is implemented by introducing a parameter β to the third equation in Eqn (7.36). According to Eqn (7.36) , the correct value of β is 3; however β increases linearly from 3 to 8 when Fault 4 occurs. Five hundred samples are collected for each fault.

The developing fault is used to illustrate the behaviour of the deviation contribution when the monitoring statistic SPE has reached its limit in RBF-KPCA.

RBF-KPCA trains a monitoring model for fault detection using the training data and SPE, as defined in Eqn (7.29), is used as the monitoring statistic. The monitoring model is then applied to the validation data and the faulty data. The deviation contributions of the three variables x_1, x_2 and x_3 to SPE are calculated using the formula Eqn (7.30) derived in Section 7.4. The reference samples are selected according to Eqn (7.26) in Section 7.3.4.

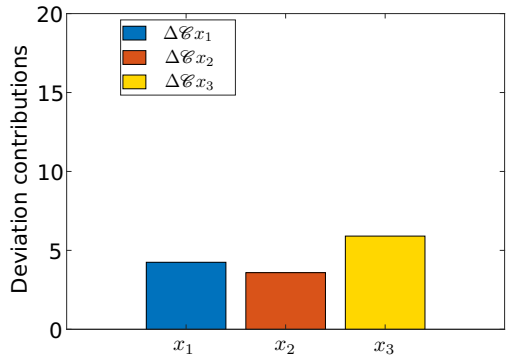
Results and discussion

Fig. 7.1 presents the contribution plots of the three process variables in the validation set and the first three faults in Table 7.1 from Fault 1 to Fault 3. The contributions visualized in Fig. 7.1 are the deviation contributions. In order to compare the magnitude of the contributions, the range of the vertical axes is the same for all contribution plots. The average deviation contribution over all samples can be used because the first three faults are persistent and the contribution of each variable should not change over time.

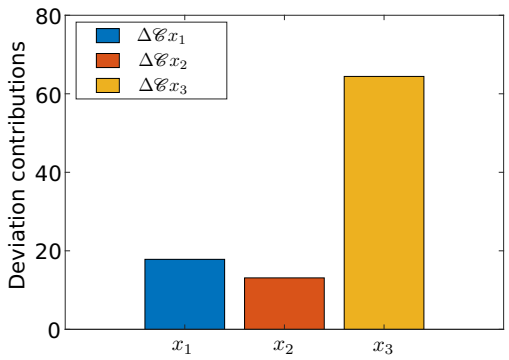
When calculating deviation contributions, it is necessary to notice that the nearest neighbour of a training sample will be the training sample itself according to Eqn (7.26). Therefore, the deviation contribution will always be zero for all training samples. Hence Fig. 7.1 does not present the deviation contributions for the training data. Fig. 7.1(a) demonstrates that the deviation contributions of all variables are small for the validation data because these data are healthy. Fig. 7.1(b) shows that variable x_3 has the largest average deviation contribution among all three variables when Fault 1 occurs. This can be justified because Fault 1 is the scenario where the equation for x_3 has changed. Fig. 7.1(c) shows that x_1 and x_2 both have large average deviation contribution. This is consistent with the description of Fault 2 such that both x_1 and x_2 have constant bias. Fig. 7.1(d) indicates that variable x_2 has the largest average deviation contribution, however the difference between the variables is not as significant as the other two examples. This can be explained by the nature of Fault 3. When Fault 3 occurs, the large variance of the noise in x_2 means that the data are likely to be faulty. On the other hand, the data from Fault 1 and Fault 2 are constantly faulty. Therefore, the impact of Fault 3 on x_2 will not be as significant as the impact of Fault 1 and Fault 2 on x_1 and x_3, respectively.

The result of Fault 4 is visualized in trends because the fault develops as the parameter β deviates from its correct value. Fig. 7.2(a) shows the SPE obtained for the data from Fault 4. Fig. 7.2(b) shows the trends of the deviation contributions of the process variables in this scenario. Fig. 7.2(a) shows that, apart from the fluctuation due to the noise in the data, SPE approaches to a constant after the 150th sample. Although the parameter β continues to deviate from the correct value, SPE does not demonstrate significant difference. Section 5.4.1 provided a reasoning for such behaviour of SPE.

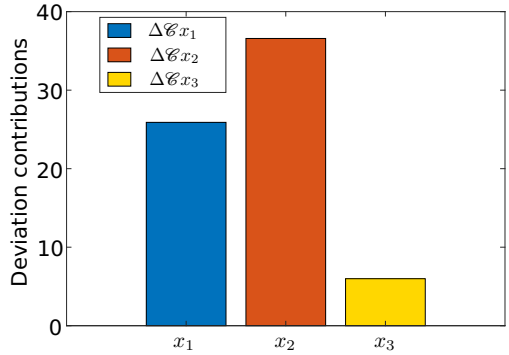
The deviation contribution of x_3 is significantly larger than the deviation contributions of x_1 and x_2. This is in line with the description of Fault 4 because this fault is inserted to



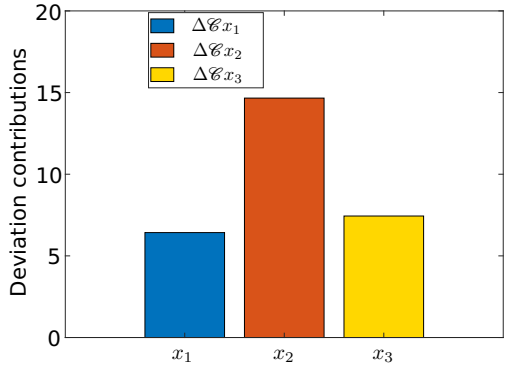
(a) Average deviation contributions of variables in validation set



(b) Average deviation contributions of variables in Fault 1



(c) Average deviation contributions of variables in Fault 2



(d) Average deviation contributions of variables in Fault 3

Figure 7.1: Average deviation contributions for tested datasets

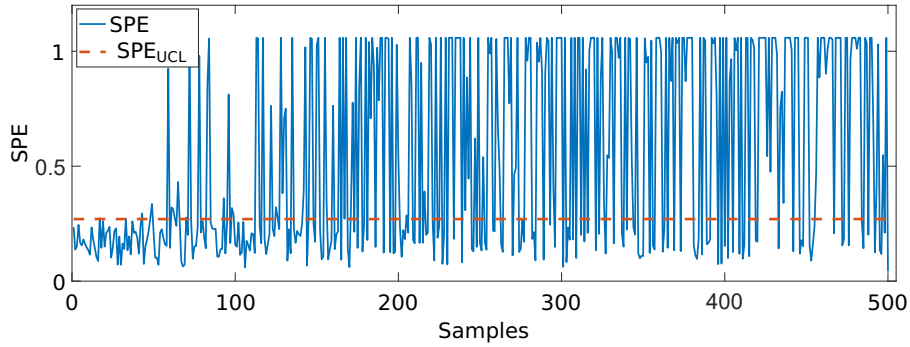
x_3. The deviation contribution of x_3 continues to increase as the fault develops even if SPE does not have such a behaviour. The reason is that Eqn (7.18) takes the derivative of two samples and the difference between the test sample and the reference sample into account. On the other hand, when SPE approach to a constant, i.e. SPE $\rightarrow$ SPE$_{*,\text{lim}}$, the first order derivative of SPE with respect to individual variables will approach to zero:

$$\left.\frac{\partial \text{SPE}_*}{\partial x_r}\right|_{\boldsymbol{x}=\boldsymbol{x}_*} \rightarrow 0$$

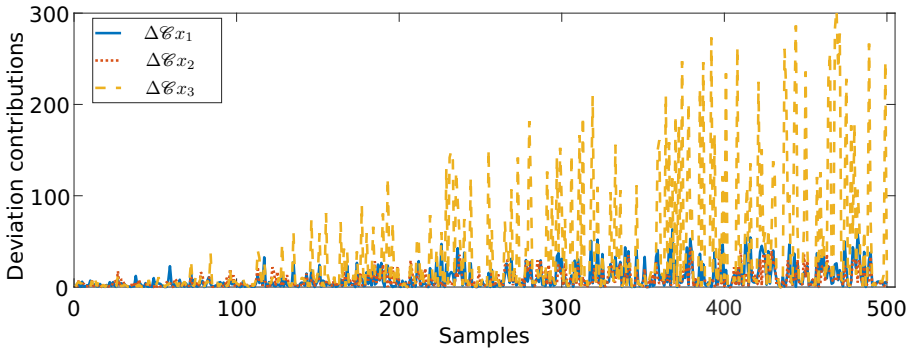
Nevertheless, the first order derivative of the reference sample

$$\left.\frac{\partial \text{SPE}_*}{\partial x_r}\right|_{\boldsymbol{x}=\boldsymbol{x}_{\text{ref}}}$$

is not zero because the reference sample is healthy. The difference between the test sample and the reference sample can therefore be taken into account in the deviation contribution.



(a) Trend plot of SPE in Fault 4



(b) Deviation contributions of variables in Fault 4

Figure 7.2: Trend plots of SPE and deviation contributions of Fault 4 of the example from Dong and McAvoy (1996)

To conclude, the deviation contribution is a way to calculate the contributions of process variables to monitoring statistics when kernel methods are applied, enabling contribution-based fault diagnosis. Moreover, the deviation contribution overcomes the issue such that SPE converges to a constant in RBF-KPCA and provides a way for fault diagnosis in RBF-KPCA.

7.5.2 Fault diagnosis of the PRONTO dataset

This section uses RBF-KPCA and the deviation contribution to diagnose the air blockage fault in the PRONTO dataset.

Data description

This example uses the same datasets as the example in Section 5.6.2. The training dataset contains healthy data from the operating condition with an inlet air flow rate of 120 $m^3\,h^{-1}$

Table 7.2: Process variables used in the example

Name	Description
FT305/2	Input air flow rate
PT312	Air delivery pressure
FT102/4	Input water flow rate
PT408	Riser top pressure
PT403	Top separator pressure
PT501	Three-phase separator pressure
LI502	Water-oil level in three-phase separator
LVC502	Water coalescer outlet valve opening

and an inlet water flow rate of 0.1 $\mathrm{kg\,s^{-1}}$. This dataset is used for training the monitoring model.

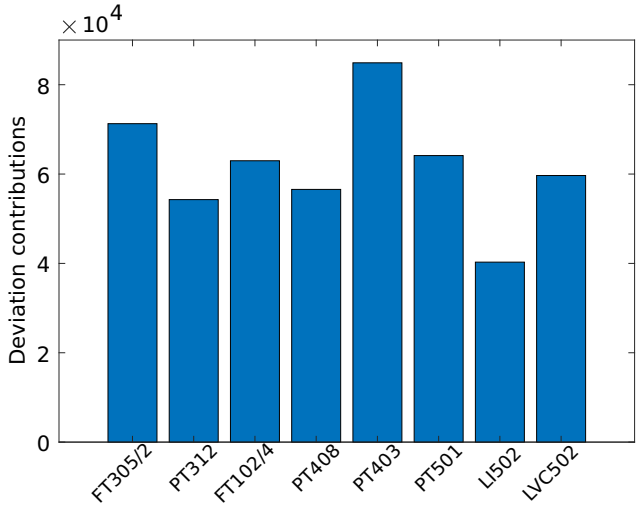
The P&ID of the test rig is shown in Fig. 4.1. The faulty dataset was obtained when the air blockage was induced to the process by gradually closing the manual valve V11. In Section 5.6.2, the sequence of valve opening for inducing the fault was presented in Fig. 5.10 and the faulty data were presented in Fig. 5.11.

The RBF-KPCA is applied to the two datasets and the SPE is used as the monitoring statistic. Based on the tuning strategy proposed in Section 5.5.2, Fig. 5.12 in Section 5.6.2 has determined that the optimal kernel width when using SPE is $\sigma^2 = 45$.

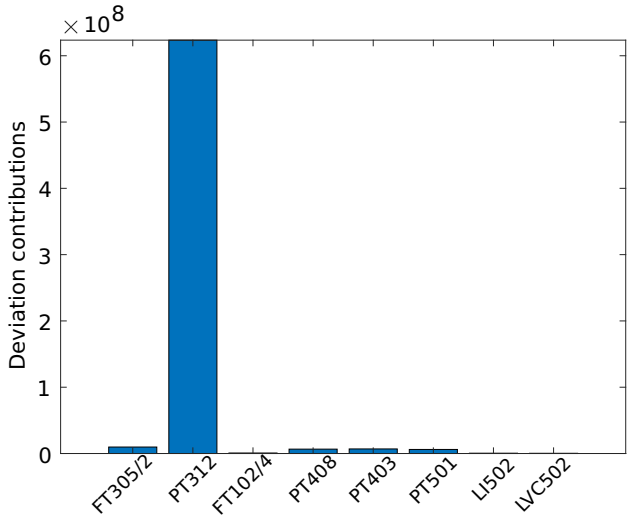
Table 7.2 summarizes the process variables used in this example. According to Fig. 4.1, the manual valve V11 is located on the input air line between the inlet air flow meter FT305/302 and the pressure sensor PT312. Therefore, the blockage here will have significant influence on the air delivery pressure measured at PT312.

Results and discussions

To validate the proposed deviation contribution plot, the deviation contributions of process variables to SPE are calculated using Eqn (7.30) for each tested sample. According to the result in Section 5.6.2, Fig. 5.13(a) shows that the air blockage fault is detected by SPE starting from around the 1600th sample of the test dataset. Therefore, Fig. 7.3 compares the bar plots of the accumulated deviation contributions of the eight process variables to



(a) Deviation contribution plot when no fault is detected

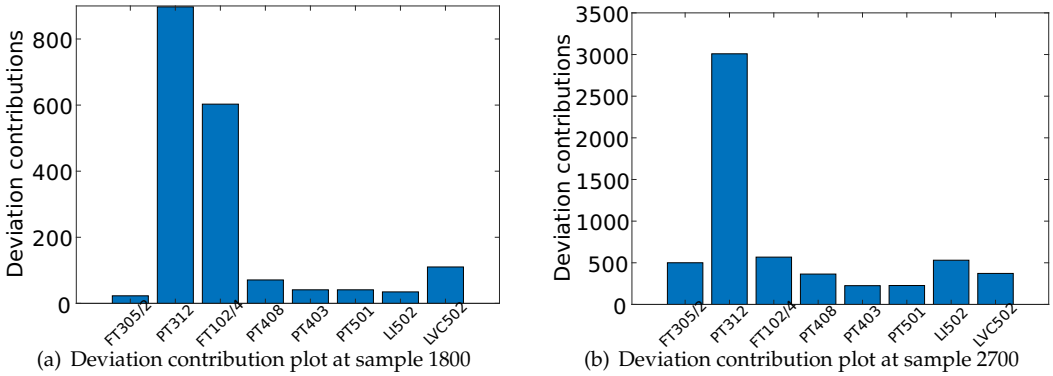


(b) Plot of accumulated deviation contributions for the air blockage fault

Figure 7.3: Deviation contribution plots of the PRONTO dataset

SPE for the period when the process is healthy (Fig. 7.3(a)) and the period when the fault is detected (Fig. 7.3(b)).

Fig. 7.3(a) shows that all the process variables have similar magnitudes of deviation contributions to the SPE when the process is healthy. Fig. 7.3(b) shows that PT312, the air delivery pressure, has the highest contribution to SPE when the fault occurs while the other process variables, including the input air flow rate (F305/2), do not have significant contributions. The fault can then be diagnosed by inspecting the P&ID of the process in Fig. 4.1. It can be seen from the P&ID that the input air flow first passes the flow rate sensors FT305/302 then goes through the manual valve V11, and finally the delivery pressure of the input air flow is measured by PT312. The input air flow is still healthy when measured by the flow



(a) Deviation contribution plot at sample 1800

(b) Deviation contribution plot at sample 2700

Figure 7.4: Deviation contribution plots of two samples in the faulty dataset

rate sensors FT305/302 and the pressure measurement of the same air flow is abnormal. Therefore, the likely diagnosis is that the manual valve V11 has faults because this valve is located on the way of the air flow from FT305/302 to PT312. Since the air blockage fault is induced by gradually reducing the valve opening of valve V11, the diagnosis result obtained with the assistance of the deviation contribution plot has supported in identifying the location of the blockage fault.

In addition, Fig. 7.4 compares the two contribution plots for the 1800th sample and the 2700th sample in the faulty dataset, respectively. It can be seen that the deviation contribution of PT312 shown in Fig. 7.4(b) is higher than the deviation contribution of PT312 in Fig. 7.4(a). This indicates that the contribution of PT312 gets more significant as the fault severity develops over time. It may be noticed that the magnitude of contributions in Fig. 7.4 is different from the magnitude of contributions in Fig. 7.3. This is because Fig. 7.4 shows the contributions at individual samples while Fig. 7.3 shows the accumulated contributions overtime.

The example using PRONTO dataset demonstrates how the proposed deviation contribution plots can be used in association with RBF-KPCA to diagnose faults in a real-life dataset.

7.6 Chapter summary

In this chapter, a novel deviation contribution is proposed for fault diagnosis when applying kernel methods. Contribution-based fault diagnosis is challenging when kernel methods are used because the contributions of variables to monitoring statistics cannot be explicitly defined. This deviation contribution solves this problem.

The chapter proposed the general formulation and derived the formula for calculating deviation contributions in RBF-KPCA. The formula was then applied to a numerical example and the PRONTO dataset. The numerical example shows that variables related to certain faults have larger deviation contributions to SPE than those variables that are not related to faults. Moreover, it was highlighted that the deviation contribution using proper reference samples can identify the development of contributions of variables even though SPE is no longer able to reflect the development of the fault. The example using the PRONTO dataset demonstrated that the proposed deviation contribution can identify the process variables which are most influential in the detection of an anomaly caused by a fault in the process.

Chapter 8

Critical evaluation and future research directions

This chapter provides a critical evaluation of the thesis. The methodology used to develop and validate the monitoring methods is reviewed and evaluated in this chapter. This is followed by a discussion on the novelty and achievements of the works presented in the thesis. In particular, the chapter describes how the research outcomes are relevant for industrial practice. The chapter ends with a discussion on the assumptions made in the thesis and outlines several extensions based on the work in the thesis and two directions for future research projects.

8.1 Evaluation of methodology

This section summarizes the methodology used for experiment design, data generation and validation of the proposed methods. The advantages of the methodology are evaluated by comparing the methodology with the current practices described in literature.

8.1.1 Experimental design in the PRONTO dataset

The objective of the experiment highlighted in Chapter 4 was to obtain a benchmark dataset for testing methods for fault detection and diagnosis (FDD) and data fusion. The dataset

includes data from heterogeneous sources including process measurements, alarm records, high frequency ultrasonic flow and pressure measurements, an operation log and video recordings. Data were recorded both with and without induced faults.

The experiment reported in Chapter 4 followed the current practice for creating an experimental benchmark dataset using the multiphase flow facility, as described by Ruiz-Cárcel et al. (2015). The current practices include the following points:

1. Several operating modes were tested;
2. Artificial faults were designed to develop gradually;
3. The experiment allowed time for the process to stabilize with each specific degree of fault severity;
4. The high frequency pressure and the ultrasonic measurements were synchronized;
5. Process measurements were taken continuously. High-frequency measurements were taken when the process reached its steady state or when the process had cyclic slugging behaviour;
6. Operation logs contain information about the experiment, such as the change of setpoints and the recording of high frequency data.

When testing the various operating modes, both healthy and faulty operating conditions of the process should be tested such that the dataset has healthy data and anomalous data. Particularly for the thesis, since the objective is to use the dataset for test multimodal process monitoring methods, the process needs to run in several operating modes.

Gradual development of the faults allowed time for the process to stabilize at various levels of fault severity and sufficient data can be collect at each level of severity. Process data collected from various levels of fault severity are useful for validation of monitoring methods. For example, the data from gradually developing faults in the PRONTO dataset are used in Chapter 5 to demonstrate whether a monitoring statistic reacts to a developing fault.

Since there are multiple sources of measurements, the experience from the experiment is to record the time stamps of the measurements from several sources so that these measurements can be synchronized. In particular, the high frequency and the ultrasonic measurements that are triggered manually should be synchronized in order to enhance the performance of the data-driven monitoring methods. It is also useful to record the changes made

by the researchers during the experiment. These records can be useful for data labelling when analysing the data.

In the thesis, the design and implementation of the experiment for obtaining the PRONTO dataset have considered the following aspects from the current practices to make the dataset reliable and sufficient for testing monitoring methods.

For the healthy modes, the experiment guaranteed that the process reached a steady state and enough data were collected before switching to a new mode. The process did not stabilize in slugging modes, but the experiment allowed enough time to record several cycles of the cyclic behaviour caused by slugging.

The logs are useful for understanding the sequence of actions taken in the experiment. They are also useful for labelling the process data because the process data were collected in one data file for a whole day, where several operating modes and fault scenarios were tested.

There are a few points to be considered in order to enhance the experiment and the dataset. Feedback from users of the benchmark dataset is that it would have been useful to have several datasets for the same operating mode. In hindsight, it would have been useful to repeat the experiment for all of the operating modes. It would also make the dataset more realistic to induce developing faults such that the severity does not have step changes. The fault may have been generated in an automated manner so that the step size can be reduced. However, the experiments had ended and the facility was no longer available, and it was not possible.

The current practices for generating experimental datasets have been implemented in the experiment for generating the PRONTO dataset. There has also be several adjustments when it is not possible to follow these practices due to the constraints of time and resources.

8.1.2 Evaluation of monitoring performance

The review in Section 2.4.1 established that false alarm rates and missed alarm rates are widely used for evaluating the performance of monitoring methods. The false alarm rate indicates how robust a monitoring method is to the variability in healthy data. The missed alarm rate indicates how sensitive a monitoring method is to anomalous data that are different from the healthy data. A conclusion from the review was that the false alarm rate

can be used for tuning the kernel width in kernel methods. Detection time is an alternative metric to the missed alarm rate when a fault develops over time. These monitoring metrics are widely accepted as the current practice of evaluating the performance of monitoring methods.

The approaches taken in the thesis conform with the appropriate metrics described above. It is usually possible to calculate false alarm rates in the training stage because healthy data are available. The work presented in this thesis uses the false alarm rate as a metric for performance evaluation. Missed alarm rates can only be calculated when there are known and labelled faulty cases, which may not always be available in the training stage. However, several of the simulated examples did use data from simulated faulty operating conditions. In the thesis, the work reported in Chapter 5 considers the extreme case where a test sample $\boldsymbol{x}_{\text{test}}$ approaches infinity and analyses the asymptotic behaviour of the monitoring methods. The analysis is useful because a test sample is likely to be anomalous if it is located extremely far away from the training data. A monitoring method should be able to detect this sample as anomalous. The asymptotic analysis gives a way of evaluating the sensitivity of monitoring models even if there is no anomalous data available.

It is also widely accepted by previous research in the field of process monitoring to use the false alarm rate for tuning the parameters of monitoring methods. The research in the thesis extends this current practice by proposing a tuning strategy for kernel widths which take both the false alarm rate and the influence of the kernel width into consideration.

In the example of using the PRONTO dataset reported in Chapter 5, a fault is detected when the monitoring statistic exceeds its control limit for a continuous sequence of 50 samples. The purpose is to minimize false alarms that are caused by the noise in the process measurements. Nevertheless, this may cause a time delay in detecting a fault due to the waiting time. Alternatively, the issue of false alarms may be addressed by filtering the monitoring statistics. This approach can reduce false alarms without introducing delays to fault detection and may facilitate the calculation of deviation contributions if the derivatives need to be calculated numerically.

The research presented in the thesis has attempted to use the current practices when possible, and shows how the existing practices can be extended.

8.1.3 Monte Carlo simulation in the simulated examples

False alarm rates and missed alarm rates need to be calculated in an accurate way to provide reliable evaluation of monitoring methods. This section discusses how the research in the thesis implements the current practice in literature to obtain these alarm rates such that the alarm rates accurately reflect the performance of monitoring models.

Metropolis and Ulam (1949) described that one may perform a large number of experiments in order to estimate the probability of observing certain outcomes of an experiment. This is the idea of *Monte Carlo simulation*. Monte Carlo simulation is widely adopted by researchers in the field of process monitoring because it can reduce the influence of the randomness in the data on the metrics of monitoring performance, such as the false alarm rate and the missed alarm rate. The calculated false and missed alarm rates when applying such method can reflect the sensitivity and the robustness of the monitoring method because these metrics represent how well a monitoring model can detect anomalous samples and how well it can account for healthy samples, respectively. In the thesis, Monte Carlo simulations were used for generating the simulated datasets and for calculating the alarm rates.

In Chapters 5 and 6, the kernel width is tuned according to the false alarm rates on the validation dataset. The range of kernel widths to be examined is first determined. For each value of the kernel width within this range, the healthy data available for training are randomly partitioned into a training set and a validation set. The monitoring model is trained using the training set and the alarm rate on the validation set represents how over-fitted the monitoring model is. The assumption is that the false alarm rates obtained by a model, which is not over-fitted, for the training data and for the validation data should be similar because the training data and the validation data are drawn independently from the same mathematical model. The partitioning of data is repeated several times and the result is averaged in order to reduce the variance of the false alarm rate. The missed alarm rate is calculated also by averaging the missed alarm rates of several Monte Carlo simulations. The use of Monte Carlo simulations gives an accurate estimation of the false alarm rate and the missed alarm rate so that the performance of the monitoring method can be evaluated.

The work in the thesis assumes that there is no autocorrelation between samples of process data. Therefore, the samples collected in one specific operating mode are considered to be generated independently from the same mathematical model. The randomness in the

samples is due to the disturbance and measurement noise, which are not autocorrelated. It is only possible to apply Monte Carlo simulation by randomly partitioning the data for training, validation, and test when there is no autocorrelation between the samples. When the autocorrelation between samples cannot be neglected, the experiment may need to be repeated several times in order to generate training and validation data as suggested in Section 8.1.1. The Appendix C also describes a numerical example where data are generated for validating a method developed for autocorrelated data.

The research in the thesis has implemented the current practices for constructing datasets when possible, and has made adjustments when necessary. Moreover, the research extends the current practice by applying Monte Carlo simulation to training data for tuning the kernel width and training the monitoring model.

8.2 Novelty and achievements

This section summarizes the novelty and achievements of the research outcomes presented in the thesis. The research outcomes are compared against the objectives set at the beginning of the thesis. The section highlights how the research outcomes are relevant for industrial practice. The section also presents several examples demonstrate that the outcomes have been recognized by the research community.

8.2.1 Industrial relevance

According to Section 2.6.1, the following criteria are important when developing and evaluating monitoring methods.

1. The monitoring statistic should increase monotonically as the fault severity develops;
2. The monitoring statistic should be of the same magnitude for various operating modes if the process is healthy.

These criteria are important from the perspective of operators, who will read the charts with monitoring statistics generated by monitoring systems and who will make decisions related to process operations based on the charts. Thus these two criteria can help the delivery of monitoring results to end-users and can make monitoring methods useful.

The research in the thesis is an example of the theoretical development of data-driven monitoring methods that can provide useful results to support end-users in decision-making. The research also shows that the considerations originating from industrial practice can lead to novel development in the theory of data-driven methods.

8.2.2 Achievements of the technical objectives

Table 3.5 in Chapter 3 summarized the technical tasks of the research reported in the thesis. The technical chapters have addressed these tasks as follows:

1. Chapter 5 investigated the tuning of kernel widths and the behaviour of monitoring statistics in kernel principal component analysis with radial basis function kernels for fault detection. The chapter proposed a tuning strategy for the kernel width and explained for why SPE is qualified as a monitoring statistic while T^2 is inadequate;
2. Chapter 6 proposed the new non-stationary discrete convolution kernel for building monitoring models for multimodal process monitoring. The chapter demonstrated that the new kernel outperforms the radial basis function kernel in building monitoring models for multimodal data;
3. Chapter 6 also proposed an on-line framework for implementing kernel methods with this new kernel. This framework incorporates the Dirichlet process with the new non-stationary discrete convolution kernel and enables on-line update of monitoring models when there are new data available from process operations;
4. Chapter 7 proposed the concept of deviation contributions for identifying influential variables when kernel methods are applied for fault detection. The new definition of contributions is compatible with kernel methods and provides a way for fault diagnosis.

The technical chapters have accomplished the tasks and achieved the research objectives of the thesis. All of the research outcomes are novel contributions of the thesis.

8.2.3 Recognition by the research community

In addition to the publications and conference presentations listed in Section 1.3, the research outcomes of the thesis have been widely recognized by the research community.

The PRONTO dataset was published in Zenodo on the 16th of May, 2019. Stief et al. (2019c), the paper describing the dataset, has been in the top five most downloaded papers of Journal of Process Control since it was published in June 2019. This indicates how popular the dataset has been in the community of researchers.

The paper entitled "Statistical monitoring of processes with multiple operating modes" was presented in the keynote session of data analytics and machine learning in the 12th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems (DYCOPS2019). As the first and corresponding author of the paper, the author of the thesis was one of the three finalists for the Young Author Award.

The paper entitled "Contribution plots-based fault diagnosis of a multiphase flow facility with PCA-enhanced canonical variate analysis" received the Best Student Paper Award in the 23rd International Conference on Automation and Computing (ICAC'17).

8.3 Extensions of the research outcomes

This section gives suggestions for how the research outcomes in the thesis can be extended further. The proposal for new directions for future research will be introduced in the next section.

8.3.1 Labelling process data and reference to expert knowledge

The research outcomes in the thesis depend on prior knowledge of process data. The DP-GMM method for clustering assumes that each data cluster follows a multivariate Gaussian distribution. The performance of this method should be assessed on multimodal datasets where the relationship between process variables is nonlinear in some of the operating modes. The influence of nonlinearity is not significant in the results presented in the thesis because the assumption is that data from different operating modes are significantly different, making it possible for DP-GMM to cluster these data. However, such an assumption may not always be true in practice when the operating modes are not significantly different from each other. A useful extension of the work of the thesis will be to explore other clustering methods that are suitable for multimodal data with other characteristics in order to prepare the data for the new non-stationary discrete convolution kernel. Clustering

methods, such as spectral clustering (Von Luxburg, 2007), are recommended because these methods can cope with complex relationships existing in the data.

8.3.2 Monitoring methods towards autonomous operations

Another useful extension would be to further automate the on-line monitoring framework presented in Section 6.6 for the purposes of adopting the NSDC-KPCA method for monitoring and updating the monitoring model during on-line implementation. Gamer et al. (2020) proposed a hierarchy of autonomy in industrial plants. An increased level of process autonomy can enhance the production by allowing processes to react to disruptions in real-time and to handle complexities in the processes whilst requiring less human intervention especially in risky environments. According to Gamer et al. (2020), level 0 of autonomy means that there is no autonomy and humans are in control. Level 5, the highest level of autonomy means that the operation is autonomous in all situations and humans may not be needed in any situation. The work in the thesis also contributes to improving the autonomy of processes. The on-line monitoring framework proposed in Chapter 6 belongs to level 2, where autonomy is limited to occasional situations and operators need to confirm the alarms flagged by the monitoring system. The next level of autonomy aims to achieve full control by the system in certain situations while operators supervise. The research of the next level of autonomous monitoring systems involves a variety of topics. For example, it would be useful for the methods to build monitoring models that can interpolate between operating modes so that new healthy modes can be automatically acknowledged.

Another example is the collaboration of monitoring systems, control systems and alarm systems. When process operators switch the process to a new operating mode, the alarm system may temporarily suppress the alarms triggered due to the transition. The control system can record the changes of set-points in the process. The monitoring system can then take the actions of the alarm system and the control system into consideration. Data collected from heterogeneous sources may have multiple sampling rates and may be stored in several formats with different data acquisition policies. This motivates future research on fusion of data from heterogeneous sources.

Therefore, the following extensions are recommended:

1. Applying other clustering approaches so that non-Gaussian data can be accounted for;

2. Further automating monitoring systems such as building monitoring models that can interpolate between operating modes;

3. Incorporating data from heterogeneous sources.

8.3.3 Fault diagnosis and fault prognosis in multimodal processes

The work reported in Chapter 7 extended the contributions of process variables to kernel methods. One of the objectives of fault diagnosis is to help operators to locate the fault so that operators can conduct inspection and maintenance on the right equipment. Therefore, it is necessary to apply the proposed contribution plots to real-life datasets because it is possible to compare the influential variables identified by their contributions against the variables that are related to the fault in reality. The example of the PRONTO dataset in Section 7.5.2 shows that the proposed contribution plots can work for real-life data and the next step will be testing the performance of the contribution plots when the dataset has multiple operating modes. It will then be possible to assess whether the contribution-based method can identify the correct variables that should be inspected by operators.

The work reported in Chapters 5 and 6 considered the monotonicity property of the monitoring statistics with respect to the development of faults. This is particularly important for fault prognosis because a monitoring statistic that increases monotonically with respect to faults can be a measure of the magnitude of the fault. Such a monitoring statistic can facilitate the prediction of the magnitude of the fault.

8.4 Directions for future research

This section presents two directions that can be considered for future research projects. The first direction is to configure other kernel methods so that these methods can generate useful results for end-users. The second direction is for future research in kernel-based monitoring of nonlinear dynamic processes.

8.4.1 Future directions in development and configuration of other kernel methods

The analysis in Chapter 5 shows that, in kernel principal component analysis, the kernel mapping makes the behaviour of monitoring statistics different from the behaviour of the same statistics when PCA is applied. Therefore, it is desirable to apply the analysis to other kernel methods in order to discover the behaviour of the monitoring statistics in these methods.

This direction of future research is suitable as a roadmap for appropriate development and configuration of kernel methods in general, using the practical considerations in the thesis as a guideline. Similarly, the research outcomes of the thesis provide a way from a theoretical perspective to analyse, to develop, and to implement kernel methods.

This roadmap should include the following:

1. Analysis of other kernel methods applied to process monitoring, such as kernel partial least square and support vector machine, to understand the influence of tuning and the behaviour of monitoring statistics;
2. Development and the configuration of these kernel methods so that these methods can deliver useful monitoring results;
3. Summary of practical considerations in other areas of application of data-driven methods, such as process modelling and prediction, in order to make these methods also useful for process operators;
4. Specification of the configuration of kernel methods in modelling and prediction so that these methods can deliver useful results to process operators.

8.4.2 Future directions in nonlinear dynamic process monitoring

The KPCA framework using RBF kernels and NSDC kernels assumes the process to be static. This indicates that a sample of the process variables at a particular time is independent from the past samples. For example, the illustrative examples in Chapters 5 and 6 are algebraic models and the data generated using these models are not autocorrelated.

However, real-life industrial processes often have large time constants when compared to the sampling interval of SCADA systems, resulting in process data with autocorrelation. Moreover, changes in the operating mode and faults in the process can lead to changes in the autocorrelation relationship in process data. Hence the objective of *dynamic process monitoring* is to build monitoring models for the autocorrelations existing in the healthy operations so that changes in the autocorrelations can be detected. Models that capture the autocorrelation between samples may achieve an earlier fault detection than those algebraic models. For instance, the numerical example presented in Fig. 6.10, Section 6.5.1, has two fault sequences that are overlapped partially with the training data. If the fault sequences are time series, the beginnings of the sequences are located in the middle of a healthy mode and NSDC-KPCA can only detect the fault when it develops to a stage that the samples are no longer overlapped with the healthy data. However, the autocorrelation of the fault sequences is significantly different from the autocorrelation of the healthy data. Hence if a dynamic approach is applied, even the beginning of the fault sequence can be detected.

A future research direction is to adapt kernel methods. Models based on these kernel methods should be able to describe the autocorrelation existing in multiple operating modes of the process and to detect the changes in the autocorrelation caused by faults. Furthermore, the new kernel methods for monitoring nonlinear dynamic processes should also satisfy the practical considerations discussed in the thesis.

The following directions can be considered in the future research of kernel-based monitoring of nonlinear dynamic processes:

1. Development of a new kernel for kernel-based monitoring of nonlinear dynamic processes;
2. Proper implementation of the new kernel by understanding the influence of tuning and the behaviour of monitoring statistics when using this new kernel;
3. Validation of the new kernel according to the practical considerations proposed in the thesis;
4. Comparison of the new kernel and existing kernel methods for nonlinear dynamic process monitoring.

Appendix C presents a preliminary study of the formulation of dynamic kernels. That study extends applies an approach which is similar to the approach in Chapter 6 to adapt the RBF kernel to a dynamic version.

8.5 Chapter summary

The chapter evaluated the results in the thesis from several perspectives. It first evaluated the methodology used for experimental design and validation of monitoring methods. The chapter then highlighted the novelty and the achievements, especially the industrial relevance, of the research reported in the thesis.

The assumptions made by the thesis were examined and discussed. Adjusting these assumptions leads to several directions for future research. These include several extensions of the research outcomes and two main pieces of future work. The chapter then outlined the directions for the configuration of other kernel methods and for the kernel-based nonlinear dynamic process monitoring.

Chapter 9

Conclusions

This chapter reviews the research contained in each chapter. This is followed by a summary of the findings that answer the research questions and meet the research objectives. The chapter then summarizes the directions for future research and gives final comments.

9.1 Summary of the chapters

Chapter 1 introduced the thesis and discussed the topic of process monitoring for supporting operators in making decisions regarding the operation of a plant. The chapter discussed the existence of multiple operating modes in processes. The aim of the research is to develop data-driven monitoring approaches which can deal effectively with the challenge posed by varying production regimes and to provide better decision-support to operators by achieving a more accurate evaluation of the process performance. The chapter outlined the questions to be addressed by the research presented in the thesis.

The research problem was formulated in Chapter 2. The research problem is to use process data to generate monitoring models for processes with multiple operating modes. In particular, several practical requirements were discussed to guide the design of the monitoring methods so that they might deliver useful results to end-users for decision support. The chapter also demonstrated the challenges in process monitoring posed by multiple operating modes.

The literature survey in Chapter 3 presented an examination of the published state-of-the-art in process monitoring. The review covered the theoretical formulations and the methods to deliver the results of the monitoring methods. Multivariate statistical process monitoring methods are considered in the thesis because process data are often recorded from processes with multiple operating modes during typical operation. The survey of kernel methods in Chapter 3 identified the open questions in applying kernel methods to multimodal process monitoring. Based on the findings of the literature review, the technical objectives of the thesis listed in Table 3.5 were proposed in order to answer the research questions and meet the practical requirements.

Chapter 4 presented the PRONTO benchmark dataset which was collected from an industrial-scale plant with multiple operating modes. Observations from the process data obtained in this dataset were presented and summarized. The technical chapters use the PRONTO dataset to validate the methods proposed.

Chapters 5 to 7 addressed the technical objectives identified in Table 3.5. Chapter 5 first investigated the behaviour of monitoring statistics and the tuning of the kernel width when Radial Basis Function (RBF) kernels are used. This enables the proper usage of RBF kernels in kernel methods. Chapter 6 first illustrated that the radial basis function kernel has limitations in modelling multimodal data, which led to the proposal of the new Non-stationary Discrete Convolution (NSDC) kernel, which can achieve better monitoring performance than RBF kernels when applied to multimodal process monitoring. The tuning and the findings about monitoring statistics in Chapter 5 also apply to the new kernel. Chapter 7 utilized the behaviour of monitoring statistics achieved in previous chapters and proposed deviation contribution plots in order to highlight influential variables related to faults. This new definition of contributions further made it possible to identify the influential variables after a fault is detected by the kernel methods.

Chapter 8 evaluated the methodology and the research outcomes of the thesis. It also identifies directions for future research. In particular, Chapter 8 presented two major directions for future research, which are the configuration of kernel methods applied to other use cases and the development of kernel methods for nonlinear dynamic process monitoring. Appendix C presents an example of dynamic adaptation of the radial basis function kernel and the preliminary results achieved by the dynamic kernel for dynamic process monitoring.

Table 9.1: Research questions of the thesis

No.	Research questions
1	What information does an operator need from a monitoring method in order to make decisions on the operation of a process?
2	Are kernel methods suitable for monitoring processes with multiple operating modes?
3	How should kernel methods be configured so that they perform properly when applied to monitoring processes with multiple operating modes?
4	How can kernel methods form part of a monitoring system?

9.2 Answers to the research questions

This section discusses how the research and the chapters in the thesis have answered the research questions proposed at the beginning of the thesis. The research questions formulated in Chapter 1 are reviewed in Table 9.1.

9.2.1 Practical considerations for multimodal process monitoring

Chapter 2 answered the first research question in Table 9.1, namely what information an operator needs from a monitoring method. Section 2.4 summarized the considerations of practical application of monitoring methods. The considerations suggested that, in order to be useful for operators, monitoring methods are expected to have the following behaviour:

1. Low false alarm rates and low missed alarm rates;
2. Interpretable monitoring results generated by the monitoring methods;
3. Robust monitoring methods to new unseen operating modes;
4. Reliable detection of faults in new operating modes.

These considerations were then used for guiding the development of kernel methods for multimodal process monitoring. The PRONTO benchmark dataset presented in Chapter 4 is a real-life example which verifies the expected behaviour outlined by Chapter 2 from the practical perspective. Later in the thesis, this dataset was also used to determine if the proposed kernel methods can provide the required useful information for an operator.

9.2.2 Applicability of kernel methods in multimodal process monitoring

The second research question in Table 9.1 regarding whether kernel methods are suitable for multimodal processes was answered in several steps. Section 2.6 demonstrated the following challenges in fault detection and diagnosis of mutlimodal processes:

1. To devise a monitoring statistic of which the same magnitude indicates the same level of fault severity across multiple modes;
2. To make the contributions of process variables from multiple modes comparable with each other when faults occur.

The literature review in Chapter 3 concluded that kernel methods have been applied to multimodal process monitoring. However, the following open questions in kernel methods needed to be clarified in order to address the challenges discussed in Chapter 2:

1. What is the optimal method for tuning kernel parameters?
2. Is there a kernel suitable for multimodal process monitoring?
3. How do monitoring statistics behave when kernel methods are applied?
4. How to apply contribution-based fault diagnosis in kernel methods?

In Section 6.2, it has been demonstrated that the radial basis function kernel has its limitations in generating monitoring models for multimodal data and new kernels need to be developed for this purpose.

Therefore, the answer to this question is that kernel methods can be used for monitoring multimodal processes; however the standard kernels, such as the radial basis function kernel, have limitations in coping with multiple operating modes.

9.2.3 Development and configuration of kernel methods for multimodal process monitoring

The third research question in Table 9.1 is how kernel methods should be configured so that they perform properly. Section 3.5.2 set up specific technical objectives for addressing this question. By incorporating the practical considerations, the challenges in multimodal

Table 9.2: Technical objectives of the thesis

No.	Technical objectives of the thesis
1	A tuning strategy for parameters of kernels
2	A new kernel for multimodal data
3	An on-line monitoring framework for incorporating new modes
4	A new type of contribution plot for fault diagnosis in kernel methods

process monitoring and the open questions in kernel methods, Section 3.5.2 stated that the following technical objectives need to be achieved in order to make kernel methods perform properly, as summarized in Table 9.2.

The technical work in the thesis answers the third research question by achieving the first two technical objectives in Table 9.2, which are the tuning strategy for kernel methods and a new kernel for multimodal data.

In Chapter 5, the main findings on the topic of the behaviour of monitoring statistics were that the squared prediction error is a better monitoring statistic for kernel methods, and that T^2 is inappropriate as a monitoring statistic. The third finding on the topic of the tuning of kernels was a tuning strategy for the kernel widths. These findings together achieved the first technical objective in Table 3.5 because they provide a way to tune the kernel width and to choose the monitoring statistic when developing kernel methods.

A finding of Chapter 6 was the non-stationary discrete convolution kernel for multimodal process monitoring. The results show that the new kernel can achieve better monitoring performance than several existing methods for multimodal process monitoring. Therefore, the work in Chapter 6 achieved the second technical objective presented in Table 9.2 because the new kernel can build good monitoring models for multimodal processes.

By achieving the first two technical objectives in Table 9.2, the research in the thesis has demonstrated how kernel methods should be developed so that these methods can perform now perform properly for multimodal processes.

9.2.4 Implementation of kernel methods in monitoring systems

The last research question in Table 9.1 regarding how kernel methods can form part of a process monitoring system is answered by achieving the third and the fourth technical objectives in Table 9.2. The third technical objective focuses on an on-line monitoring framework and the fourth technical objective is to develop contribution-based methods for fault diagnosis in kernel methods. Both of the objectives are useful if one wants to implement kernel methods as part of a monitoring system.

The on-line monitoring framework presented in Section 6.6 provides a method for implementing the new non-stationary discrete convolution kernel in an on-line process monitoring system. This framework also enables fault detection in new operating modes that emerge in on-line operation and that are not in the historical data used for modelling. The deviation contribution plot presented in Chapter 7 is a way to diagnose a fault when kernel methods are applied.

The main finding of Chapter 7 is the new concept of deviation contributions, which quantifies the contribution of process variables to monitoring statistics when using kernel methods. Therefore, the work on deviation contributions achieved the last technical objective because the deviation contribution makes it possible to identify influential variables and to diagnose the fault when kernel methods are used for monitoring. This approach can suggest the process variables that are connected to the fault occurrence, making the approach useful as part of a monitoring system because the results can support the decision of an operator.

9.3 Future Work

Whilst performing the research described in this thesis, a number of potential further directions for investigation were identified. These potential future research directions are summarized in Table 9.3. The suggestions of future work are divided into the extensions of the research outcomes reported in the thesis and the directions for future research projects.

There are possibilities for extending the research outcomes reported in the thesis and two extensions were suggested in Section 8.3. The on-line framework proposed in Section 6.6

Table 9.3: Summary of future works

Type	Future work
Extensions	Labelling process data
	Using real-life datasets to test the ability of the deviation contribution plots in identifying fault-related variables
Future research	Development and configuration of other kernel methods
	Kernel-based monitoring for dynamic processes

uses the Dirichlet process Gaussian mixture model to cluster the multimodal data. Other clustering methods can be applied to cope with data with other properties, such as non-Gaussianity, which may often be found in real-life data.

A direction of future research is to develop and configure other kernel methods in process monitoring and other use cases. Section 8.4.1 discussed that, in addition to the kernel principal component analysis considered in the thesis, it is useful to properly develop and configure other kernel methods applied to process monitoring such that these methods can provide useful information for operators. Moreover, it will be useful to understand the preference of end-users in other use cases of kernel methods, such as process modelling and prediction, and to properly develop and configure the kernel methods for these use cases.

Another direction is kernel-based monitoring of nonlinear dynamic processes. As suggested in Section 8.4.2, a new kernel needs to be designed such that it can capture both the nonlinear behaviour and the dynamic behaviour in process data. Monitoring models can be trained using the new kernel and faults that cause changes to process dynamics or can be detected. The new kernel also needs to be configured properly and verified using simulated and real-life datasets.

9.4 Final comments

The research described in the thesis has developed data-driven process monitoring methods which can cope with the challenges posed by multiple, varying operating modes. These results extend the field of process monitoring. The monitoring methods address the practical considerations and deliver results which have been identified as being valuable to operators.

The research in the thesis also contributes to the theory of kernel methods for monitoring processes with multiple operating modes. The findings reported in the thesis, including the behaviour of monitoring statistics achieved by kernel methods and the tuning of kernel parameters, provide important insights for researchers in the field of kernel methods. The novel kernel reported in the thesis has been published and gives a significant advance in monitoring for processes with multiple operating modes.

List of references

Afzal, M. S., Tan, W., and Chen, T. (2017). Process monitoring for multimodal processes with mode-reachability constraints. *IEEE Transactions on Industrial Electronics*, 64(5):4325–4335.

Ahmed, U., Ha, D., Shin, S., Shaukat, N., Zahid, U., and Han, C. (2017). Estimation of disturbance propagation path using principal component analysis (PCA) and multivariate Granger causality (MVGC) techniques. *Industrial & Engineering Chemistry Research*, 56(25):7260–7272.

Alcala, C. F. and Qin, S. J. (2009). Reconstruction-based contribution for process monitoring. *Automatica*, 45(7):1593–1600.

Alcala, C. F. and Qin, S. J. (2010). Reconstruction-based contribution for process monitoring with kernel principal component analysis. *Industrial & Engineering Chemistry Research*, 49(17):7849–7857.

Amari, S.-i. and Wu, S. (1999). Improving support vector machine classifiers by modifying kernel functions. *Neural Networks*, 12(6):783–789.

Amin, M. T., Imtiaz, S., and Khan, F. (2018). Process system fault detection and diagnosis using a hybrid technique. *Chemical Engineering Science*, 189:191–211.

Barnard, G. A. (1959). Control charts and stochastic processes. *Journal of the Royal Statistical Society: Series B* (*Methodological*), 21(2):239–257.

Bauer, M., Cox, J. W., Caveness, M. H., Downs, J. J., and Thornhill, N. F. (2007). Finding the direction of disturbance propagation in a chemical process using transfer entropy. *IEEE Transactions on Control Systems Technology*, 15(1):12–21.

Bayes, T. (1763). An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society of London*, 53:370–418.

Borror, C. M., Montgomery, D. C., and Runger, G. C. (1999). Robustness of the EWMA control chart to non-normality. *Journal of Quality Technology*, 31(3):309–316.

Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*. ACM. 144–152.

Box, G. E., Hunter, W. G., Hunter, J. S., et al. (1978). *Statistics for Experimenters*. John Wiley & Sons, New York.

BSI (2010). *BS EN 13306:2010. Maintenance – Maintenance terminology*. Standard, British Standards Institution.

BSI (2012). *BS ISO 13372:2012. Condition monitoring and diagnostics of machines – Vocabulary*. Standard, British Standards Institution.

BSI (2015). *BS EN 62682:2015. Management of alarms systems for the process industries*. Standard, British Standards Institution.

Cai, B., Huang, L., and Xie, M. (2017). Bayesian networks in fault diagnosis. *IEEE Transactions on Industrial Informatics*, 13(5):2227–2240.

Cai, L., Tian, X., and Chen, S. (2017). Monitoring nonlinear and non-Gaussian processes using Gaussian mixture model-based weighted kernel independent component analysis. *IEEE Transactions on Neural Networks and Learning Systems*, 28(1):122–135.

Chakour, C., Benyounes, A., and Boudiaf, M. (2018). Diagnosis of uncertain nonlinear systems using interval kernel principal components analysis: Application to a weather station. *ISA Transactions*, 83:126–141.

Chang, Y., Yang, W., and Zhao, D. (2018). Energy efficiency and emission testing for connected and automated vehicles using real-world driving data. arXiv: 1805.07643 [cs.OH].

Chen, B., Liu, H., and Bao, Z. (2008). Optimizing the data-dependent kernel under a unified kernel optimization framework. *Pattern Recognition*, 41(6):2107–2119.

Chen, J. and Liu, J. (1999). Mixture principal component analysis models for process monitoring. *Industrial & Engineering Chemistry Research*, 38(4):1478–1488.

Chen, J. and Liu, K.-C. (2002). On-line batch process monitoring using dynamic PCA and dynamic PLS models. *Chemical Engineering Science*, 57(1):63–75.

Chen, T. (2010). On reducing false alarms in multivariate statistical process control. *Chemical Engineering Research and Design*, 88(4):430–436.

Chen, T. and Sun, Y. (2009). Probabilistic contribution analysis for statistical process monitoring: A missing variable approach. *Control Engineering Practice*, 17(4):469–477.

Chen, T. and Zhang, J. (2010). On-line multivariate statistical monitoring of batch processes using Gaussian mixture model. *Computers & Chemical Engineering*, 34(4):500–507.

Chiang, L. H., Russell, E. L., and Braatz, R. D. (2000a). *Fault Detection and Diagnosis in Industrial Systems*. Springer Science & Business Media.

Chiang, L. H., Russell, E. L., and Braatz, R. D. (2000b). Fault diagnosis in chemical processes using Fisher discriminant analysis, discriminant partial least squares, and principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 50(2):243–252.

Cho, J.-H., Lee, J.-M., Choi, S. W., Lee, D., and Lee, I.-B. (2005). Fault identification for process monitoring using kernel principal component analysis. *Chemical Engineering Science*, 60(1):279–288.

Choi, S. W., Lee, C., Lee, J.-M., Park, J. H., and Lee, I.-B. (2005a). Fault detection and identification of nonlinear processes based on kernel PCA. *Chemometrics and Intelligent Laboratory Systems*, 75(1):55–67.

Choi, S. W. and Lee, I.-B. (2004). Nonlinear dynamic process monitoring based on dynamic kernel PCA. *Chemical Engineering Science*, 59(24):5897–5908.

Choi, S. W., Martin, E. B., Morris, A. J., and Lee, I.-B. (2005b). Fault detection based on a maximum-likelihood Principal Component Analysis (PCA) mixture. *Industrial & Engineering Chemistry Research*, 44(7):2316–2327.

Dash, S., Rengaswamy, R., and Venkatasubramanian, V. (2003). Fuzzy-logic based trend classification for fault diagnosis of chemical processes. *Computers & Chemical Engineering*, 27(3):347–362.

Dempster, A. P., Laird, N. M., and B., R. D. (1977). Maximum likelihood from imcomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39.

Deng, X., Tian, X., Chen, S., and Harris, C. J. (2018). Nonlinear process fault diagnosis based on serial principal component analysis. *IEEE Transactions on Neural Networks and Learning Systems*, 29(3):560–572.

Deng, X. and Wang, L. (2018). Modified kernel principal component analysis using double-weighted local outlier factor and its application to nonlinear process monitoring. *ISA Transactions*, 72:218–228.

Deng, X., Zhong, N., and Wang, L. (2017). Nonlinear multimode industrial process fault detection using modified kernel principal component analysis. *IEEE Access*, 5:23121–23132.

Ding, S., Zhang, P., Jeinsch, T., Ding, E., Engel, P., and Gui, W. (2011). A survey of the application of basic data-driven and model-based methods in process monitoring and fault diagnosis. *IFAC Proceedings Volumes*, 44(1):12380–12388. 18th IFAC World Congress.

Ding, S., Zhang, P., Naik, A., Ding, E., and Huang, B. (2009). Subspace method aided data-driven design of fault detection and isolation systems. *Journal of Process Control*, 19(9):1496–1510.

Ding, S. X. (2008). *Model-based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*. Springer Science & Business Media.

Dong, D. and McAvoy, T. J. (1996). Nonlinear principal component analysis—based on principal curves and neural networks. *Computers & Chemical Engineering*, 20(1):65–78.

Downs, J. J. and Vogel, E. F. (1993). A plant-wide industrial process control problem. *Computers & Chemical Engineering*, 17(3):245–255.

Escobar, M. D. (1988). *Estimating the Means of Several Normal Populations by Nonparametric Estimation of the Distribution of the Means*. PhD thesis, Department of Statistics, Yale University.

Escobar, M. D. (1994). Estimating normal means with a Dirichlet Process prior. *Journal of the American Statistical Association*, 89(425):268–277.

Evsukoff, A. and Gentil, S. (2005). Recurrent neuro-fuzzy system for fault detection and isolation in nuclear reactors. *Advanced Engineering Informatics*, 19(1):55–66.

Fazai, R., Taouali, O., Harkat, M. F., and Bouguila, N. (2016). A new fault detection method for nonlinear process monitoring. *The International Journal of Advanced Manufacturing Technology*, 87(9):3425–3436.

Feital, T., Kruger, U., Dutra, J., Pinto, J. C., and Lima, E. L. (2013). Modeling and performance monitoring of multivariate multimodal processes. *AIChE Journal*, 59(5):1557–1569.

Finch, F. E. and Kramer, M. A. (1988). Narrowing diagnostic focus using functional decomposition. *AIChE Journal*, 34(1):25–36.

Forbes, C., Evans, M., Hastings, N., and Peacock, B. (2011). *Statistical Distributions*. John Wiley & Sons.

Gajjar, S., Kulahci, M., and Palazoglu, A. (2018). Real-time fault detection and diagnosis using sparse principal component analysis. *Journal of Process Control*, 67:112–128.

Gamer, T., Hoernicke, M., Kloepper, B., Bauer, R., and Isaksson, A. J. (2020). The autonomous industrial plant – future of process engineering, operations and maintenance. *Journal of Process Control*, 88:101–110.

Garg, S., Steeg, G. V., and Galstyan, A. (2018). Stochastic learning of nonstationary kernels for natural language modeling. arXiv:1801.03911v2 [cs.CL].

Ge, Z. (2017). Review on data-driven modeling and monitoring for plant-wide industrial processes. *Chemometrics and Intelligent Laboratory Systems*, 171:16–25.

Ge, Z. (2018). Process data analytics via probabilistic latent variable models: A tutorial review. *Industrial & Engineering Chemistry Research*, 57(38):12646–12661.

Ge, Z. and Song, Z. (2010). Maximum-likelihood mixture factor analysis model and its application for process monitoring. *Chemometrics and Intelligent Laboratory Systems*, 102(1):53–61.

Ge, Z., Song, Z., and Gao, F. (2013). Review of recent research on data-based process monitoring. *Industrial & Engineering Chemistry Research*, 52(10):3543–3562.

Ge, Z., Yang, C., and Song, Z. (2009). Improved kernel PCA-based monitoring approach for nonlinear processes. *Chemical Engineering Science*, 64(9):2245–2255.

Gertler, J. (1991). Analytical redundancy methods in fault detection and isolation - survey and synthesis. *IFAC Proceedings Volumes*, 24(6):9–21.

Gertler, J. (2017). *Fault Detection and Diagnosis in Engineering Systems*. Routledge.

Gönen, M. and Alpaydin, E. (2008). Localized multiple kernel learning. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, page 352–359, New York, NY, USA. Association for Computing Machinery.

Gonzalez, R., Huang, B., and Lau, E. (2015). Process monitoring using kernel density estimation and Bayesian networking with an industrial case study. *ISA Transactions*, 58:330–347.

Görür, D. and Rasmussen, C. E. (2010). Dirichlet process Gaussian mixture models: Choice of the base distribution. *Journal of Computer Science and Technology*, 25(4):653–664.

Grasso, M., Colosimo, B. M., Semeraro, Q., and Pacella, M. (2015). A comparison study of distribution-free multivariate SPC methods for multimode data. *Quality and Reliability Engineering International*, 31(1):75–96.

He, F., Wang, C., and Fan, S.-K. S. (2018). Nonlinear fault detection of batch processes based on functional kernel locality preserving projections. *Chemometrics and Intelligent Laboratory Systems*, 183:79–89.

He, Q. P. and Wang, J. (2018). Statistical process monitoring as a big data analytics tool for smart manufacturing. *Journal of Process Control*, 67:35–43.

Heckerman, D. (1995). A Bayesian approach to learning causal networks. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, UAI'95, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc. 285–295.

Higdon, D. (1998). A process-convolution approach to modelling temperatures in the North Atlantic Ocean. *Environmental and Ecological Statistics*, 5(2):173–190.

Hoffmann, H. (2007). Kernel PCA for novelty detection. *Pattern Recognition*, 40(3):863–874.

Hotelling, H. (1947). Multivariate quality control-illustrated by the air testing of sample bombsights. In *Techniques of Statistical Analysis*. McGraw-Hill. 111–184.

Hu, Y., Ma, H., and Shi, H. (2013). Enhanced batch process monitoring using just-in-time-learning based kernel partial least squares. *Chemometrics and Intelligent Laboratory Systems*, 123:15–27.

Hwang, D.-H. and Han, C. (1999). Real-time monitoring for a process with multiple operating modes. *Control Engineering Practice*, 7(7):891–902.

Hyvärinen, A. and Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural Networks*, 13(4-5):411–430.

IEEE (1991). *610–1990 - IEEE standard computer dictionary: A compilation of IEEE standard computer glossaries*. Standard, Institute of Electrical and Electronics Engineers, New York.

Isermann, R. (1984). Process fault detection based on modeling and estimation methods—a survey. *Automatica*, 20(4):387–404.

Isermann, R. (1997). Supervision, fault-detection and fault-diagnosis methods—an introduction. *Control Engineering Practice*, 5(5):639–652.

Jackson, J. E. (1985). Multivariate quality control. *Communications in Statistics-Theory and Methods*, 14(11):2657–2688.

Jackson, J. E. and Mudholkar, G. S. (1979). Control procedures for residuals associated with principal component analysis. *Technometrics*, 21(3):341–349.

Jampana, P., L. Shah, S., and Kadali, R. (2010). Computer vision based interface level control in separation cells. *Control Engineering Practice*, 18(4):349–357.

Jansen, F., Shoham, O., and Taitel, Y. (1996). The elimination of severe slugging—experiments and modeling. *International Journal of Multiphase Flow*, 22(6):1055–1072.

Jardine, A. K., Lin, D., and Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical Systems and Signal Processing*, 20(7):1483–1510.

Jia, M., Xu, H., Liu, X., and Wang, N. (2012). The optimization of the kind and parameters of kernel function in KPCA for process monitoring. *Computers & Chemical Engineering*, 46:94–104.

Jiang, B., Guo, Z., Zhu, Q., and Huang, G. (2019). Dynamic minimax probability machine-based approach for fault diagnosis using pairwise discriminate analysis. *IEEE Transactions on Control Systems Technology*, 27(2):806–813.

Jiang, B., Huang, D., Zhu, X., Yang, F., and Braatz, R. D. (2015). Canonical variate analysis-based contributions for fault identification. *Journal of Process Control*, 26:17–25.

Jiang, H., Patwardhan, R., and Shah, S. L. (2009). Root cause diagnosis of plant-wide oscillations using the concept of adjacency matrix. *Journal of Process Control*, 19(8):1347–1354. Special Section on Hybrid Systems: Modeling, Simulation and Optimization.

Jiang, Q. and Yan, X. (2018). Parallel PCA–KPCA for nonlinear process monitoring. *Control Engineering Practice*, 80:17–25.

Jiang, Q. and Yan, X. (2019). Multimode process monitoring using variational Bayesian inference and canonical correlation analysis. *IEEE Transactions on Automation Science and Engineering*, 16(4):1814–1824.

Kachko, A., van der Ham, L. V., Geers, L. F., Huizinga, A., Rieder, A., Abu-Zahra, M. R., Vlugt, T. J., and Goetheer, E. L. (2015). Real-time process monitoring of CO_2 capture by aqueous AMP-PZ using chemometrics: Pilot plant demonstration. *Industrial & Engineering Chemistry Research*, 54(21):5769–5776.

Keerthi, S. S. and Lin, C.-J. (2003). Asymptotic behaviors of support vector machines with Gaussian kernel. *Neural Computation*, 15(7):1667–1689.

Kerr, T. (1977). Real-time failure detection: A nonlinear optimization problem that yields a two-ellipsoid overlap test. *Journal of Optimization Theory and Applications*, 22(4):509–536.

Kiasi, F., Prakash, J., Patwardhan, S., and Shah, S. (2013). A unified framework for fault detection and isolation of sensor and actuator biases in linear time invariant systems using marginalized likelihood ratio test with uniform priors. *Journal of Process Control*, 23(9):1350–1361.

Kim, D. and Lee, I.-B. (2003). Process monitoring based on probabilistic PCA. *Chemometrics and Intelligent Laboratory Systems*, 67(2):109–123.

Kourti, T., Nomikos, P., and MacGregor, J. F. (1995). Analysis, monitoring and fault diagnosis of batch processes using multiblock and multiway PLS. *Journal of Process Control*, 5(4):277–284.

Kramer, M. A. and Palowitch, B. (1987). A rule-based approach to fault diagnosis using the signed directed graph. *AIChE journal*, 33(7):1067–1078.

Ku, W., Storer, R. H., and Georgakis, C. (1995). Disturbance detection and isolation by dynamic principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 30(1):179–196.

Lahdhiri, H., Ben Abdellafou, K., Taouali, O., Mansouri, M., and Korbaa, O. (2019). New online kernel method with the tabu search algorithm for process monitoring. *Transactions of the Institute of Measurement and Control*, 41:2687–2698.

Lee, J.-M., Qin, S. J., and Lee, I.-B. (2007). Fault detection of non-linear processes using kernel independent component analysis. *The Canadian Journal of Chemical Engineering*, 85(4):526–536.

Lee, J.-M., Yoo, C., Choi, S. W., Vanrolleghem, P. A., and Lee, I.-B. (2004a). Nonlinear process monitoring using kernel principal component analysis. *Chemical Engineering Science*, 59(1):223–234.

Lee, J.-M., Yoo, C., and Lee, I.-B. (2004b). Statistical process monitoring with independent component analysis. *Journal of Process Control*, 14(5):467–485.

Lennox, B., Montague, G. A., Frith, A. M., Gent, C., and Bevan, V. (2001). Industrial application of neural networks — an investigation. *Journal of Process Control*, 11(5):497–507.

Li, H., Wang, H., and Fan, W. (2017). Multimode process fault detection based on local density ratio-weighted support vector data description. *Industrial & Engineering Chemistry Research*, 56(9):2475–2491.

Li, N. and Yang, Y. (2015). Ensemble kernel principal component analysis for improved nonlinear process monitoring. *Industrial & Engineering Chemistry Research*, 54(1):318–329.

Lloyd, S. P. (1982). Least squared quantization in PCM. *IEEE Transactions on Information Theory*, 28:129–137.

Lowry, C. A. and Montgomery, D. C. (1995). A review of multivariate control charts. *IIE Transactions*, 27(6):800–810.

Lu, Q., Jiang, B., Gopaluni, R., Loewen, P., and Braatz, R. (2018). Locality preserving discriminative canonical variate analysis for fault diagnosis. *Computers & Chemical Engineering*, 117:309–319.

Lucke, M., Chioua, M., Grimholt, C., Hollender, M., and Thornhill, N. F. (2019a). Advances in alarm data analysis with a practical application to online alarm flood classification. *Journal of Process Control*, 79:56–71.

Lucke, M., Mei, X., Stief, A., Chioua, M., and Thornhill, N. F. (2019b). Variable selection for fault detection and identification based on mutual information of alarm series. *IFAC-PapersOnLine*, 52(1):673–678. 12th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems DYCOPS 2019.

Lucke, M., Stief, A., Chioua, M., Ottewill, J. R., and Thornhill, N. F. (2020). Fault detection and identification combining process measurements and statistical alarms. *Control Engineering Practice*, 94:104195.

Luo, L., Bao, S., Mao, J., and Tang, D. (2015). Nonlinear process monitoring using data-dependent kernel global-local preserving projections. *Industrial & Engineering Chemistry Research*, 54(44):11126–11138.

Ma, H., Hu, Y., and Shi, H. (2013). Fault detection and identification based on the neighborhood standardized local outlier factor method. *Industrial & Engineering Chemistry Research*, 52(6):2389–2402.

MacGregor, J. and Kourti, T. (1995). Statistical process control of multivariate processes. *Control Engineering Practice*, 3(3):403–414.

Mahadevan, S. and Shah, S. L. (2009). Fault detection and diagnosis in process data using one-class support vector machines. *Journal of Process Control*, 19(10):1627–1639.

Metropolis, N. and Ulam, S. (1949). The Monte Carlo method. *Journal of the American Statistical Association*, 44(247):335–341.

Miller, P., Swanson, R. E., and Heckler, C. E. (1998). Contribution plots: A missing link in multivariate quality control. *Applied Mathematics and Computer Science*, 8(4):775–792.

Murphy, K. (2007). *Conjugate Bayesian analysis of the Gaussian distribution*. Technical report, The University of British Columbia, Canada.

Musulin, E., Yélamos, I., and Puigjaner, L. (2006). Integration of principal component analysis and fuzzy logic systems for comprehensive process fault detection and diagnosis. *Industrial & Engineering Chemistry Research*, 45(5):1739–1750.

Navi, M., Meskin, N., and Davoodi, M. (2018). Sensor fault detection and isolation of an industrial gas turbine using partial adaptive KPCA. *Journal of Process Control*, 64:37–48.

Nnabuife, S. G., Pilario, K. E. S., Lao, L., Cao, Y., and Shafiee, M. (2019). Identification of gas-liquid flow regimes using a non-intrusive Doppler ultrasonic sensor and virtual flow regime maps. *Flow Measurement and Instrumentation*, 68:101568.

Nomikos, P. and MacGregor, J. F. (1994). Monitoring batch processes using multiway principal component analysis. *AIChE Journal*, 40(8):1361–1375.

Nor, N. M., Hassan, C. R. C., and Hussain, M. A. (2019). A review of data-driven fault detection and diagnosis methods: Applications in chemical process systems. *Reviews in Chemical Engineering*, 0. `http://doi.org/10.1515/revce-2017-0069`.

Oakland, J. S. (2007). *Statistical Process Control*. Routledge.

Odgaard, P. F. and Stoustrup, J. (2012). Results of a wind turbine FDI competition. *IFAC Proceedings Volumes*, 45(20):102–107.

Odgaard, P. F., Stoustrup, J., and Kinnaert, M. (2013). Fault-tolerant control of wind turbines: A benchmark model. *IEEE Transactions on Control Systems Technology*, 21(4):1168–1182.

Odiowei, P. and Cao, Y. (2009). Nonlinear dynamic process monitoring using canonical variate analysis and kernel density estimations. *Computer Aided Chemical Engineering*, 27:1557–1562.

Onel, M., Kieslich, C. A., and Pistikopoulos, E. N. (2019). A nonlinear support vector machine-based feature selection approach for fault detection and diagnosis: Application to the Tennessee Eastman process. *AIChE Journal*, 65(3):992–1005.

Paciorek, C. J. (2003). *Nonstationary Gaussian Processes for Regression and Spatial Modelling*. PhD thesis, Carnegie Mellon University Pittsburgh.

Patton, R. and Chen, J. (1991). A review of parity space approaches to fault diagnosis. *IFAC Proceedings Volumes*, 24(6):65–81. IFAC/IMACS Symposium on Fault Detection, Supervision and Safety for Technical Processes (SAFEPROCESS'91), Baden-Baden, Germany, 10-13 September 1991.

Pilario, K. E., Shafiee, M., Cao, Y., Lao, L., and Yang, S.-H. (2020). A review of kernel methods for feature extraction in nonlinear process monitoring. *Processes*, 8(1):24.

Pilario, K. E. S. and Cao, Y. (2018). Canonical variate dissimilarity analysis for process incipient fault detection. *IEEE Transactions on Industrial Informatics*, 14(12):5308–5315.

Pilario, K. E. S., Cao, Y., and Shafiee, M. (2019). Mixed kernel canonical variate dissimilarity analysis for incipient fault monitoring in nonlinear dynamic processes. *Computers & Chemical Engineering*, 123:143–154.

Pirdashti, M., Curteanu, S., Kamangar, M. H., Hassim, M. H., and Khatami, M. A. (2013). Artificial neural networks: Applications in chemical engineering. *Reviews in Chemical Engineering*, 29(4):205–239.

Qin, J. S. (2003). Statistical process monitoring: Basics and beyond. *Journal of Chemometrics*, 17(8-9):480–502.

Qin, S. J. (2012). Survey on data-driven industrial process monitoring and diagnosis. *Annual Reviews in Control*, 36(2):220–234.

Qin, S. J. and Chiang, L. H. (2019). Advances and opportunities in machine learning for process data analytics. *Computers & Chemical Engineering*, 126:465–473.

Quiñones-Grueiro, M., Prieto-Moreno, A., Verde, C., and Llanes-Santiago, O. (2019). Data-driven monitoring of multimode continuous processes: A review. *Chemometrics and Intelligent Laboratory Systems*, 189:56–71.

Rakitzis, A. C., Chakraborti, S., Shongwe, S. C., Graham, M. A., and Khoo, M. B. C. (2019). An overview of synthetic-type control charts: Techniques and methodology. *Quality and Reliability Engineering International*, 35(7):2081–2096.

Raveendran, R. and Huang, B. (2017). Two layered mixture Bayesian probabilistic PCA for dynamic process monitoring. *Journal of Process Control*, 57:148–163.

Reynolds, D. (2015). Gaussian mixture models. *Encyclopedia of Biometrics*, pages 827–832.

Ricker, N. (1995). Optimal steady-state operation of the Tennessee Eastman challenge process. *Computers & Chemical Engineering*, 19(9):949–959.

Robinson, D. (2018). What's the difference between data science, machine learning, and artificial intelligence? `http://varianceexplained.org/r/ds-ml-ai/`. Accessed 01 January 2020.

Ruiz-Cárcel, C., Cao, Y., Mba, D., Lao, L., and Samuel, R. (2015). Statistical process monitoring of a multiphase flow facility. *Control Engineering Practice*, 42:74–88.

Russell, E. L., Chiang, L. H., and Braatz, R. D. (2000). Fault detection in industrial processes using canonical variate analysis and dynamic principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 51(1):81–93.

Schölkopf, B., Smola, A., and Müller, K.-R. (1997). Kernel principal component analysis. In *International Conference on Artificial Neural Networks*. Springer. 583–588.

Schölkopf, B., Smola, A., and Müller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319.

Shang, C., Ji, H., Huang, X., Yang, F., and Huang, D. (2019). Generalized grouped contributions for hierarchical fault diagnosis with group LASSO. *Control Engineering Practice*, 93:104193.

Shang, C., Yang, F., Gao, X., Huang, X., Suykens, J. A., and Huang, D. (2015). Concurrent monitoring of operating condition deviations and process dynamics anomalies with slow feature analysis. *AIChE Journal*, 61(11):3666–3682.

Shewhart, W. A. (1931). *Economic Control of Quality of Manufactured Product*. Van Nostrand.

Singh, M. K., Venkatesh, K., and Dutta, A. (2016). Kernel based approach for accurate surface estimation. *Computers & Electrical Engineering*, 56:763–772.

Sánchez-Fernández, A., Baldán, F., Sainz-Palmero, G., Benítez, J., and Fuente, M. (2018). Fault detection based on time series modeling and multivariate statistical process control. *Chemometrics and Intelligent Laboratory Systems*, 182:57–69.

Song, B. and Shi, H. (2018). Temporal-spatial global locality projections for multimode process monitoring. *IEEE Access*, 6:9740–9749.

Song, B., Tan, S., and Shi, H. (2016). Key principal components with recursive local outlier factor for multimode chemical process monitoring. *Journal of Process Control*, 47:136–149.

Sorsa, T., Koivo, H. N., and Koivisto, H. (1991). Neural networks in process fault diagnosis. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(4):815–825.

Stief, A., Ottewill, J. R., and Baranowski, J. (2019a). Investigation of the diagnostic properties of sensors and features in a multiphase flow facility case study. *IFAC-PapersOnLine*, 52(1):772–777.

Stief, A., Ottewill, J. R., Tan, R., and Cao, Y. (2018). Process and alarm data integration under a two-stage Bayesian framework for fault diagnostics. *IFAC-PapersOnLine*, 51(24):1220–1226.

Stief, A., Tan, R., Cao, Y., and Ottewill, J. R. (2019b). PRONTO heterogeneous benchmark dataset. http://doi.org/10.5281/zenodo.1341583.

Stief, A., Tan, R., Cao, Y., Ottewill, J. R., Thornhill, N. F., and Baranowski, J. (2019c). A heterogeneous benchmark dataset for data analytics: Multiphase flow facility case study. *Journal of Process Control*, 79:41–55.

Sukchotrat, T., Kim, S. B., and Tsung, F. (2009). One-class classification-based control charts for multivariate process monitoring. *IIE Transactions*, 42(2):107–120.

Suresh, R., Sivaram, A., and Venkatasubramanian, V. (2019). A hierarchical approach for causal modeling of process systems. *Computers & Chemical Engineering*, 123:170–183.

Tan, R. and Cao, Y. (2019). Deviation contribution plots of multivariate statistics. *IEEE Transactions on Industrial Informatics*, 15(2):833–841.

Tan, R., Cong, T., Ottewill, J. R., Baranowski, J., and Thornhill, N. F. (2020a). An on-line framework for monitoring nonlinear processes with multiple operating modes. *Journal of Process Control*, 89:119–130.

Tan, R., Cong, T., Thornhill, N. F., Ottewill, J. R., and Baranowski, J. (2019). Statistical monitoring of processes with multiple operating modes. *IFAC-PapersOnLine*, 52(1):635–642.

Tan, R., Ottewill, J. R., and Thornhill, N. F. (2020b). Monitoring statistics and tuning of kernel principal component analysis with radial basis function kernels. *IEEE Access*. under review.

Tan, R., Ottewill, J. R., and Thornhill, N. F. (2020). Nonstationary discrete convolution kernel for multimodal process monitoring. *IEEE Transactions on Neural Networks and Learning Systems*, 31(9):3670–3681.

Teh, Y. W. (2011). Dirichlet process. In *Encyclopedia of Machine Learning*, pages 280–287. Springer.

Thorn, R., Johansen, G. A., and Hjertaker, B. T. (2012). Three-phase flow measurement in the petroleum industry. *Measurement Science and Technology*, 24(1):012003.

Thornhill, N., Shah, S., and Huang, B. (2001). Detection of distributed oscillations and root-cause diagnosis. *IFAC Proceedings Volumes*, 34(27):149–154. 4th IFAC Workshop on On-Line Fault Detection and Supervision in the Chemical Process Industries 2001, Jejudo Island, Korea, 7-8 June 2001.

Thornhill, N. F., Choudhury, M. S., and Shah, S. L. (2004). The impact of compression on data-driven process analyses. *Journal of Process Control*, 14(4):389–398.

Thornhill, N. F., Patwardhan, S. C., and Shah, S. L. (2008). A continuous stirred tank heater simulation model with applications. *Journal of Process Control*, 18(3):347–360.

Tong, C., Lan, T., and Shi, X. (2017). Fault detection and diagnosis of dynamic processes using weighted dynamic decentralized PCA approach. *Chemometrics and Intelligent Laboratory Systems*, 161:34–42.

Treasure, R. J., Kruger, U., and Cooper, J. E. (2004). Dynamic multivariate statistical process control using subspace identification. *Journal of Process Control*, 14(3):279–292.

Twining, C. J. and Taylor, C. J. (2003). The use of kernel principal component analysis to model data distributions. *Pattern Recognition*, 36(1):217–227.

Van Impe, J. and Gins, G. (2015). An extensive reference dataset for fault detection and identification in batch processes. *Chemometrics and Intelligent Laboratory Systems*, 148:20–31.

van Lith, P. F., Betlem, B. H., and Roffel, B. (2003). Combining prior knowledge with data driven modeling of a batch distillation column including start-up. *Computers & Chemical Engineering*, 27(7):1021–1030.

Venkatasubramanian, V. (2019). The promise of artificial intelligence in chemical engineering: Is it here, finally? *AIChE Journal*, 65(2):466–478.

Venkatasubramanian, V., Rengaswamy, R., and Kavuri, S. N. (2003a). A review of process fault detection and diagnosis: Part II: Qualitative models and search strategies. *Computers & Chemical Engineering*, 27(3):313–326.

Venkatasubramanian, V., Rengaswamy, R., Kavuri, S. N., and Yin, K. (2003b). A review of process fault detection and diagnosis: Part III: Process history based methods. *Computers & Chemical Engineering*, 27(3):327–346.

Venkatasubramanian, V., Rengaswamy, R., Yin, K., and Kavuri, S. N. (2003c). A review of process fault detection and diagnosis: Part I: Quantitative model-based methods. *Computers & Chemical Engineering*, 27(3):293–311.

Von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416.

Wang, J., Ge, W., Zhou, J., Wu, H., and Jin, Q. (2017). Fault isolation based on residual evaluation and contribution analysis. *Journal of the Franklin Institute*, 354(6):2591–2612.

Wang, J., Yang, F., Chen, T., and Shah, S. L. (2016). An overview of industrial alarm systems: Main causes for alarm overloading, research status, and open problems. *IEEE Transactions on Automation Science and Engineering*, 13(2):1045–1061.

Wang, K., Rippon, L., Chen, J., Song, Z., and Gopaluni, R. B. (2019). Data-driven dynamic modeling and online monitoring for multiphase and multimode batch processes with uneven batch durations. *Industrial & Engineering Chemistry Research*, 58(30):13628–13641.

Willis, A. (2010). Condition monitoring of centrifuge vibrations using kernel PLS. *Computers & Chemical Engineering*, 34(3):349–353.

Wilson, A. G. (2014). *Covariance Kernels for Fast Automatic Pattern Discovery and Extrapolation with Gaussian Processes*. PhD thesis, University of Cambridge.

Wise, B. M. and Gallagher, N. B. (1996). The process chemometrics approach to process monitoring and fault detection. *Journal of Process Control*, 6(6):329–348.

Wise, B. M., Gallagher, N. B., Butler, S. W., White, D. D., and Barna, G. G. (1999). A comparison of principal component analysis, multiway principal component analysis, trilinear decomposition and parallel factor analysis for fault detection in a semiconductor etch process. *Journal of Chemometrics*, 13(3-4):379–396.

Xiao, Y., Wang, H., Xu, W., and Zhou, J. (2016). Robust one-class SVM for fault detection. *Chemometrics and Intelligent Laboratory Systems*, 151:15–25.

Xie, W., Xie, M., and Goh, T. (1995). A Shewhart-like charting technique for high yield processes. *Quality and Reliability Engineering International*, 11(3):189–196.

Xie, X. and Shi, H. (2012). Dynamic multimode process modeling and monitoring using adaptive Gaussian mixture models. *Industrial & Engineering Chemistry Research*, 51(15):5497–5505.

Xing, L., Yeung, H., Shen, J., and Cao, Y. (2013). Experimental study on severe slugging mitigation by applying wavy pipes. *Chemical Engineering Research and Design*, 91(1):18–28.

Xu, J., Wang, J., Izadi, I., and Chen, T. (2012). Performance assessment and design for univariate alarm systems based on FAR, MAR, and AAD. *IEEE Transactions on Automation Science and Engineering*, 9(2):296–307.

Xu, R. and Wunsch, D. (2005). Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, 16(3):645–678.

Xu, S., Lu, B., Baldea, M., Edgar, T. F., Wojsznis, W., Blevins, T., and Nixon, M. (2015a). Data cleaning in the process industries. *Reviews in Chemical Engineering*, 31(5):453–490.

Xu, Y., Zhou, Z.-Q., and Zhu, Q.-X. (2015b). A new feedback DE-ELM with time delay-based EFSM approach for fault prediction of non-linear processes. *The Canadian Journal of Chemical Engineering*, 93(9):1603–1612.

Yang, F., Duan, P., Shah, S. L., and Chen, T. (2014). *Capturing Connectivity and Causality in Complex Industrial Processes*. Springer.

Yang, F., Shah, S. L., and Xiao, D. (2010). SDG (Signed Directed Graph) based process description and fault propagation analysis for a tailings pumping process. *IFAC Proceedings Volumes*, 43(9):50–55. 13th IFAC Symposium on Automation in Mining, Mineral and Metal Processing.

Yim, S. Y., Ananthakumar, H. G., Benabbas, L., Horch, A., Drath, R., and Thornhill, N. F. (2006). Using process topology in plant-wide control loop performance assessment. *Computers & Chemical Engineering*, 31(2):86–99.

Yin, S., Ding, S. X., Haghani, A., Hao, H., and Zhang, P. (2012). A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process. *Journal of Process Control*, 22(9):1567–1581.

Yin, S., Ding, S. X., Xie, X., and Luo, H. (2014). A review on basic data-driven approaches for industrial process monitoring. *IEEE Transactions on Industrial Electronics*, 61(11):6418–6428.

Yochum, B. T. (1973). Offshore riser slug flow avoidance: Mathematical models for design and optimization. In *SPE European Meeting*. Society of Petroleum Engineers. `http://doi.org/10.2118/4312-MS`.

Yu, J. and Qin, S. J. (2008). Multimode process monitoring with Bayesian inference-based finite Gaussian mixture models. *AIChE Journal*, 54(7):1811–1829.

Yu, J., Yoo, J., Jang, J., Park, J. H., and Kim, S. (2018). A novel hybrid of auto-associative kernel regression and dynamic independent component analysis for fault detection in nonlinear multimode processes. *Journal of Process Control*, 68. 129–144.

Zhang, C., Gao, X., Xu, T., Li, Y., and Pang, Y. (2018). Fault detection and diagnosis strategy based on a weighted and combined index in the residual subspace associated with pca. *Journal of Chemometrics*, 32(11):e2981.

Zhang, H., Qi, Y., Wang, L., Gao, X., and Wang, X. (2017). Fault detection and diagnosis of chemical process using enhanced KECA. *Chemometrics and Intelligent Laboratory Systems*, 161:61–69.

Zhang, K., Peng, K., and Dong, J. (2018). A common and individual feature extraction-based multimode process monitoring method with application to the finishing mill process. *IEEE Transactions on Industrial Informatics*, 14(11):4841–4850.

Zhang, X., Yan, W., Zhao, X., and Shao, H. (2007). Nonlinear biological batch process monitoring and fault identification based on kernel fisher discriminant analysis. *Process Biochemistry*, 42(8):1200–1210.

Zhang, Y., Zhang, L., and Lu, R. (2013). Fault identification of nonlinear processes. *Industrial & Engineering Chemistry Research*, 52(34):12072–12081.

Zhao, C., Wang, W., Qin, Y., and Gao, F. (2015). Comprehensive subspace decomposition with analysis of between-mode relative changes for multimode process monitoring. *Industrial & Engineering Chemistry Research*, 54(12):3154–3166.

Zhao, S. J., Zhang, J., and Xu, Y. M. (2006). Performance monitoring of processes with multiple operating modes through multiple PLS models. *Journal of Process Control*, 16(7):763–772.

Zhou, L., Zheng, J., Ge, Z., Song, Z., and Shan, S. (2018). Multimode process monitoring based on switching autoregressive dynamic latent variable model. *IEEE Transactions on Industrial Electronics*, 65(10):8184–8194.

Zhou, M., Liu, Z., Cai, Y., and Pan, H. (2019). Incipient fault detection based on exergy efficiency and support vector data description. *Journal of Chemical Engineering of Japan*, 52(6):562–569.

Zhu, Z., Song, Z., and Palazoglu, A. (2012). Process pattern construction and multi-mode monitoring. *Journal of Process Control*, 22(1):247–262.

Appendices

A Appendix for Chapter 5: derivation of the tuning point in T^2

This appendix gives a detailed derivation of Eqn (5.41) in Section 5.5.1. The maximum δ value δ_{max} is to ensure that the tuning point of the monitoring statistic T^2 is contained within the healthy data and T^2 decreases monotonically when a test sample moves away from the healthy data.

Fig. 5.5(b) has shown that a common turning point at a specific value of d^2/δ^2 exists for T^2 when using large δ values, resulting in a non-monotonic T^2. We explore the non-monotonicity behaviour of T^2 and the common turning point in this Appendix. To find the turning point, the derivative of T^2_{test} with respect to x_{test} is investigated. Since T^2_{test} is a function of $\tilde{K}_{i,\text{test}}$, the problem is to find the local optimum for T^2_{test} by finding a local optimum that applies to all $\tilde{K}_{i,\text{test}}$.

It is clear that $K_{i,\text{test}}$ is monotonically decreasing with respect to the Euclidean distance between $\boldsymbol{x}_{\text{test}}$ and $\boldsymbol{x}_i$, i.e. $d_{i,\text{test}} = \sqrt{\|\boldsymbol{x}_{\text{test}} - \boldsymbol{x}_i\|_2}$, because of the RBF function:

$$K_{i,\text{test}} = \exp\left(-\frac{(\boldsymbol{x}_i - \boldsymbol{x}_{\text{test}})^\top(\boldsymbol{x}_i - \boldsymbol{x}_{\text{test}})}{\delta^2}\right). \tag{A.1}$$

However, $\tilde{K}_{i,\text{test}}$ may not be monotonic after centering (Eqn (5.8)). The derivative of $\tilde{K}_{i,\text{test}}$ with respect to $\boldsymbol{x}_{\text{test}}$ is investigated to check its monotonicity:

$$\frac{\partial \tilde{K}_{i,\text{test}}}{\partial \boldsymbol{x}_{\text{test}}} = \frac{\partial\left[K_{i,\text{test}} - \bar{K}_{i,\text{test}} + C_i\right]}{\partial \boldsymbol{x}_{\text{test}}} \tag{A.2}$$

where $C_i = 1/n^2 \sum\sum(K_{.,.}) - 1/n \sum_i (K_{i,.})$ is constant with respect to $\boldsymbol{x}_{\text{test}}$. For simplicity, x_{test} is assumed to be a scalar.

By assuming $x_{\text{test}} > x_i$, Eqn (A.2) can be simplified as:

$$\begin{aligned}\frac{\partial \tilde{K}_{i,\text{test}}}{\partial x_{\text{test}}} &= \frac{\partial \exp\left(-\frac{(x_{\text{test}}-x_i)^2}{\delta^2}\right)}{\partial x_{\text{test}}} - \frac{\frac{1}{n}\sum_{j=1}^{n} \exp\left(-\frac{(x_{\text{test}}-x_j)^2}{\delta^2}\right)}{\partial x_{\text{test}}} \\ &= -\frac{2}{\delta^2}(x_{\text{test}} - x_i)\exp\left(-\frac{(x_{\text{test}}-x_i)^2}{\delta^2}\right) + \\ &\frac{2}{n\delta^2}\sum_{j=1}^{n}(x_{\text{test}} - x_j)\exp\left(-\frac{(x_{\text{test}}-x_j)^2}{\delta^2}\right).\end{aligned} \tag{A.3}$$

When the stationary point occurs for $\tilde{K}_{i,\text{test}}$, Eqn (A.3) will be equal to zero. Denoting $a_i = 2/\delta^2(x_{\text{test}} - x_i)\exp\left(-(x_{\text{test}} - x_i)^2/\delta^2\right)$, Eqn (A.4) holds for the stationary point:

$$a_i - \frac{1}{n}\sum_{j=1}^{n} a_j = 0. \tag{A.4}$$

Since

$$T^2_{\text{test}} = \sum_{l=1}^{L} \lambda_l^{-1} \left(\sum_{i=1}^{n} \tilde{\alpha}_i^{(l)} \tilde{K}_{i,\text{test}}\right)^2 \tag{A.5}$$

and

$$\frac{\partial T^2_{\text{test}}}{\partial x_{\text{test}}} = 2\sum_{l=1}^{L} \lambda_l^{-1} \left(\sum_{i=1}^{n} \tilde{\alpha}_i^{(l)} \tilde{K}_{i,\text{test}}\right)\left(\sum_{i=1}^{n} \tilde{\alpha}_i^{(l)} \frac{\partial \tilde{K}_{i,\text{test}}}{\partial x_{\text{test}}}\right), \tag{A.6}$$

a sufficient but not necessary condition for T^2 having a maximum is that all $\tilde{K}_{i,\text{test}}$ have the same local maxima or minima. If there exists a common stationary point for all $\tilde{K}_{i,\text{test}}$, where $i = \{1, \ldots, n\}$, at x_{test}, the following matrix equation

$$\begin{bmatrix} 1/n-1 & 1/n & \ldots & 1/n \\ 1/n & 1/n-1 & \ldots & 1/n \\ \ldots & & & \\ 1/n & 1/n & \ldots & 1/n-1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \ldots \\ a_n \end{bmatrix} = \mathbf{0} \tag{A.7}$$

is valid. The non-zero solution to Eqn (A.7) is any non-zero vector $\{a_1, a_2, \ldots, a_n\}$ that satisfies $a_1 = a_2 = \cdots = a_n$. However, since $x_{\text{test}} - x_i$ are different for different i values, the solution to Eqn (A.7) is infeasible. Instead, one may consider the following minimization

problem where all $\tilde{K}_{i,\text{test}}$ have their stationary points in a very small neighbourhood:

$$\begin{aligned} &\min_{x_{\text{test}}} \sum_{i=1}^{n} \left(a_i - \frac{1}{n} \sum_{j=1}^{n} a_j \right)^2 \\ \text{s.t.} \quad &a_i = \frac{2}{\delta^2}(x_{\text{test}} - x_i) \exp\left(-\frac{(x_{\text{test}} - x_i)^2}{\delta^2} \right). \end{aligned} \tag{A.8}$$

To simplify this problem:

$$\begin{aligned} \arg\min_{x_{\text{test}}} \sum_{i=1}^{n} \left(a_i - \frac{1}{n} \sum_{j=1}^{n} a_j \right)^2 &= \arg\min_{x_{\text{test}}} \left[\frac{n-1}{n} \sum_{i=1}^{n} a_i^2 - \frac{1}{n} \sum_{i=1}^{n} \sum_{j \neq i}^{n} a_i a_j \right] \\ &= \arg\min_{x_{\text{test}}} \sum_{i=1}^{n} \sum_{j \neq i}^{n} (a_i - a_j)^2 \end{aligned} \tag{A.9}$$

where

$$a_i - a_j = \frac{2}{\delta^2}(x_{\text{test}} - x_i) \exp\left(-\frac{(x_{\text{test}} - x_i)^2}{\delta^2} \right) - \frac{2}{\delta^2}(x_{\text{test}} - x_j) \exp\left(-\frac{(x_{\text{test}} - x_j)^2}{\delta^2} \right). \tag{A.10}$$

Now considering a function of x:

$$f(x) = \frac{2}{\delta^2} x \exp\left(-\frac{x^2}{\delta^2} \right), \tag{A.11}$$

Eqn (A.10) can be written as:

$$a_i - a_j = f(x_{\text{test}} - x_i) - f(x_{\text{test}} - x_j). \tag{A.12}$$

It is reasonable to assume that $|x_i|, |x_j| << x_{\text{test}}$. Eqn (A.12) becomes:

$$a_i - a_j = [f(x - x_i) - f(x - x_j)]\Big|_{x = x_{\text{test}}} \approx f'(x)\Big|_{x = x_{\text{test}}} (x_j - x_i). \tag{A.13}$$

Then the optimization problem in (A.9) becomes:

$$\arg\min_{x_{\text{test}}} \sum_{i=1}^{n} \sum_{j \neq i}^{n} \left(f'(x)\Big|_{x = x_{\text{test}}} (x_j - x_i) \right)^2 = \arg\min_{x_{\text{test}}} f'(x)\Big|_{x = x_{\text{test}}} \sum_{i=1}^{n} \sum_{j \neq i}^{n} (x_i - x_j)^2. \tag{A.14}$$

Given that $\sum_{i=1}^{n}\sum_{j\neq i}^{n}(x_i - x_j)^2$ is constant when the training set is fixed, the solution to this optimization problem will be $\hat{x}_{\text{test}}$ such that:

$$f'(x)\Big|_{x=\hat{x}_{\text{test}}} = \frac{2}{\delta^2}\left[(1 - \frac{2\hat{x}_{\text{test}}^2}{\delta^2})\exp\left(-\frac{\hat{x}_{\text{test}}^2}{\delta^2}\right)\right] = 0. \tag{A.15}$$

The solutions to this condition are:

$$\hat{x}_{\text{test},1} = \frac{\delta}{\sqrt{2}} \quad \text{and} \quad \hat{x}_{\text{test},2} = \infty. \tag{A.16}$$

In this univariate case, the distance between x_{test} and the origin is $d_{\text{test}} = |x_{\text{test}}|$. Therefore, the tuning point of T^2_{test} exists such that $\hat{d}^2_{\text{test}} = \delta^2/2$, which explains the behaviour in Fig. 5.5(b).

B Appendix for Chapter 6: derivation for setting the constant σ_w

This appendix explains why setting $\sigma_w^2 = \sigma_0^2/p$ in Eqn (6.9) in Section 6.3.1 guarantees that the value of the covariance function does not go to infinity.

The following derivation shows that it is necessary to set $\sigma_w^2 = \sigma_0^2/p$ to guarantee that Eqn (6.8) does not approach infinity. Considering $\phi_{\boldsymbol{c}^{(i)}}(\boldsymbol{x}) = \exp\left(-\frac{(\boldsymbol{x}-\mathbf{c}^{(i)})^\top(\boldsymbol{x}-\mathbf{c}^{(i)})}{l^2}\right)$ in Eqn (6.7), $\text{cov}(y, y^*)$ will approach infinity as the number of base functions p increases given σ^2 is an arbitrary constant:

$$\begin{aligned}\text{cov}(y, y^*) = \sigma^2 \lim_{p\to\infty} \sum_{i=1}^{p} \exp\left(-\frac{(\boldsymbol{x}-\boldsymbol{c}^{(i)})^\top(\boldsymbol{x}-\boldsymbol{c}^{(i)})}{l^2}\right) \exp\left(-\frac{(\boldsymbol{x}^*-\boldsymbol{c}^{(i)})^\top(\boldsymbol{x}^*-\boldsymbol{c}^{(i)})}{l^2}\right) \\ \to \infty.\end{aligned} \tag{B.17}$$

Since

$$0 \leq \exp\left(-\frac{(\boldsymbol{x}^*-\boldsymbol{c}^{(i)})^\top(\boldsymbol{x}^*-\boldsymbol{c}^{(i)})}{l^2}\right) \leq 1, \tag{B.18}$$

the asymptotic behaviour of $\text{cov}(y, y^*)$ may be approximated using triangular inequality:

$$O(p) = \sum_{i=1}^{p} \exp\left(-\frac{(\boldsymbol{x}-\boldsymbol{c}^{(i)})^\top(\boldsymbol{x}-\boldsymbol{c}^{(i)})}{l^2}\right) \exp\left(-\frac{(\boldsymbol{x}^*-\boldsymbol{c}^{(i)})^\top(\boldsymbol{x}^*-\boldsymbol{c}^{(i)})}{l^2}\right) \tag{B.19}$$

where $O(p)$ denotes the group of functions with first order infinity with respect to p. Therefore, the variance of w_i is set to be δ_0^2/p so as to avoid the divergence of $\text{cov}(y, y^*)$.

C Appendix for Chapter 8: a dynamic RBF kernel for non-linear dynamic process monitoring

This appendix presents a new formulation of a dynamic kernel. The new formulation is compared with an existing dynamic adaptation of KPCA. Some preliminary results of fault detection using this new dynamic kernel are presented. The section outlines several directions for future research.

C.1 Background

Ge et al. (2013) reported that the *dynamic* behaviour of a process often results in the autocorrelation between different sampling points of process variables. According to Ge et al. (2013), the autocorrelation may be due to the time constant of process, feedback control loops, or autocorrelated process noise. The method for dynamic process monitoring investigated here focuses on the autocorrelation caused by large time constants of processes.

This section aims to formulate a kernel that is suitable for building monitoring models for dynamic processes. The formulation is based on the way the RBF kernel is formulated as a covariance function. Chapter 6 describes an example of deriving a new kernel by revising the formulation of RBF kernels. This Appendix explores how a dynamic kernel can be derived using this formulation.

Section C.2 gives the dynamic adaptation of the RBF kernel. Section C.3 compares this adaptation with the dynamic KPCA proposed by Choi and Lee (2004). Section C.4 presents preliminary results of applying the new kernel to dynamic process monitoring. Section C.5 outlines the directions for future works towards kernel-based monitoring of dynamic processes.

C.2 Dynamic RBF kernels

Review of the convolution kernel formulation

Chapter 6 discussed the formulation of kernel functions as covariance functions. If y is defined as Eqn (C.20), then the covariance of y can be used as a kernel function of $\boldsymbol{x}$:

$$y = \sum_{i=1}^{p} w_i \phi_i(\boldsymbol{x}) \tag{C.20}$$

where $w_i \sim N\left(0, \sigma_w^2\right)$ are the regression coefficients with independent and identical Gaussian distribution corresponding to basis functions $\phi_i(\boldsymbol{x})$ for $i = 1, 2, \cdots, p$.

RBF kernels are formulated using radial basis function in Eqn (C.20). The number of basis functions $p \to \infty$. The centers $\boldsymbol{c}^{(i)}$ are allocated evenly from $-\infty$ to ∞. Given $\mathrm{d}\boldsymbol{c} = \boldsymbol{c}^{(i+1)} - \boldsymbol{c}^{(i)}$, the kernel function becomes:

$$\begin{aligned}
\operatorname{cov}(y, y^*) &= \sigma_w^2 \sum_{i=1}^{p} \phi_i(\boldsymbol{x}) \phi_i(\boldsymbol{x}^*) \\
&= \sigma_w^2 \sum_{i=1}^{p} \left[\exp\left(-\frac{(\boldsymbol{x} - \boldsymbol{c}^{(i)})^\top (\boldsymbol{x} - \boldsymbol{c}^{(i)})}{l^2} \right) \exp\left(-\frac{(\boldsymbol{x}^* - \boldsymbol{c}^{(i)})^\top (\boldsymbol{x}^* - \boldsymbol{c}^{(i)})}{l^2} \right) \right] \\
&= \sqrt{\pi} l \sigma_0^2 \exp\left(-\frac{(\boldsymbol{x} - \boldsymbol{x}^*)^\top (\boldsymbol{x} - \boldsymbol{x}^*)}{\delta^2} \right) \\
&= k_{\text{RBF}}(\boldsymbol{x}, \boldsymbol{x}^*)
\end{aligned} \tag{C.21}$$

where σ_w^2 is selected as σ_0^2 / p and σ_0 is a finite constant. $\delta^2 = 2l^2$ is the kernel width of the RBF kernel.

Dynamic adaptation of RBF kernels

This section investigates the dynamic adaptation of RBF kernels by revising Eqn (C.20). The adapted kernel uses the measurements in a given time window. y_t at time t is now defined as Eqn (C.22) using $\boldsymbol{x}_t, \boldsymbol{x}_{t-1}, \cdots, \boldsymbol{x}_{t-\tau+1}$, the samples obtained in the time window $[t - \tau + 1, t]$.

$$y_t = \sum_{j=0}^{\tau-1} \sum_{i=1}^{p} w_{i,\tau} \phi_i(\boldsymbol{x}_{t-j}) \tag{C.22}$$

The covariance function of y_{t_1} and y_{t_2} is derived accordingly:

$$\begin{aligned}\text{cov}(y_{t_1}, y_{t_2}) &= k(X_{t_1:t_1-\tau+1}, X_{t_2:t_2-\tau+1}) \\ &= \mathbb{E}\left[\sum_{q=0}^{l-1}\sum_{i=1}^{p} w_{i,q}\phi_i(\boldsymbol{x}_{t_1-q})\sum_{q=0}^{\tau-1}\sum_{i=1}^{p} w_{i,q}\phi_i(\boldsymbol{x}_{t_2-q})\right]\end{aligned} \tag{C.23}$$

where $X_{t_1:t_1-\tau+1} = [\boldsymbol{x}_{t_1}, \boldsymbol{x}_{t_1-1}, \cdots, \boldsymbol{x}_{t_1-\tau+1}]$ denotes the lagged data at time t_1. $w_{i,q}$ are Gaussian distributed and their variances σ_q^2 and covariances are functions of the specific time lag q.

The coefficients of the terms in Eqn (C.23) are considered according to several assumptions of the autocorrelations existing between the basis functions and the lagged samples:

1. $\mathbb{E}[w_{i,q}^2] = \sigma_q^2$ for the i-th basis function ϕ_i given the two samples $\boldsymbol{x}_{t_1-q}$ and x_{t_2-q} with the same time lag q.
2. $\mathbb{E}[w_{i,q}w_{j,q}] = 0$ if $i \neq j$. This is because two different basis functions ϕ_i and ϕ_j should be orthogonal to each other. The coefficients $w_{i,q}$ and $w_{j,q}$ should therefore be independent.
3. $\mathbb{E}[w_{i,q_1}w_{i,q_2}] = \alpha_{q_1,q_2}$. This represents the correlation between the coefficients of the i-th basis function ϕ_i given two different lagged samples $\boldsymbol{x}_{t-q_1}$ and $\boldsymbol{x}_{t-q_2}$.
4. $\mathbb{E}[w_{i,q_1}w_{j,q_2}] = 0$ for two different basis functions ϕ_i and ϕ_j when $i \neq j$.

The covariance function is derived from Eqn (C.23) using the coefficients discussed previously.

$$\text{cov}(y_{t_1}, y_{t_2}) = \sum_{q=0}^{\tau-1}\sigma_q^2\sum_{i=1}^{p}\phi_i(\boldsymbol{x}_{t_1-q})\phi_i(\boldsymbol{x}_{t_2-q}) + \sum_{q_1=0}^{\tau-1}\sum_{q_2\neq q_1}\alpha_{q_1,q_2}\sum_{i=1}^{p}\phi_i(\boldsymbol{x}_{t_1-q_1})\phi_i(\boldsymbol{x}_{t_2-q_2}). \tag{C.24}$$

If the assumptions for developing RBF kernels are applied (ϕ_is are radial basis functions and $p \to \infty$), the covariance function becomes:

$$\begin{aligned}\text{cov}(y_{t_1}, y_{t_2}) &= \sum_{q=0}^{\tau-1}\sigma_q^2 k_{\text{RBF}}(\boldsymbol{x}_{t_1-q}, \boldsymbol{x}_{t_2-q}) + \sum_{q_1=0}^{\tau-1}\sum_{q_2\neq q_1}\alpha_{q_1,q_2}k_{\text{RBF}}(\boldsymbol{x}_{t_1-q_1}, \boldsymbol{x}_{t_2-q_2}) \\ &= k_{\text{DRBF}}(X_{t_1:t_1-\tau+1}, X_{t_2:t_2-\tau+1})\end{aligned} \tag{C.25}$$

This is achieved by substituting the result in Eqn (C.21) to Eqn (C.24). The new Dynamic Radial Basis Function (DRBF) kernel is a function of two groups of lagged data, for example

$X_{t_1:t_1-\tau+1}$ and $X_{t_2:t_2-\tau+1}$. The new kernel is a weighted combination of entries in the $\tau \times \tau$ sub-matrix $K_{\text{RBF}}(t_1-\tau+1:t_1, t_2-\tau+1:t_2)$ of the RBF kernel matrix K_{RBF}. The parameters α_{q_1,q_2} represent how strong the autocorrelation is between lagged samples.

The parameter α_{q_1,q_2} lies in $(0,1)$ because it represents the autocorrelation. α_{q_1,q_2} should also be monotonically decreasing if $|q_1 - q_2| \in \{1, 2, \cdots, \tau-1\}$ increases, which indicates that the correlation will be small if the time lag between two samples is large. The parameters α_{q_1,q_2} can either be fitted to the training data or be tuned manually. For example, the α_{q_1,q_2} can be set as:

$$\alpha_{q_1,q_2} = \frac{1}{|q_1 - q_2| + 1}. \tag{C.26}$$

If the values of σ_q^2 are considered to be a constant $s\sigma^2$ for all time lags q, then the DRBF kernel becomes:

$$\begin{aligned} k_{\text{DRBF}}(X_{t_1:t_1-\tau+1}, X_{t_2:t_2-\tau+1}) =& \sigma^2 \sum_{q=0}^{\tau-1} k_{\text{RBF}}(\boldsymbol{x}_{t_1-q}, \boldsymbol{x}_{t_2-q}) + \\ & \sum_{q_1=0}^{\tau-1} \sum_{q_2 \neq q_1} \frac{1}{|q_1 - q_2| + 1} k_{\text{RBF}}(\boldsymbol{x}_{t_1-q_1}, \boldsymbol{x}_{t_2-q_2}) \end{aligned} \tag{C.27}$$

For a test sample $\boldsymbol{x}_{t_*}$ collected at time t_*, the DRBF kernel is defined for $X_{t_1:t_1-\tau+1}$ and $X_{t_*:t_*-\tau+1}$:

$$\begin{aligned} k_{\text{DRBF}}(X_{t_1:t_1-\tau+1}, X_{t_*:t_*-\tau+1}) =& \sigma^2 \sum_{q=0}^{\tau-1} k_{\text{RBF}}(\boldsymbol{x}_{t_1-q}, \boldsymbol{x}_{t_*-q}) + \\ & \sum_{q_1=0}^{\tau-1} \sum_{q_2 \neq q_1} \frac{1}{|q_1 - q_2| + 1} k_{\text{RBF}}(\boldsymbol{x}_{t_1-q_1}, \boldsymbol{x}_{t_*-q_2}) \end{aligned} \tag{C.28}$$

The kernel function k_{DRBF} can be used for calculating the kernel matrix $K^{(\text{DRBF})}$ and KPCA can be applied for feature extraction and fault detection.

Fault detection

The procedure of fault detection remains the same when the DRBF kernel is applied in KPCA. Assuming that the kernel PCs $\boldsymbol{z}^{(\text{DRBF})}$ are extracted using $K^{(\text{DRBF})}$, the SPE at time t_1 can be defined in a similar way as the definition given by Eqn (5.15):

$$\text{SPE}_{t_1} = k_{\text{DRBF}}(X_{t_1:t_1-\tau+1}, X_{t_1:t_1-\tau+1}) - 2\bar{K}_j^{(\text{DRBF})} + \bar{K}^{(\text{DRBF})} - \boldsymbol{z}_j^{(\text{DRBF})^\top} \boldsymbol{z}_j^{(\text{DRBF})} \tag{C.29}$$

It should be noted that $k_{\text{DRBF}}\left(X_{t_1:t_1-\tau+1}, X_{t_1:t_1-\tau+1}\right)$ is not necessarily equal to one for two identical groups of lagged data. When applied to test samples, the SPE at time t_* is

$$\text{SPE}_{t_*} = k_{\text{DRBF}}\left(X_{t_*:t_*-\tau+1}, X_{t_*:t_*-\tau+1}\right) - 2\bar{K}_*^{(\text{DRBF})} + \bar{K}^{(\text{DRBF})} - \boldsymbol{z}_*^{(\text{DRBF})^\top}\boldsymbol{z}_*^{(\text{DRBF})} \tag{C.30}$$

where $\boldsymbol{z}_*^{(\text{DRBF})}$ are the kernel PCs obtained for $X_{t_*:t_*-\tau+1}$, the lagged data obtained at time t_*. $\bar{K}_*^{(DRBF)} = 1/T\sum_{t=1}^{\top} k_{\text{DRBF}}(\boldsymbol{x}_t, \boldsymbol{x}_*)$ is the mean of the kernel vector obtained using the training data X and the lagged test data $X_{t_*:t_*-\tau+1}$.

C.3 Comparison with other dynamic adaptation of RBF-KPCA

There are various ways to extend KPCA to a dynamic approach. For example, the dynamic adaptation of RBF-KPCA given by Choi and Lee (2004) extended the dynamic formulation of PCA to RBF-KPCA. Ku et al. (1995) proposed the dynamic formulation of PCA by augmenting the sample vector. If one sample with m variables $\boldsymbol{x}_t = [x_{t,1}, x_{t,2}, \cdots, x_{t,m}]^\top \in \mathbb{R}^{m\times 1}$ is measured at time t, the augmented sample $\boldsymbol{x}_t^{(A)}$ with a time lag τ is defined as

$$\boldsymbol{x}_t^{(A)} = [x_{t,1}, \cdots, x_{t,m}, x_{t-1,1}, \cdots, x_{t-1,m}, \cdots, x_{t-\tau+1,m}]^\top \in \mathbb{R}^{m\tau\times 1} \tag{C.31}$$

Therefore, if the original data matrix $X = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_T\} \in \mathbb{R}^{m\times T}$ have T samples, the augmented data matrix $X^{(A)}$ is:

$$X^{(A)} = \{\boldsymbol{x}_\tau^{(A)}, \cdots, \boldsymbol{x}_T^{(A)}\} \in \mathbb{R}^{m\tau\times(T-\tau)} \tag{C.32}$$

Ku et al. (1995) formulated dynamic PCA by applying PCA to the augmented matrix $X^{(A)}$ so that the dynamic kernel features are linear combinations of lagged samples.

Choi and Lee (2004) made KPCA dynamic by augmenting the data matrix X. The kernel function $k(\boldsymbol{x}_{t_1}^{(A)}, \boldsymbol{x}_{t_2}^{(A)})$ is defined for $\boldsymbol{x}_{t_1}^{(A)}$ and $\boldsymbol{x}_{t_2}^{(A)}$, the two augmented samples at time t_1 and T_2. Thus the entry of the dynamic kernel matrix $K^{(\text{DKPCA})} \in \mathbb{R}^{(T-\tau)\times(T-\tau)}$ is

$$K_{t_1,t_2}^{(\text{DKPCA})} = k\left(\boldsymbol{x}_{t_1}^{(A)}, \boldsymbol{x}_{t_2}^{(A)}\right) \tag{C.33}$$

When the RBF kernel is used,

$$k_{\text{RBF}}\left(\boldsymbol{x}_{t_1}^{(A)}, \boldsymbol{x}_{t_2}^{(A)}\right) = \exp\left(-\frac{\left(\boldsymbol{x}_{t_1}^{(A)} - \boldsymbol{x}_{t_2}^{(A)}\right)^\top \left(\boldsymbol{x}_{t_1}^{(A)} - \boldsymbol{x}_{t_2}^{(A)}\right)}{\delta^2}\right) \tag{C.34}$$

where

$$\left(\boldsymbol{x}_{t_1}^{(A)} - \boldsymbol{x}_{t_2}^{(A)}\right)^\top \left(\boldsymbol{x}_{t_1}^{(A)} - \boldsymbol{x}_{t_2}^{(A)}\right) = \sum_{q=0}^{\tau-1} \left(\boldsymbol{x}_{t_1-q} - \boldsymbol{x}_{t_2-q}\right)^\top \left(\boldsymbol{x}_{t_1-q} - \boldsymbol{x}_{t_2-q}\right) \tag{C.35}$$

Eqn (C.34) therefore is

$$\begin{aligned} k_{\text{RBF}}\left(\boldsymbol{x}_{t_1}^{(A)}, \boldsymbol{x}_{t_2}^{(A)}\right) &= \exp\left(-\sum_{q=0}^{\tau-1} \frac{\left(\boldsymbol{x}_{t_1-q} - \boldsymbol{x}_{t_2-q}\right)^\top \left(\boldsymbol{x}_{t_1-q} - \boldsymbol{x}_{t_2-q}\right)}{\delta^2}\right) \\ &= \prod_{q=0}^{\tau-1} \exp\left(-\frac{\left(\boldsymbol{x}_{t_1-q} - \boldsymbol{x}_{t_2-q}\right)^\top \left(\boldsymbol{x}_{t_1-q} - \boldsymbol{x}_{t_2-q}\right)}{\delta^2}\right) \\ &= \prod_{q=0}^{\tau-1} k_{\text{RBF}}(\boldsymbol{x}_{t_1-q}, \boldsymbol{x}_{t_2-q}) \end{aligned} \tag{C.36}$$

It can be seen that the dynamic kernel proposed in Section C.2 and the kernel obtained under the dynamic KPCA framework given by Choi and Lee (2004) are different. The first kernel, as shown in Eqn (C.27), considers the autocorrelation of the pairs of samples with the same time lag ($\boldsymbol{x}_{t_1-q}$ and $\boldsymbol{x}_{t_2-q}$) and the autocorrelation of the pairs of samples with different time lags ($\boldsymbol{x}_{t_1-q_1}$ and $\boldsymbol{x}_{t_2-q_2}$ such that $q_1 \neq q_2$). The second kernel in Eqn (C.36) only considers the autocorrelation of the pairs of samples with the same time lag. Therefore, the monitoring models built by applying the first kernel can be more accurate than the models built by the second kernel by definition. For example, for two consecutive groups of lagged samples $X_{t_1-1:t_1-\tau} = [\boldsymbol{x}_{t_1-1}, \ldots, \boldsymbol{x}_{t_1-\tau}]$ and $X_{t_1:t_1-\tau+1} = [\boldsymbol{x}_{t_1}, \ldots, \boldsymbol{x}_{t-\tau+1}]$, the second to the τ-th samples in $X_{t_1:t_1-\tau+1}$ are identical to the first to the $\tau-1$-th samples in $X_{t_1-1:t_1-\tau}$. If a kernel considers the autocorrelation of the pairs of samples with both the same and different time lags, then the correlation between $X_{t_1:t_1-\tau+1}$ and $X_{t_1:t_1-\tau+1}$ calculated by this kernel will be high, which is true because these two groups are consecutive.

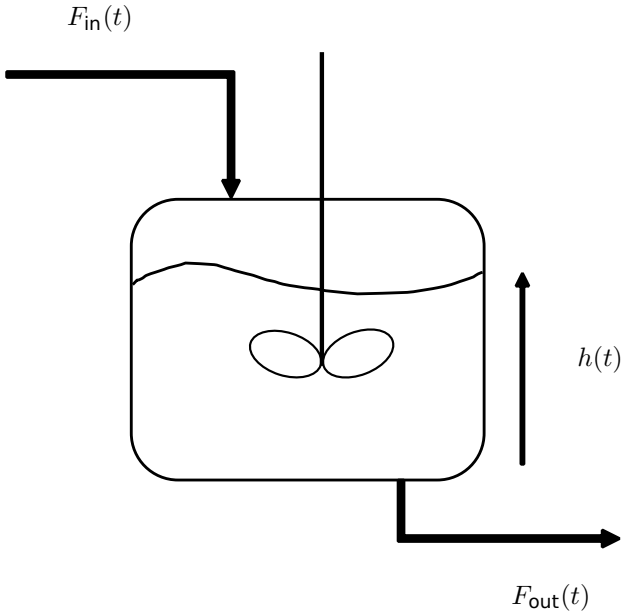


Figure C.1: Illustrative figure of the water tank

C.4 Preliminary results on a simulated example

Process description

This section uses the following example with a tank. Fig. C.1 presents the layout of the tank.

The first principles model of the tank can be written as:

$$\frac{\mathrm{d}h}{\mathrm{d}t} = \frac{1}{A\rho}\left(F_{\text{in}}(t) - F_{\text{out}}(t)\right) \tag{C.37}$$

$$F_{\text{out}}(t) = k'\sqrt{h(t)} \tag{C.38}$$

where A is the area of the tank. h is the level in the tank. F_{in} and F_{out} are the volumetric flow rates entering and existing the tank, respectively. Eqn (C.37) describes how the level h changes with respect to the inlet and outlet flow rates. F_{out} is defined by Eqn (C.38) because the outlet flow is assumed turbulent. ρ is the water density. k' is a parameter which regulates the outlet flow rate. The outlet flow rate is nonlinearly correlated with the water level in the tank. Disturbances may exist in F_{in} and noise may exist in F_{out} and h. This first principles model is used to generate simulated data.

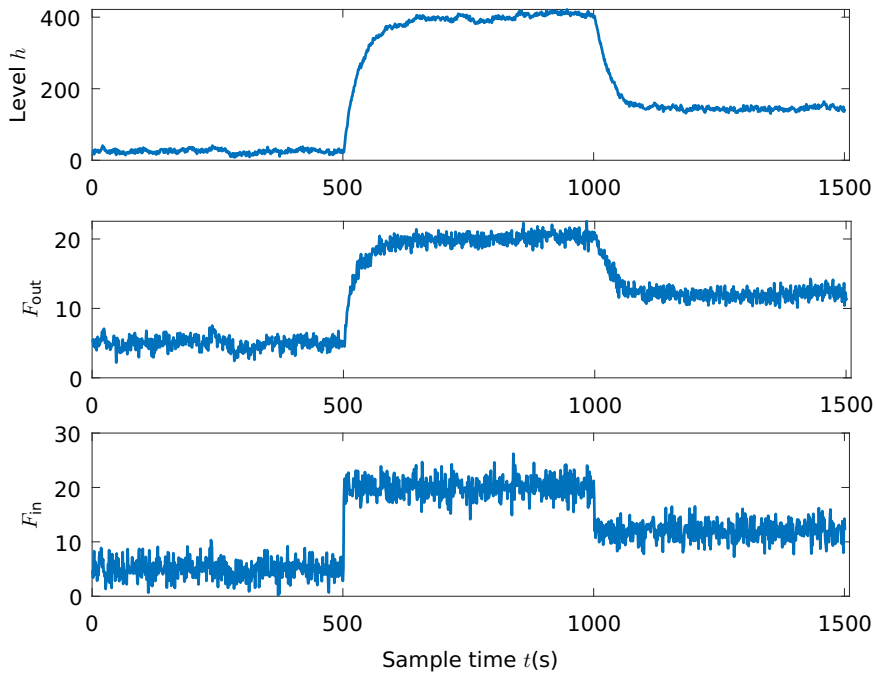


Figure C.2: Trend plots of the training data from the simulated example

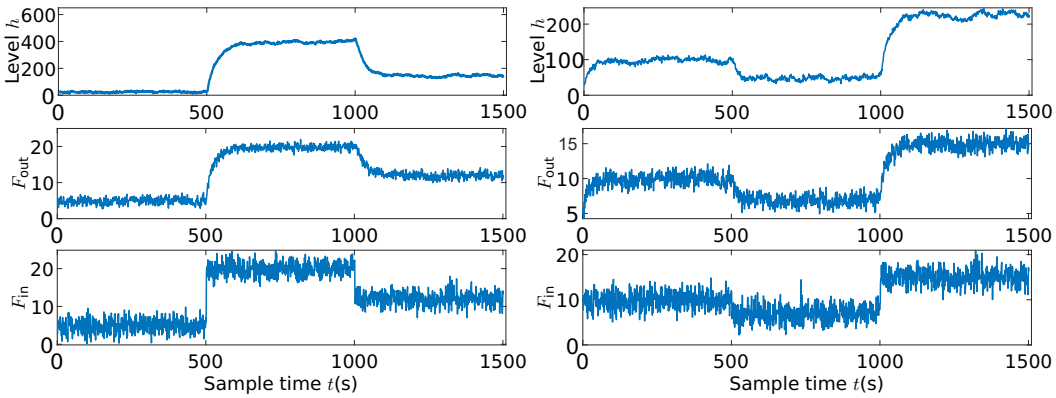
This process is nonlinear because the relationship between the outlet flow rate F_{out} and the tank level h is nonlinear. The process is also dynamic because the level h at time t depends on the inlet and outlet flow rates before time t.

This process will stabilize at a steady state when the set-point of F_{in} is constant. The first principal model can be linearized around this steady state. The linearized model will be different if the steady state is different. The various linearized models is an example of multiple operating modes. However, the underlying nonlinear model is the same for all operating modes achieved by adjusting the set-point of F_{in}. The process will move to a new steady state after an operator changes the set-point of F_{in}. Transition periods will occur after the change before the process stabilizes again.

The first principles model is used to generate data for testing the dynamic kernel methods. The training dataset has 1500 samples. Fig. C.2 gives the time trends of the level h, the outlet flow rate F_{out} and the inlet flow rate F_{in} in the training dataset. There are three operating modes specified by F_{in}. Initially a period of steady state operation in the first operating mode was simulated for 500 seconds, then the second operating mode was simulated for 500 seconds. Finally the process moved to the third mode and the simulation ran for 500 seconds. The inlet flow F_{in} has disturbance while the level h has measurement noise.

Table C.1: Tested cases in the simulated example

No.	Tested case	Description
V1	Validation data	The random sequences for noise and disturbance of the three variables changed.
V2	Changed steady states	The set-points of F_{in} are different from the set-points in Fig. C.2.
C1	Drifting outlet flow F_{out}	F_{out} starts to reduce slowly because of an additive ramp error at $t = 750$s.
C2	Drifting parameter k'	k' starts to increase slowly at $t = 750$s.
C3	Drifting tank level F_{out} with changed steady states	F_{out} starts to reduce slowly because of an additive ramp error at $t = 750$s. The set-points of F_{in} are the same as the set-points used in case V2.



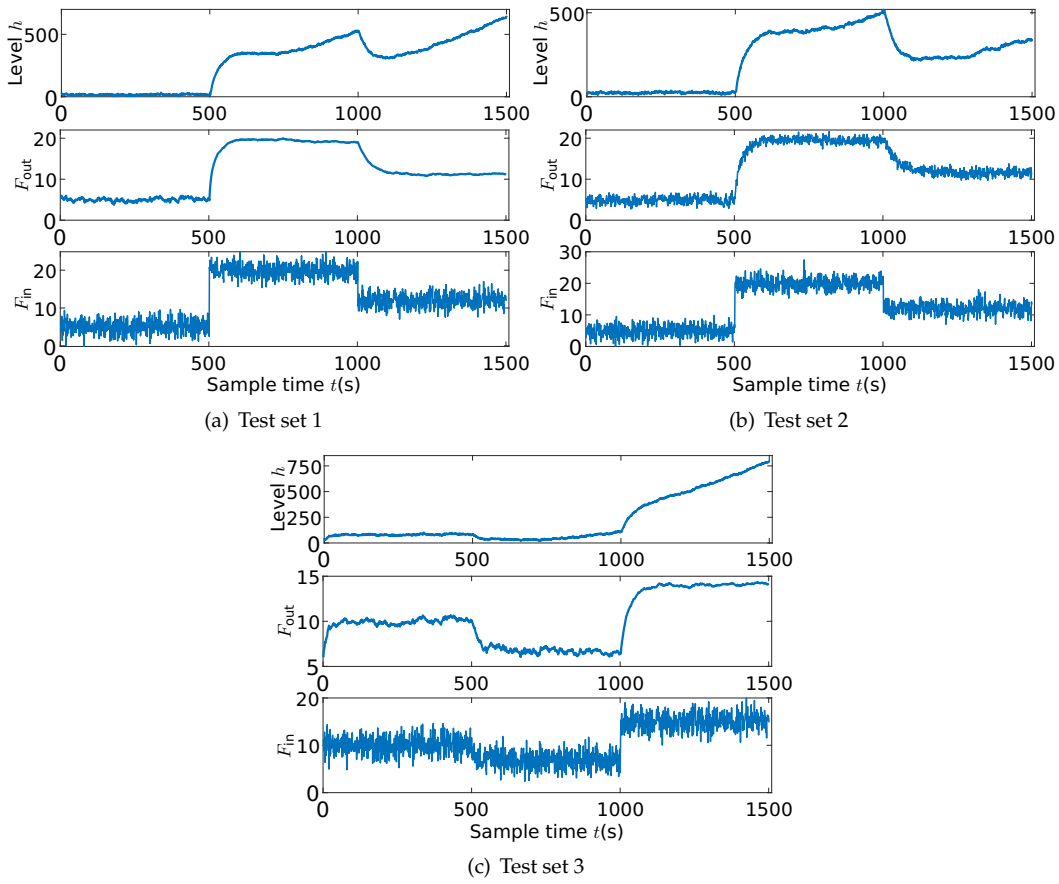
(a) Validation set 1 (b) Validation set 2

Figure C.3: Trend plots of two validation datasets from the simulated example

Table C.1 summarizes several tested scenarios for validating the monitoring model and for testing the performance of fault detection. Each tested case generate 1500 samples.

Fig. C.3 plots the time trends of the three variables for the tested cases V1 and V2. These two cases are for validation because the underlying nonlinear model of the process does not change. A dynamic method for process monitoring should be able to acknowledge the data in these two cases as healthy although the steady states in V2 were not included in the training data.

Fig. C.4 plots the time trends of the three variables obtained for the tested cases C1, C2, and C3. These cases are considered as faulty because the behaviour of the process variables have changed. In reality, a blockage in the outlet pipeline may lead to a reduction in F_{out} or an increase in the value of parameter k'. As the blockage develops, k' and F_{out} continue to reduce. Hence the tested cases use ramp errors to simulate the development of blockage.



(a) Test set 1

(b) Test set 2

(c) Test set 3

Figure C.4: Trend plots of the three test datasets from the simulated example

A monitoring method should be able to detect the occurrence of the fault when the fault is not significant. In particular, the method should be able to distinguish between the change in the steady states and the change of the behaviour of process variables. The challenging period is the period from the 1000th to around the 1200th sample in Fig. 4(b). This is because of the mode change. The developing blockage and the transition to the new operating mode coexist in the process for this period. Since the set-point of the inlet flow rate F_{in} reduces, the outlet flow F_{out} also reduces. This is a healthy behaviour of the process. In the meantime, F_{out} gradually reduced because of the drifting in k', which is a faulty behaviour. The change in F_{out} due to the fault may not be obvious in the time trends when compared with the change in F_{out} due to the operating mode. A monitoring method should detect the fault because, although the change of operating modes result in a similar behaviour of the process variables to the behaviour caused by the fault, the first principles model changes due to the fault.

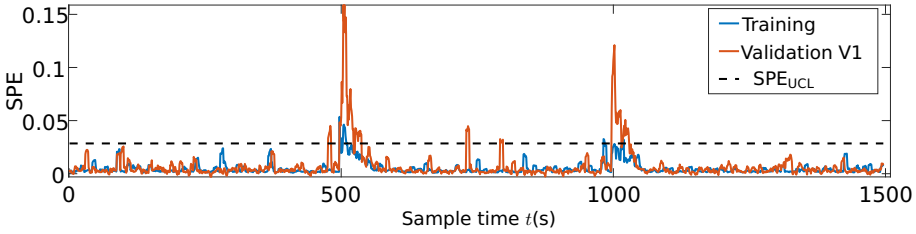
Results

The DRBF kernel proposed in Section C.2 is applied to the datasets. The time lag τ is set as five seconds. The number of kernel PCs are retained such that 99% of variability is explained by the retained kernel PCs. The same criterion applies when the RBF kernel is used. Both methods use SPE as the monitoring statistic and the confidence interval of the control limit of SPE is set to be 99%.

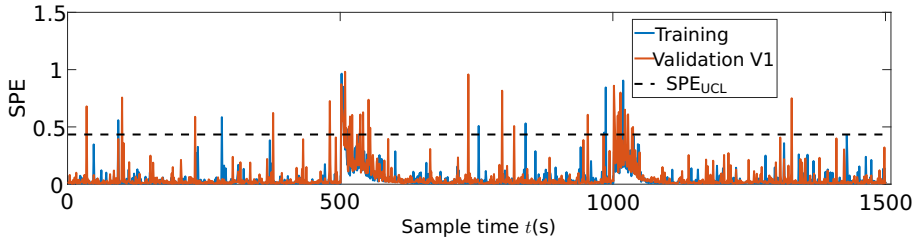
The validation results obtained by KPCA with the RBF kernel and KPCA with the DRBF kernel are compared in Figs C.5 and C.6. Fig. C.5 shows that the RBF kernel and DRBF kernel are robust to mode changes when all the modes are known in the training data. However, the RBF kernel is sensitive to noise and disturbance and the SPE obtained by the RBF kernel is noisy with jumps, as shown in Fig. 5(b). Such behaviour of SPE will result in a relaxed upper control limit SPE_{UCL} in order to reduce false alarms caused by the jumps. This can lead to missed alarms when a fault occurs. In contrast, the SPE obtained by the DRBF kernel is smooth. The transition period has an influence on the control limits in both methods because kernels still react to the transition periods caused by the step changes in the set-points of F_{in}.

Fig. 6(a) demonstrates that the magnitude of SPE obtained by the DRBF kernel in the new modes is similar to the magnitude of SPE in the training data. This shows that the DRBF kernel can account for new modes that do not exist in the training data if the underlying model of the process remains the same. The RBF kernel results in changes in the magnitude of SPE when the operating mode changes. The behaviour is particularly obvious for the first 500 samples and for the last 500 samples in Fig. 6(b). The relaxed SPE_{UCL} for RBF-KPCA leads to increased missed alarms. The sensitivity to faults can be enhanced by neglecting the peaks in SPE and selecting a smaller SPE_{UCL}. Nevertheless, such selection will lead to false alarms when new operating modes appear, such as the period starting from the 1000th sample in Fig. 6(b).

Figs C.7-C.9 present the monitoring result of KPCA with the DRBF kernel on the tested cases C1-C3. It can be seen that the proposed DRBF kernel can detect the faults caused by drifting values of the outlet flow rate F_{out} and the parameter k' soon after the fault was induced (the 750th sample). In particular, the method detects the period between the 1000th sample and the 1200th sample in Fig. C.8 as faulty even if the influence of the change in the operating mode is dominant. Since DRBF-KPCA is a dynamic approach, it

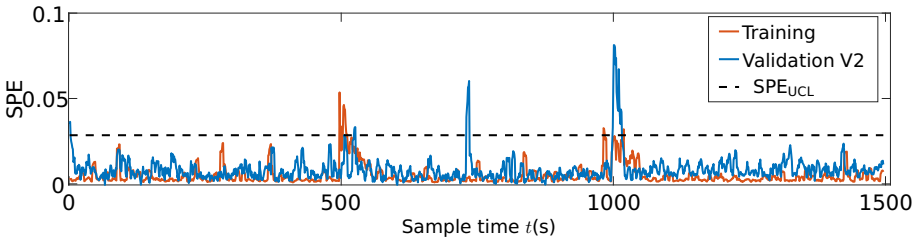


(a) DRBF: V1

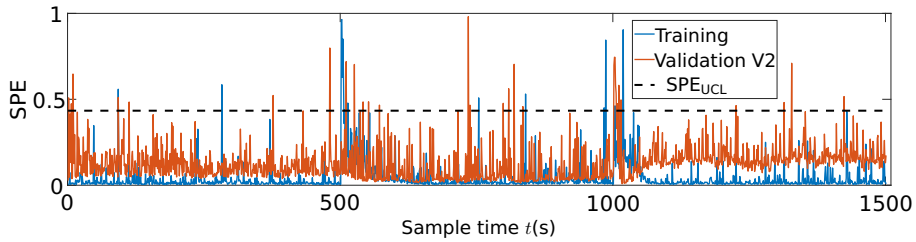


(b) RBF: V1

Figure C.5: Comparison between KPCA with RBF kernel and KPCA with DRBF kernel



(a) DRBF: V2



(b) RBF: V2

Figure C.6: Validation result of V2

is not necessary for the monitoring statistic to exceed the control limit for a sequence of samples. Instead, fault detection occurs when the SPE exceeds SPE_{UCL}. False alarms still exist for the transition periods between operating modes.

C.5 Research directions for nonlinear dynamic process monitoring

The future research nonlinear dynamic process monitoring can be based on the DRBF kernel. Several topics need to be investigated in order to make the DRBF kernel useful for

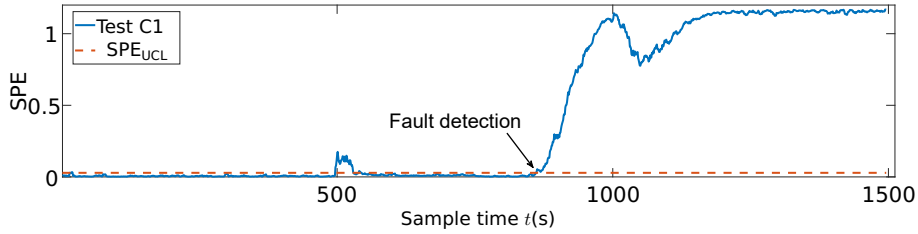


Figure C.7: Test result of C1

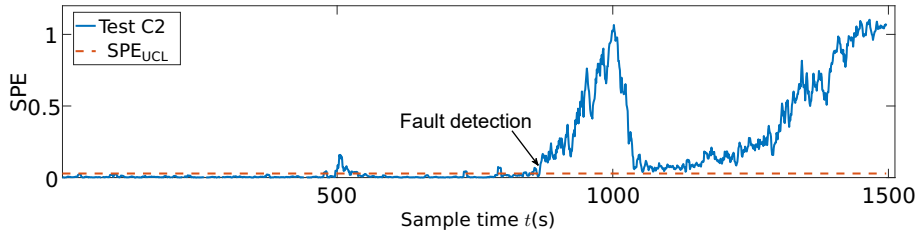


Figure C.8: Test result of C2

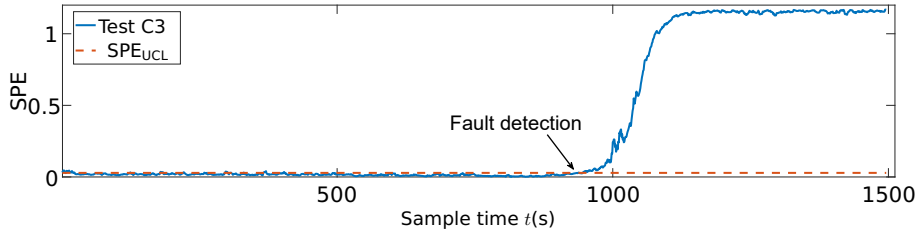


Figure C.9: Test result of C3

monitoring nonlinear and dynamic processes. The roadmap starts with further development of the DRBF kernel and moves towards the comparison with other methods and the validation of the DRBF kernel. The future direction of kernel-based process monitoring will be the design of customized kernels that are suitable for processes with specific characteristics.

Tuning the DRBF kernel

The coefficients α_{q_1,q_2} represent how the autocorrelation between two samples decays as the difference between the time lags increases. Section C.2 gave an example of manually specifying α_{q_1,q_2}. It may also be possible to estimate the values of α_{q_1,q_2} by fitting the model to the training data. This makes it possible to optimize the coefficients α_{q_1,q_2} in order to improve the performance of KPCA with the proposed DRBF kernel in fault detection.

The selection of the time lag τ exists in methods for dynamic process monitoring. In the DRBF kernel, the time lag determines the number of basis functions used for constructing the kernel, as shown in Eqn (C.22). It is necessary to have a certain lag for such that the

DRBF-KPCA model can capture the autocorrelation in time trends of variables. It is also necessary to avoid an excessively large τ value. The first reason is that the autocorrelation in process data often decays as the time lag increases, and the autocorrelation can be neglected when the time lag is large enough. The second reason is that a large value of τ lead to a large number of basis functions and a large number of coefficients to be tuned. There is a risk of over-fitting. Furthermore, the method may be computationally expensive because all τ samples in $X_{t_1-\tau+1:t_1}$ need to be stored and used for fault detection at time t_1.

The time lag τ and the coefficients α_{q_1,q_2} should be tuned jointly because the autocorrelation in time trends of process data often decays as the time lag increases.

Comparison with other methods

It is important to compare the performance of these methods using simulated and real-life examples. Furthermore, it is more important to compare the formulations of these methods and the assumptions made by these methods. Section C.3 is an example of comparing the DRBF kernel and the RBF kernel in the dynamic adaptation of KPCA proposed by Choi and Lee (2004). To summarize, the comparison with other methods should include the following aspects:

1. Comparison of the assumptions made when formulating each of the methods;
2. Comparison of the mathematical formulations and the procedures used for the methods;
3. Comparison of monitoring performance on simulated examples and real-life datasets.

Validation and practical considerations

The simulated example is an example of nonlinear dynamic process. Linearization of the first principles model at some steady states will obtain multiple linear models. Therefore, such a process is multimodal because of linearization at the operating modes. In Figs 6(a) and C.9, the DRBF kernel shows an ability of extrapolation between operating modes such that false alarms are not triggered for the healthy operating modes that were not seen in the training data. It is ideal to have such behaviour for multimodal process monitoring because, if the underlying process dynamic does not change, a monitoring system with

such methods may be able to automatically acknowledge new operating modes without referring to expert knowledge of operators.

Such ability needs to be further verified by testing on other examples of nonlinear and dynamic processes. In particular, the test may take several steps by using the following examples in each step:

1. Simulated examples such as the stirring tank heater presented by Thornhill et al. (2008);
2. Complex simulated examples such as the Tennessee Eastman process (Downs and Vogel, 1993);
3. Real-life datasets such as the PRONTO benchmark dataset.

One task of validating a method is to identify the scenarios where the method is applicable and the scenarios where the method could fail. The DRBF kernel proposed here aims to account for the autocorrelation existing due to the time constant of processes. The performance of this kernel needs to be tested when control loops exist in the process and when measurement noise is also autocorrelated. Moreover, techniques developed for tuning this kernel should be validated together with the kernel using the steps mentioned before.

Further development of the kernel

The way of formulating kernels as covariance functions enable the formulation of a variety of kernels. New kernels can be formulated in order to account for sophisticated behaviour of the process. For example, a process may have a nonlinear first principles relationship while the parameters in the first principles relationship are varying over time. The DRBF kernel may need to be revised in order to build monitoring models for such processes.

The way forward is to organize the basis functions in a way that suits the assumptions of the process. By doing so, it is possible to develop new kernels that are useful for processes with specific behaviour.