CRANFIELD UNIVERSITY

RAPHAEL TARI SAMUEL

NONLINEAR DYNAMIC PROCESS MONITORING USING KERNEL METHODS

OIL AND GAS ENGINEERING CENTRE
SCHOOL OF WATER, ENERGY AND ENVIRONMENT (SWEE)

PhD

Academic Year 2015-2016

Supervisors: Dr Yi Cao
Dr Giorgos Kopanos

August 2016

CRANFIELD UNIVERSITY

OIL AND GAS ENGINEERING CENTRE
SCHOOL OF WATER, ENERGY AND ENVIRONMENT (SWEE)

PhD Thesis

Academic Year 2015-2016

RAPHAEL TARI SAMUEL

NONLINEAR DYNAMIC PROCESS MONITORING USING KERNEL METHODS

Supervisors: Dr Yi Cao
Dr Giorgos Kopanos

August 2016

This thesis is submitted in partial fulfilment of the requirement for the degree of Doctor of Philosophy

Abstract

The application of kernel methods in process monitoring is well established. However, there is need to extend existing techniques using novel implementation strategies in order to improve process monitoring performance. For example, process monitoring using kernel principal component analysis (KPCA) have been reported. Nevertheless, the effect of combining kernel density estimation (KDE)-based control limits with KPCA for nonlinear process monitoring has not been adequately investigated and documented. Therefore, process monitoring using KPCA and KDE-based control limits is carried out in this work. A new KPCA-KDE fault identification technique is also proposed.

Furthermore, most process systems are complex and data collected from them have more than one characteristic. Therefore, three techniques are developed in this work to capture more than one process behaviour. These include the linear latent variable-CVA (LLV-CVA), kernel CVA using QR decomposition (KCVA-QRD) and kernel latent variable-CVA (KLV-CVA).

LLV-CVA captures both linear and dynamic relations in the process variables. On the other hand, KCVA-QRD and KLV-CVA account for both nonlinearity and process dynamics. The CVA with kernel density estimation (CVA-KDE) technique reported does not address the nonlinear problem directly while the regular kernel CVA approach require regularisation of the constructed kernel data to avoid computational instability. However, this compromises process monitoring performance.

The results of the work showed that KPCA-KDE is more robust and detected faults higher and earlier than the KPCA technique based on Gaussian assumption of process data. The nonlinear dynamic methods proposed also performed better than the afore-mentioned existing techniques without employing the ridge-type regularisation.

Acknowledgements

The PhD route could be long and tortuous. It would not have been possible to get to this point without the support of several good spirited people. I therefore use this opportunity to express my profound gratitude to some of them.

I thank my supervisors, Dr Yi Cao and Dr Giorgos Kopanos. The professionalism exhibited by Dr Cao and his attention to details contributed immensely to the completion of this work. I thank him so dearly. Although, my contact hours with Dr Kopanos were few because he joined the supervisory team late in my research, yet his suggestions on the soft issues were very helpful.

I will also like to specially thank Sam Skears and Samara Ahmad for their assistance in administrative and immigration issues respectively. I am also grateful to my office mates, friends and colleagues: Richard, Nonso, Reward and Cristobal for making my stay at Cranfield memorable. I will not also forget to thank Rev. Olabiyi Ajala, Dr. Sola Adesola, Dr Crispin Allison and the Holding Forth/CPA family for their support during my stay at Cranfield.

I also wish to express my heart-felt gratitude to my wife, Tariere, my children, Kuro and Mercy, and my extended family for their sacrifice throughout my research work. Above all, thanks be to the Almighty God for His love and tender mercies. I dedicate this work to His Name and glory.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The subject of the research work reported in this thesis is the development and testing of kernel-based multivariate statistical algorithms for monitoring nonlinear dynamic processes. This introduction chapter provides the background to the work and attempts to address how the work fits into the broader context of the process monitoring and control discipline.

As a result of technological development, modern process facilities have become larger, more complex and highly integrated. At the same time, the regulations that govern their operations are now more stringent. Therefore, the need to operate these facilities in an efficient but sustainable manner has become more challenging and critical.

Although process controllers are able to compensate for a number of disturbances that occur in a process, there are some process changes or faults which controllers cannot handle adequately (Russel et al., 2000). These include faulty actuators or analysers, contaminated sensors, leaks, clogged filters, changing feedstock properties, degraded catalysts, etc. If they are not detected and corrected in time, these faults can cause equipment malfunction or failure, unscheduled plant or unit shut-downs, poor product quality, industrial accidents, devastating environmental impacts and huge financial losses.

In large complex industrial plants with automatic data acquisition facilities, several variables are measured and data are recorded very frequently. Therefore, the total amount of data collected during routine plant operations have increased dramatically. The sheer volume of data makes it difficult for human operators to react to faults appropriately. Thus, the benefits gained from closer process management due to increased data availability can often be offset by losses arising from time spent in dealing with unexpected situations. Furthermore, the large volume of data or information generated from the plethora of process measurements has also increased the pressure on human operators to make very important and complex decisions often within a very short interval of time. However, information overload can make human operators prone to making decisions and taking actions that make things even worse in their attempt to correct faults. Incidents like Three Mile Island, Bhopal, and Chernobyl (Lees, 2005) are tragic examples of faults that turned into disasters, partly due to wrong actions on the part of operators, who were probably overwhelmed by too much information. Hence, the development of effective process monitoring techniques that enable automated fault detection and diagnosis in industrial systems is desirable.

Proper process monitoring will ensure timely detection of abnormal situations and give room for early intervention. This will improve safety, product quality, safeguard the environment and enhance overall system reliability. Prevention of equipment malfunctions or failures and associated cost and downtime will improve economic savings significantly and increase profitability. These incentives have spurred the study and development of automatic process monitoring methods starting from the early 1970s (Isermann and Ballé, 1997). Furthermore, research in data-driven process monitoring methods have received much attention in the last 25 years resulting in the development of several multivariate statistical methods (Saxen et al., 2013; MacGregor and Cinar, 2012; Yin et al., 2014; Ding et al., 2013; Dai and Gao, 2013).

1.1 Process monitoring tasks

Process monitoring is the checking of measurable variables against tolerances and raising alarms for operator action when a tolerance is exceeded (Isermann, 2005). The goal of process monitoring is to detect, identify and diagnose faults timely so that appropriate actions are taken to remove assignable cause(s) while the process is still controllable. Consequently, process monitoring is associated with the following tasks (Raich and Cinar, 1996): fault detection, fault identification, fault diagnosis and process recovery or intervention (Fig. 1.1).

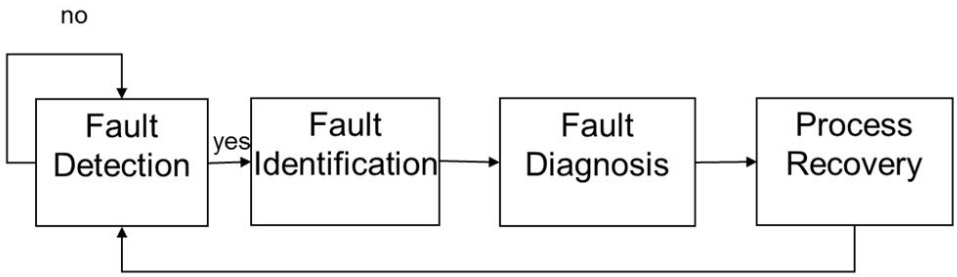


Figure 1.1: Flowchart showing four general process monitoring tasks. When fault is detected, the variables associated with fault are first identified and the process is recovered after determining the source of the fault (fault diagnosis) and removing it.

- ***Fault detection***: determining the occurrence of a fault.
- ***Fault identification***: identifying the variables immediately impacted by a fault.
- ***Fault diagnosis***: determining the source of the fault.
- ***Process recovery or intervention***: removing the effect of the fault.

It is necessary to note here, that the terminology associated with process monitoring lacks consistency in the fault detection and diagnosis (FDD) literature. For instance, Isermann and Ballé (1997) defines fault diagnosis as the determination of the nature, time, locality and extent of a given fault. An alternative viewpoint is that fault diagnosis consists of fault isolation and fault identification (see Gonzalez

and Castanon, 2011, pg. 99). According to this viewpoint, fault isolation is the determination of the faulty component while fault identification is the determination of the magnitude of the fault. In this context, the term fault detection and isolation (FDI) is adopted when the identification task is not deemed to justify the effort. In some cases, "diagnosis" is used only as a synonym to "isolation" (Gertler, 1998, pg. 3).

1.2 Motivation of the research

Techniques based on multivariate statistics are well suited for monitoring large complex processes. These approaches which are generally referred to as multivariate statistical process monitoring (MSPM) methods or statistical process control (SPC) can be used to process multidimensional data and account for correlation or redundant information in data. MSPM approaches are more effective and more efficient than univariate methods. Univariate methods deal with only one variable at a time. Therefore, they lack the ability to describe relationships between variables in a dataset. Furthermore, multivariate statistical techniques can be used to perform dimensionality reduction. Hundreds or even thousands of highly correlated variables can be reduced to a few latent variables without sacrificing critical information. The lower dimensional data can be further reduced to three, two or even one monitoring measure(s). This simplifies the monitoring process and improves working conditions by helping operators to focus on fewer variables to monitor.

Nevertheless, traditional MSPM techniques such as principal component analysis (PCA) (Wold et al., 1987; Jolliffe, 2002) and partial least squares (PLS) (Wilson and Irwin, 2000; Muradore and Fiorini, 2012) assume that the process being monitored is linear and static. On the contrary, nonlinearity and dynamics exist widely in the process industry. Therefore, traditional MSPM techniques perform poorly in practice (Yin et al., 2012). Hence, there is need to develop process monitoring algorithms that effectively capture process nonlinearity and dynamics in order to improve monitoring performance (Chen, 2013; Yang et al., 2012). This need serves

as part of the motivation for this research work.

To address the nonlinearity problem, traditional techniques such polynomials, splines, and neural networks (NNs) have been used (Mathews, 1991; Wold, 1992; Kramer, 1992; Haykin, 1999). However, many of these approaches involve iterative nonlinear solution methods and/or are computationally expensive. In particular, NNs suffer from long-time training, slow convergence and local minima (Gou and Fyfe, 2004). More recently, kernel methods, have gained popularity as an attractive framework for tackling nonlinear problems (Schölkopf and Smola, 2002; Shawe-Taylor and Cristianini, 2004). The key principle of kernel methods which is also the main motivation for using kernel methods in this work is the *kernel trick*.

The kernel trick is based on the fact that many data processing approaches depend on the inner products between data samples; not on the individual data samples. It is therefore possible to develop a nonlinear extension of a linear algorithm by mapping the original data into a high-dimensional feature space via a nonlinear mapping and reformulating the algorithm in a way that needs only values of the inner products in the feature space. In kernel methods, the inner products of the mapped samples in the high dimensional feature space are defined by using a kernel function of the corresponding samples in the original data space (Qin, 2012; Honeine and Richard, 2011a). Hence, kernel algorithms are very efficient and do not involve high computational complexity.

Valid kernels can be constructed for even non vectorial data such as strings and graphs by simply replacing the classical inner product by an appropriate similarity measure for the data. Therefore, kernel methods have extended the use of classical algorithms to many situations where data cannot be readily represented in a vectorial form by directly working with pairwise distances or similarities between non-vectorial objects (Duin et al., 1998).

Kernel methods were proposed by Vapnik in Support Vector Machines, SVMs (Vapnik, 2000) but are now employed in classification (Mika, Ratsch, Weston, Schölkopf and Muller, 1999), regression (Rosipal and Trejo, 2002), bioengineering (Camps-

Valls et al., 2006), and image de-noising (Mika, Schölkopf, Smola, Müller, Scholz and Rätsch, 1999). Successful application of kernel methods have also been reported in time series prediction (Richard et al., 2009), novelty detection (Schölkopf et al., 1999), and process condition monitoring (Lee et al., 2004; Choi et al., 2005; Tan et al., 2010).

1.3 Research gaps

Although kernel-based process monitoring is not new, the need for extending existing approaches and developing alternative implementation strategies to improve monitoring performance still exist. For example, a number of nonlinear process monitoring studies using kernel PCA (KPCA) have been reported. Nevertheless, the effect of combining kernel density estimation (KDE)-based control limits with KPCA for nonlinear process monitoring has not been adequately investigated and documented. Hence, a study on process monitoring using KPCA and KDE-based control limits is carried out in this work. In particular, the performance and robustness of KPCA-KDE-based process monitoring is determined and the results obtained are compared with results obtained with KPCA and control limits based on the assumption of normally distributed process data.

Furthermore, due to the nonlinear transformation involved in kernel methods, till date, fault identification is still an unsolved problem in kernel-based nonlinear process monitoring (Deng et al., 2013). The techniques reported in the literature are computationally expensive and difficult to generalise. Consequently, a new KPCA-KDE-based fault identification process is proposed in this thesis.

More importantly, very limited research has been reported on the use of kernel methods in nonlinear dynamic process monitoring. The dynamic principal component technique proposed by Choi and Lee (2004) does not capture process dynamics adequately. Conversely, canonical variate analysis (CVA) is reported to be an efficient multivariate approach for monitoring dynamic systems (Ruiz-Cárcel et al.,

2015) but it does not address nonlinearity. The CVA with KDE technique proposed by Odiowei and Cao (2010) to adapt the CVA for nonlinear dynamic process monitoring did not address the nonlinear problem directly. On the other hand, directly applying the kernel canonical correlation analysis (KCCA) algorithm to dynamic systems result in singular kernel matrices which require regularisation in order to avoid potential computational instabilities (Huang et al., 2009; Schölkopf and Smola, 2002; Giantomassi et al., 2014). Furthermore, such an approach often leads to poor process monitoring performance.

To address the above problems, two new kernel-based methods are proposed in this thesis for nonlinear dynamic process monitoring. These techniques address both nonlinearity and process dynamics directly and do not require the determination of an optimum regularisation parameter value to perform well.

1.4 Aim and objectives

The main aim of this work is to develop and test kernel-based multivariate statistical algorithms for improved nonlnear dynamic process monitoring. Specifically, the objectives of this research are to:

1. Study the effect of combining kernel density estimation (KDE)-based confidence limits with KPCA for nonlinear process monitoring instead of using confidence limits based on the Gaussian assumption.

2. Develop a novel kernel-based approach for fault identification in a nonlinear process.

3. Develop the linear latent variable-CVA (LLV-CVA) approach for monitoring linear dynamic processes.

4. Develop a new kernel CVA technique based on QR decomposition.

5. Develop the kernel latent variable-CVA (KLV-CVA) approach for monitoring nonlinear dynamic processes.

6. Evaluate the fault detection performance of the techniques developed.

7. Carry out comparison study of the developed approaches with existing methods.

1.5 Publications

Four conference papers and one journal article have resulted from this work. A second journal paper has been submitted.

Conference papers

Samuel, R.T. and Cao, Y. (2014), Fault detection in a multivariate process based on kernel PCA and kernel density estimation, 20$^{\text{th}}$ International Conference on Automation and Computing (ICAC) Cranfield, Bedfordshire, United Kingdom, September 12-13, pp. 146-151.

Samuel, R. T. and Cao, Y. (2015), Kernel canonical variate analysis for nonlinear dynamic process monitoring, 9$^{\text{th}}$ International Symposium on Advance Control of Chemical Processes, Whistler, British Columbia, Canada, June 7-10, pp. 606-611. (*This paper was awarded as the best presentation paper at the conference*).

Samuel, R.T. and Cao, Y. (2015), Improved Kernel Canonical Variate Analysis for Process Monitoring, 21$^{\text{st}}$ International Conference on Automation and Computing (ICAC), University of Strathclyde, Glasgow, UK, September 11-12, pp. 341-346.

Samuel, R. T. and Cao, Y. (2016), Dynamic latent variable modelling and fault detection of Tennessee Eastman challenge process, IEEE International Conference on Industrial Technology (ICIT), Taipei, Taiwan, March 14-17.

Journal articles

Samuel, R. T. and Cao, Y. (2016), Nonlinear process fault detection and identification using kernel PCA and kernel density estimation, Systems Science and Control Engineering, 4(1), 165-174.

Samuel, R. T. and Cao, Y., Kernel latent variable CVA for nonlinear dynamic process monitoring, IEEE International Transaction on Industrial Informatics - *Submitted*.

1.6 Thesis outline

This thesis consists of seven chapters. The first two chapters are introduction and an overview of process monitoring methods respectively. Chapters 3 to 6 contain individual algorithms developed in this thesis. These chapters employ the kernel principle except Chapter 4 which is a linear dynamic method. All of the four chapters are presented in a journal publication style commonly used in the process monitoring literature - introduction, methodology and application study. The application study section covers algorithm testing, results/discussion and concluding remarks. Chapter 7 summarises the conclusions drawn from the work and highlights recommendations for future work. A summary of each of the seven chapters is presented below.

Chapter 1: Introduction

In Chapter 1 the background and motivation of the thesis is presented. The gap this research work seeks to address, the aims and objectives of the work are also presented in this chapter. The chapter ends with a thesis outline.

Chapter 2: Overview of process monitoring methods

Chapter 2 gives an overview of process monitoring methods. Some basic definitions and the theory of kernel functions is also presented in this chapter.

Chapter 3: Nonlinear process fault detection and identification using kernel PCA and kernel density estimation

In Chapter 3, the kernel KPCA with KDE technique is developed and evaluated. Fault detection and identification performance as well as robustness of the technique are assessed and compared with KPCA based on the Gaussian assumption.

Chapter 4: Statistical process monitoring using linear latent variable CVA

It is possible for a process to posses both linear and dynamic properties. A technique capable of capturing both of these properties is therefore necessary. In this chapter, the linear latent variable technique (LLV-CVA) is developed to address this case. The effectiveness of the techniques is assessed and compared with the DPCA and CVA using the TE process.

Chapter 5: Kernel canonical variate analysis using QR decomposition

Chapter 5 is dedicated to the development of the kernel CVA technique based on QR decomposition. The approach is also tested on the Tennessee Eastman process and the results are presented.

Chapter 6: Kernel latent variable CVA for nonlinear dynamic process monitoring

Chapter 6 addressed the development of kernel latent variable CVA (KLV-CVA) for monitoring nonlinear dynamic processes. The technique is tested on the three difficult faults (3, 9 and 15) of the TE process. The performance of the technique is compared with the traditional KCVA based on KCCA and the kernel dynamic PCA (KDPCA).

Chapter 7: Conclusions and future work

This chapter summarizes how the objectives of the research were fulfilled. The contributions and limitations of this work are also presented in this chapter. The chapter ends with some recommendations for future work.

Chapter 2

Overview of Process Monitoring Methods and Positive Definite Kernels

An overview of process monitoring methods is presented in this chapter. The chapter also provides some relevant definitions and the theory of kernel functions - the key ingredient of kernel methods. The concept of reproducing kernel Hilbert spaces, nonlinear mapping and the feature space, and the general implementation strategy implied in kernel methods are discussed. Common examples of kernels and kernel-based algorithms are also presented.

Process monitoring methods may be classified into three categories: data-based, knowledge-based and model-based (Chiang et al., 2001). An elaborate discussion and description of these categories is captured in a three-part series by Venkatasubramanian and other researchers published in 2003. The different monitoring categories are reviewed in this section.

2.1 Model-based methods

Model-based or analytical approaches are based on explicit mathematical models of the monitored plant (Isermann, 1984). These fundamental models describe the internal dynamics and explain the behaviour of the process based on physical, chemical or biological laws. Most model-based fault detection and diagnosis methods are based on the concept of *analytical redundancy*. This involves comparing measured performance or outputs with analytically predicted or estimated outputs. It can also mean comparing values of two analytically computed quantities from different sets of variables. This is in contrast to *physical redundancy* which involves comparing measurements obtained from several sensors. In analytical redundancy methods, the occurrence of a fault is captured by the difference between the plant output and the model prediction or estimate (that is, the residual). Thus, residuals can be thought of as "artificial signals" that reflect possible faults of the process. Techniques such as parameter estimation (Isermann, 1993), observer based design (Frank, 1990) or parity relations (Xiong et al., 2006) are used to generate the residuals.

The analytical redundancy approach in model-based fault detection and diagnosis consists of two key stages: residual generation and residual evaluation (see Fig. 2.1). In the residual generation stage, the difference between the system and model outputs is generated. On the other hand, in the residual evaluation stage, the decision rules used for analysing the residuals to arrive at the monitoring decisions are chosen. This stage is essentially the decision making stage.

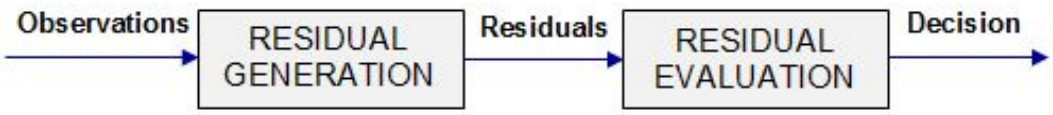


Figure 2.1: Flowchart showing the stages of a model-based fault detection and diagnosis procedure. Residuals are generated from the difference between system and model outputs. The residuals are then evaluated using specified rules for a decision to made whether or not a fault has occurred.

Due to the effect of noise and modelling errors, the residuals may not be zero even

if no fault exists. Hence, residuals are usually tested against empirically or theoretically derived thresholds to detect the presence of faults. The four residual generation methods associated with model-based process monitoring listed in Chapter 1 are now presented.

2.1.1 Kalman filter

The generation of residuals by Kalman filters for fault detection and diagnosis was introduced by Mehra and Peschon (1971). This technique is a recursive data processing algorithm that uses series of measurements observed over time which contain statistical noise and inaccuracies to estimate unknown variables. The difference between the process measurement and the algorithm output (prediction error) is used to monitor the process. Fault diagnosis is done by carrying out statistical tests on whiteness, mean and covariance of the residuals (Hwang et al., 2010). Statistical tests based on the Kalman filter framework are easy to construct because the residuals form uncorrelated time-series. Nevertheless, fault diagnosis is believed to be awkward as it requires running one filter for every suspected fault and for every possible arrival time. In addition, filter outputs need to be checked to identify the one that can be matched with the actual observations (Gertler, 1998).

2.1.2 Observers

An observer is a tool used to estimate the internal states of a system based on the measured inputs and outputs. The key concern of this method is to generate sets of residuals that detect and identify different faults in a unique way. This is achieved by designing observers which are sensitive to a different subset of faults and insensitive to other faults and the unknown inputs. Essentially, under normal operating conditions, the observer estimates will closely follow the plant output resulting in a small residual which reflects the unknown inputs only. However, when a fault occurs, observers that have been designed to be insensitive to the

subset of faults will maintain small residuals which reflect the unknown inputs only. Nonetheless, observers that are sensitive to the faults will give results that differ significantly from the process giving rise to large residuals. The unique residual patterns generated by different observers for different faults enhances the monitoring process. However, the models may involve complex computations (Chiang et al., 2001; Gertler, 1991).

2.1.3 Parity relations

This technique involves checking the parity or consistency of the mathematical equations of the system with the actual measurements. A fault is declared to have occurred when the preassigned threshold is exceeded. The parity relations can be subjected to a linear dynamic transformation for the transformed residuals to be used for dynamic process monitoring (Gertler, 1997). The design freedom arising from the transformation can be used to enhance the monitoring process. It has been shown that once the design objectives are selected, parity equations and observer based designs lead to equivalent residual generators (Gertler, 1991).

2.1.4 Parameter estimation

This is a natural approach to monitoring parametric faults. It involves building a reference model under a fault-free condition. Then, the parameters are re-identified repeatedly and the results obtained are compared with the reference model. Deviations from the reference model (that is, the residual) is then used as the basis for fault detection and diagnosis. This approach can be adopted only if a relation between process faults and changes in the model parameters exist and if suitable mathematical models are available (Chiang et al., 2001).

Mathematical models have been used for process monitoring and diagnosis for many decades (Campos-Delgado and Espinoza-Trejo, 2011; Rothenhagen and Fuchs, 2009). Since they are based on solid physical or engineering principles and prior

process knowledge, they give the most accurate results if they are well formulated. However, they can be complex and computationally intensive. Furthermore, detail understanding of all the phenomena at play in a complex large-scale process may be lacking. Therefore, significant amount of time and money is required to establish reliable quantitative models. Sometimes it is even infeasible to build one (Ge et al., 2013).

According to Katipamula and Brambley (2005), model-based process monitoring methods are not likely to emerge as the methods of choice in the near future due to the drawbacks enumerated earlier.

2.2 Knowledge-based methods

Knowledge-based approaches adopt the use of qualitative models e.g. causal analysis, expert systems or pattern recognition to derive the monitoring measures (Venkatasubramanian, Rengaswamy and Kavuri, 2003). No explicit system models are required for these approaches and their monitoring results are very intuitive. They are therefore used as complementary methods or as alternatives to model-based approaches. However, to a large extent, knowledge-based methods depend on previous knowledge of the behaviour of the process and the expertise of experienced plant operators. Therefore, it takes considerable time and effort to acquire the needed knowledge and expertise to routinely construct knowledge-based models for large-scale complex systems. Due to their weaknesses, model-based and knowledge-based methods are limited to relatively small systems; or systems for which it is easy to build the needed process model; or system for which adequate accumulation of the pertinent process knowledge is available.

To determine process status, knowledge-based methods employ prior knowledge of the process expressed in terms of qualitative relationships. This is in contrast to model-based methods discussed earlier in which prior process knowledge is expressed in terms of quantitative mathematical relationships. Therefore, process monitoring

methods based on knowledge-bases are also called qualitative model-based methods. These models can be obtained by modelling the causal relationships that exist in the system, using expert domain knowledge, and from detailed system descriptions or fault-system relationships (Chiang et al., 2001).

2.2.1 Causal analysis

Causal analysis methods are based on modelling the causal relationship between faults and symptoms in a system. Signed directed graph (SDG) is an example of causal analysis technique used mainly for fault diagnosis.

SDG is based on analysing initial and final responses of system variables due to deviations and deducing these dynamic behaviours using causal path propagations. SDG was first proposed for modelling chemical processes by Iri et al. (1979). A documentation of a systematic framework for developing and analysing SDG-based modelling has been made by Maurya et al. (2003).

It is also possible to draw conclusions on the overall system behaviour based on an understanding of the laws that govern the various subsystems. This approach is commonly called *abstraction hierarchies*. Its application in the process industry is well documented in Venkatasubramanian, Rengaswamy and Kavuri (2003).

2.2.2 Expert systems

The idea behind this approach is to mimic how a human expert will reason when diagnosing a fault. This is done by combining knowledge gained from first principles (or structural description of the process) with rules formulated from the experience of a domain expert. They are basically if-then-else statements and a mechanism that searches through the rule-space to arrive at conclusions deployed as a software package. These rules can be based purely on expert domain knowledge gained from experience or from first principles. Expert systems can capture diagnostic associations identified by humans which cannot be easily expressed in the form of

mathematical or causal relationships. The use of expert systems in process industries is described in detail in Venkatasubramanian, Rengaswamy, Kavuri and Yin (2003).

Expert systems are supposedly based on transparent reasoning and clear explanations can be provided for inferences made (Venkatasubramanian, Rengaswamy and Kavuri, 2003). However, they are system specific and fail woefully when operated outside the incorporated boundaries. They are also not easy to change or update.

2.3 Data-based methods

Data-based methods require only input and output data collected from routine process operations. These data are transformed in several ways (a process known as feature extraction) and used as prior knowledge of the monitored system. Due to their data-driven nature, neither rigorous system models nor detail process knowledge is required in data-based techniques. Therefore, they are simpler to build for complex large-scale systems than the model-based or knowledge-based approaches. Furthermore, historical data collected from processes are readily available and powerful data mining techniques for extracting process knowledge from measurement information are well understood.

Part III of the three-part review articles by Venkatasubramanian and other researchers published in 2003 cited in the preceding sections is dedicated to database process monitoring methods. An in-depth outlook of data-based monitoring approaches is also provided by Ge et al. (2013). Their work reviewed the natures of industrial processes, data characteristics (e.g. high dimensionality, nonlinear data relationships, non-Gaussian variable distributions, time-varying and multi-mode behaviours and data autocorrelations) as well as methodology issues and implementation procedures. Figure 2.2 shows a generalised methodology for MSPM approaches.

Collecting the dataset that correctly represent the operating conditions of the process to be monitored is an important initial step in MSPM techniques. This is

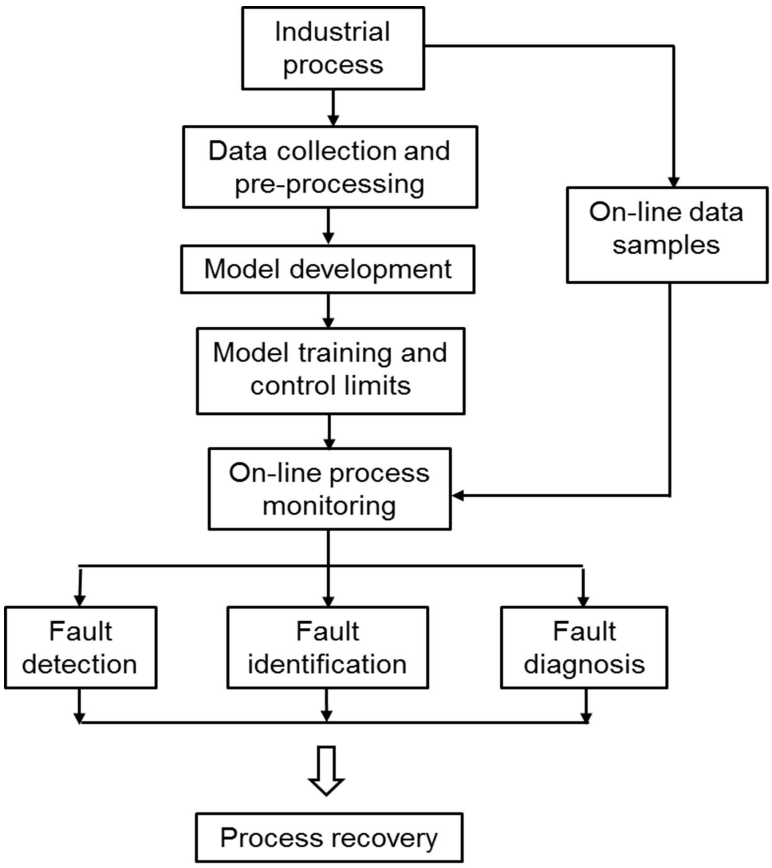


Figure 2.2: Flowchart showing a typical MSPM procedure. Normal operating data are collected and pre-processed (e.g. normalised to zero mean and unit variance). The monitoring model is developed and trained, and control limits are computed for on-line monitoring. A faulty process is recovered after successful fault detection, identification and diagnosis.

needed to avoid numerous false alarms or missed detections which are indications of an inefficient monitoring technique. Pre-processing involves transforming the original dataset to a form more appropriate for developing a reliable monitoring model. A common example of pre-processing is normalising the original data to zero means and unit variance. This is done to eliminate the influence of the different scales of the various process variables to avoid undue inclination of a given model to any one of the measured process variables. The appropriate process monitoring model is then selected based on the data characteristics of the process. The model is then trained and evaluated to ascertain its efficiency. Model training is done off-line using data collected under normal operating conditions. During the training phase, ap-

propriate statistics used for online monitoring are constructed and their thresholds or control limits are computed.

To construct the monitoring statistics, the original process variables spanning a high dimensional space are projected onto a lower dimensional space spanned by q dominant latent variables which adequately capture the relevant information in the original high dimensional data. The q latent variables are then used to derive the model for monitoring the model space (that is, the space spanned by q latent variables). Conversely, the latent variables that are not included in the model space are regarded as noise and are used to monitor the residual space (that is, the space spanned by the latent variables not included in the model space). Hotelling's T^2 and the Q statistic (also known as squared prediction error, SPE) are two indices commonly used in MSPM approaches. The T^2 is used to monitor variation in the model space while the Q statistic is used to monitor variation in the residual space. Some traditional data-driven MSPM approaches and their extensions are discussed in the following subsections.

2.3.1 Principal component analysis

Principal component analysis (PCA) is probably the most popular of the MSPM methods. Assuming a dataset with n number of observations and m variables $\mathbf{X} \in \Re^{m \times n}$ which is mean-centred and have unit variance, the sample covariance matrix is computed as

$$\mathbf{C} = \frac{1}{m-1}\mathbf{X}^T\mathbf{X}, \tag{2.1}$$

where the subscript, T represents transpose. The eigenvalue decomposition of $\mathbf{C}$ given by

$$\mathbf{C} = \mathbf{PLP}^T, \tag{2.2}$$

where $\mathbf{P}$ are the principal directions (or loading vectors) and $\mathbf{L}$ is the eigenvalue matrix whose i^{th} element represents the variation present in the data projected in the direction of the i^{th} component. Assuming that $\mathbf{P} \in \Re^{m \times q}$ are the matrix of

scores or linear latent variables associated with the dominant q singular values. The projections of the original variables in $\mathbf{X}$ into the lower dimensional space is defined as

$$\mathbf{Z} = \mathbf{XP} \Longrightarrow \hat{\mathbf{X}} = \mathbf{ZP}^T = \sum_{i=1}^{q} z_i p_i^T, \tag{2.3}$$

where z_i is an orthogonal score vector which captures the relationship between observations. Conversely, p_i is an orthonormal loading vector containing information about relationship between variables. Given that q principal components explain the variability of the process data through $\mathbf{X}$, the residual matrix $\mathbf{E}$ that captures the variations associated with the $m - q$ singular values is given by $\mathbf{X} - \hat{\mathbf{X}}$. Hence,

$$\mathbf{X} = \mathbf{ZP}^T + \mathbf{E} = \sum_{i=1}^{q} z_i p_i^T + \mathbf{E}. \tag{2.4}$$

Since $\mathbf{ZP}^T$ and $\mathbf{E}$ represent the main sources of process variability (useful information) and noise (error) respectively, the choice of q (that is, the number of PCs retained) is very important in PCA-based process monitoring. Retaining too few components gives an under-fitted model. Interpreting data analysis under such a situation implies relating only to the most dominant part of the data structure. Therefore, some significant information of the data structure will not be captured. Conversely, using too many components result in an over-fitted model which runs the risk of interpreting parts of the noise in the data. A number of methods have been suggested for determining q. These include scree tests, the average eigenvalue approach, cross-validation, parallel analysis, Akaike information criterion, and the cumulative percent eigenvalue. However, none of these methods have been proved analytically to be the best in all situations (Chiang et al., 2001).

PCA-based monitoring statistics

The Hotelling's T^2 and Q statistic or squared prediction error (SPE) are commonly used in PCA-based process monitoring. Hotelling's T^2 is the sum of the normalised

squared scores. It is computed as

$$T^2 = [z_1, \ldots, z_q] \Lambda^{-1} [z_1, \ldots, z_q]^T , \tag{2.5}$$

where q are the number of PCs retained and Λ^{-1} represents the inverse of the matrix of eigenvalues corresponding to the retained PCs. The control limit corresponding to a significance level, α, T^2_α is derived from the F-distribution analytically as

$$T^2_\alpha \sim \frac{q\,(m-1)}{m-q} F_{q,m-q,\alpha}, \tag{2.6}$$

$F_{q,m-q,\alpha}$ is the value of the F-distribution corresponding to a significance level, α, with degrees of freedom q and $m-q$ for the numerator and denominator respectively.

On the other hand, the squared prediction error or Q statistic is computed as

$$Q = e^T e = \left(I - PP^T\right) x, \tag{2.7}$$

where e is the residual vector (a projection of the observation $\mathbf{x}$ into the residual space) and I is the identity matrix. The upper confidence limit for the Q-statistic is computed from its approximate distribution as follows (Jackson, 1991):

$$Q_\alpha = \theta_1 \left[\frac{C_\alpha h_0 \sqrt{2\theta_2}}{\theta_1} + 1 + \frac{\theta_2 h_0 \,(h_0 - 1)}{\theta_1^2} \right]^{\frac{1}{h_0}} , \tag{2.8}$$

where, $\theta_i = \sum_{j=q+1}^{n} \lambda_j^i, (i = 1, 2, 3)\,, h_0 = 1 - \frac{2\theta_1\theta_3}{3\theta_2^2}$, λ_i are the eigenvalues, and C_α is the $100(1-\alpha)$ normal percentile.

PCA is essentially a dimensionality reduction technique based on a single collection of variables. However, situations often arise where two sets of variables $(\mathbf{X}, \mathbf{Y})$ are considered in multivariate statistical analysis. Methods used to handle such data include but not limited to multiple linear regression (MLR), PLS and CCA. MLR has severe problems dealing with large sets of correlated data. Apart from being cumbersome, MLR may lead to imprecise parameter estimates and poor predictions.

Conceptually, PLS is similar to PCA except that it reduces the dimensions of the two set of variables simultaneously to find their latent vectors which have the highest correlation using an iterative approach. A number of the techniques employed to address the gaps identified in this thesis (LLV-CVA, KCVA-QRD and KLV-CVA) are based on the principle underlying the formulation of CCA. Therefore, a discussion on the CCA is presented in the next subsection.

2.3.2 Canonical correlation analysis

Canonical correlation analysis was first proposed by Hotelling (1936). The goal of CCA is to identify and assess linear relations existing between two multivariate data sets by finding linear combinations of the original variables which maximise the correlation between the combinations. The linear combinations are called canonical variates while the pairwise correlations are known as canonical correlations. The strength of the association between the two sets of variables is measured by the canonical correlations.

Given two random variable vectors, $\mathbf{x} \in \Re^p$ and $\mathbf{y} \in \Re^p$, their linear combinations are defined by, $u = \mathbf{x}^T\mathbf{a}$ and $v = \mathbf{y}^T\mathbf{b}$, where $\mathbf{a}$ and $\mathbf{b}$ are combination coefficient vectors. Canonical correlation analysis seeks to find $\mathbf{a}$ and $\mathbf{b}$ such that the correlation between u and v is maximised. Numerically, let $\mathbf{X} \in \Re^{N\times p}$ and $\mathbf{Y} \in \Re^{N\times p}$ be N samples of $\mathbf{x}$ and $\mathbf{y}$, respectively. Assuming expectations, $E(\mathbf{x}) = \boldsymbol{\mu}_1$ and $E(\mathbf{y}) = \boldsymbol{\mu}_2$. Define

$$\overline{\mathbf{X}} = \begin{bmatrix} \boldsymbol{\mu}_1^T \\ \vdots \\ \boldsymbol{\mu}_1^T \end{bmatrix}, \quad \overline{\mathbf{Y}} = \begin{bmatrix} \boldsymbol{\mu}_2^T \\ \vdots \\ \boldsymbol{\mu}_2^T \end{bmatrix}.$$

Covariance and cross-covariance matrices are defined as

$$\boldsymbol{\Sigma}_{xx} = E\left\{\left(\mathbf{X}-\overline{\mathbf{X}}\right)\left(\mathbf{X}-\overline{\mathbf{X}}\right)^T\right\}, \tag{2.9}$$

$$\boldsymbol{\Sigma}_{yy} = E\left\{\left(\mathbf{Y}-\overline{\mathbf{Y}}\right)\left(\mathbf{Y}-\overline{\mathbf{Y}}\right)^T\right\}, \tag{2.10}$$

$$\boldsymbol{\Sigma}_{xy} = E\left\{\left(\mathbf{X}-\overline{\mathbf{X}}\right)\left(\mathbf{Y}-\overline{\mathbf{Y}}\right)^T\right\}. \tag{2.11}$$

where E denotes expectation. In this case, correlation is given by:

$$\rho = \max_{\mathbf{a},\mathbf{b}} \frac{\mathbf{a}^T\boldsymbol{\Sigma}_{xy}\mathbf{b}}{\left(\mathbf{a}^T\boldsymbol{\Sigma}_{xx}\mathbf{a}\right)^{1/2}\left(\mathbf{b}^T\boldsymbol{\Sigma}_{yy}\mathbf{b}\right)^{1/2}}. \tag{2.12}$$

Computing the standardised coefficients (weights) directly as $\mathbf{u} = \boldsymbol{\Sigma}_{xx}^{1/2}\mathbf{a}$ and $\mathbf{v} = \boldsymbol{\Sigma}_{yy}^{1/2}\mathbf{b}$, the CCA optimisation problem can be formulated as

$$\max_{\mathbf{u},\mathbf{v}} \quad \mathbf{u}^T\left(\boldsymbol{\Sigma}_{xx}^{-1/2}\boldsymbol{\Sigma}_{xy}\boldsymbol{\Sigma}_{yy}^{-1/2}\right)\mathbf{v}, \tag{2.13}$$

$$\text{s.t.} \quad \mathbf{u}^T\mathbf{u} = \mathbf{v}^T\mathbf{v} = 1, \tag{2.14}$$

where the solution, $\mathbf{u}$ and $\mathbf{v}$ are the left and right singular vectors of the product matrix $\mathbf{L}_1 = \boldsymbol{\Sigma}_{xx}^{-1/2}\boldsymbol{\Sigma}_{xy}\boldsymbol{\Sigma}_{yy}^{-1/2}$. Singular value decomposition can then be performed on $\mathbf{L}_1$ as shown below:

$$\mathbf{L}_1 = \boldsymbol{\Sigma}_{xx}^{-1/2}\boldsymbol{\Sigma}_{xy}\boldsymbol{\Sigma}_{yy}^{-1/2} = \mathbf{U}_1\mathbf{S}_1\mathbf{V}_1^T \tag{2.15}$$

where $\mathbf{U}_1$ and $\mathbf{V}_1$ are orthogonal matrices of the left and right singular vectors and $\mathbf{S}_1$ is a diagonal matrix whose elements are the singular values of $\mathbf{L}_1$. Sorting the elements of $\mathbf{S}_1$ in descending order and reordering the columns of $\mathbf{U}_1$ and $\mathbf{V}_1$, gives the degree of correlation between columns of $\mathbf{U}_1$ and $\mathbf{V}_1$.

MSPM approaches such PCA and CCA are static techniques. They are based on the assumption that the process data collected are time-independent. However, data generated from many real industrial processes exhibit time-dependence. Time-dependence means that an observation at the present time period is correlated with observations before and after the present time period (Yin et al., 2012). This phe-

nomenon is known as serial correlation, lagged correlation or autocorrelation. It is caused by system dynamics arising from process units that induce inertia (that is, tendency of a system to remain in the same state from one observation to the next), and high sampling rates used in modern data acquisition instrumentation (Rato and Reis, 2011; Vanhatalo and Kulahci, 2015).

Autocorrelation affects the number of independent observations. Therefore, the covariance matrix constructed from autocorrelated data without accounting for time lags cannot adequately represent the complete variation in the data. Hence, static techniques which are usually based on zero-lag covariance matrices give poor results when applied to autocorrelated data resulting from process dynamics (Jiang, Huang, Zhu, Fan and Braatz, 2015). Consequently, significant research efforts have been made in the past years to improve monitoring performance in dynamic industrial processes by incorporating the dynamic information of the process data into the monitoring model.

The simplest way to eliminate the effects of process dynamics is to increase the sampling time. This approach weakens the correlation between the data. However, it does not take into account the dynamic relationships that exist between the process variables. Hence, long sampling time reduces the sensitivity of monitoring systems and delay fault detection especially for faults that only cause changes in the time series correlation of the process variables. Therefore, other more effective methods for monitoring dynamic processes are presented next.

2.3.3 Dynamic Principal component analysis

Dynamic principal component analysis (DPCA) is an extension of the PCA technique that accounts for serial correlations. It involves augmenting each observation vector with the previous l observations and stacking the data matrix as follows:

$$\mathbf{X}_l = \begin{bmatrix} \mathbf{x}_t^T & \mathbf{x}_{t-1}^T & \dots & \mathbf{x}_{t-l}^T \\ \mathbf{x}_{t-1}^T & \mathbf{x}_{t-2}^T & \dots & \mathbf{x}_{t-l-1}^T \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}_{t+l-m}^T & \mathbf{x}_{t+l-1}^T & \dots & \mathbf{x}_{t-m}^T \end{bmatrix}, \tag{2.16}$$

where $\mathbf{X}_l$ is the augmented data matrix and x_t^T is the n-dimensional observation vector in the training data at a time period t. To extract the autocorrelation of the process data, the DPCA model is constructed by performing PCA directly on $\mathbf{X}_l$. The value of l can be determined statistically. However, according to Chiang et al. (2001), one or two lags are appropriate for DPCA-based process monitoring. The T^2 and Q statistics based on the DPCA are employed in a similar way as those from the conventional PCA for process monitoring in the model and residual spaces.

Due to their simplicity, dynamic MSPM methods are used in many cases with other developed methods. Ku et al. (1995) employed the DPCA for fault detection and isolation. Tsung (2000) examined an integrated approach to simultaneously monitor and diagnose an automatic controlled process using DPCA and minimax distance classifier. Similar to the DPCA, a dynamic version of the PLS (DPLS) was proposed by Komulainen et al. (2004) while application of DPCA and DPLS techniques in on-line monitoring of batch processes was reported by Chen and Liu (2002).

The above studies showed that dynamic MSPC methods outperformed their static counterparts. Nevertheless, it is argued dynamic MSPC methods provide only limited representation of process dynamics (Li et al., 2011; Russell et al., 2000). Kruger et al. (2004) also demonstrated that auto-correlated score variables will arise, even in the abstract case where the process variables are uncorrelated and a DPCA model is established. They therefore proposed the integration of ARMA models into a MSPM model. The results of case studies indicated that their approach extracts the autocorrelation of the process data successfully and provided better process monitoring performance in large-scale processes compared to the traditional dynamic MSPM

methods. The use of decorrelated residuals (Rato and Reis, 2013) have also been proposed to improve DPCA-based process monitoring. The CVA, a subspace modelling approach which identifies the state space model of a process, is another method widely reported for dynamic process monitoring (Jiang, Zhu, Huang, Paulson and Braatz, 2015). The CVA approach is discussed in the next subsection.

2.3.4 Canonical variate analysis

The CVA is an application of the CCA on time series data. It accounts for autocorrelation in addition to cross-correlation between variables by considering both the past and future process outputs at each time point (Wang et al., 2010; Odiowei and Cao, 2010). Assuming that $\mathbf{x}_{(k)} \in \Re^m$ are m process measurements (variables) at a given time point k acquired during normal operating process conditions. The past (p) and future (f) observation vectors are defined to capture the dynamics of the process as follows:

$$\mathbf{x}_{p(k)} = \begin{bmatrix} \mathbf{x}_{(k-1)} \\ \mathbf{x}_{(k-2)} \\ \vdots \\ \mathbf{x}_{(k-p)} \end{bmatrix} \in \Re^{mp} \text{ and } \mathbf{x}_{f(k)} = \begin{bmatrix} \mathbf{x}_{(t)} \\ \mathbf{x}_{(k+1)} \\ \vdots \\ \mathbf{x}_{(k+f-1)} \end{bmatrix} \in \Re^{mf} \tag{2.17}$$

The respective past and future observation vectors are then combined separately to obtain the past and future matrices $\mathbf{X}_p$ and $\mathbf{X}_f$. Defining $\tilde{\mathbf{X}}_p = \mathbf{X}_p - \overline{\mathbf{X}}_p$ and $\tilde{\mathbf{X}}_f = \mathbf{X}_f - \overline{\mathbf{X}}_f$ as the centred past and future matrices, where $\overline{\mathbf{X}}_p$ and $\overline{\mathbf{X}}_f$ are the sample means, the covariance and cross-covariance matrices of the past and future observations are obtained as:

$$\begin{aligned} \boldsymbol{\Sigma}_{pp} = E\left(\tilde{\mathbf{X}}_p \tilde{\mathbf{X}}_p^T\right), \ \boldsymbol{\Sigma}_{ff} = E\left(\tilde{\mathbf{X}}_f \tilde{\mathbf{X}}_f^T\right), \\ \boldsymbol{\Sigma}_{fp} = E\left(\tilde{\mathbf{X}}_f \tilde{\mathbf{X}}_p^T\right) \end{aligned} \tag{2.18}$$

The product matrix $\mathbf{H}_1$ from (2.18) is obtained and decomposed as follows:

$$\mathbf{H}_1 = \mathbf{\Sigma}_{ff}^{-1/2}\mathbf{\Sigma}_{fp}\mathbf{\Sigma}_{pp}^{-1/2} = \mathbf{U}_3\mathbf{S}_3\mathbf{V}_3^T \tag{2.19}$$

The linear combinations of the past observations that best explain the variability of the future observations are obtained by performing SVD on $\mathbf{H}_1$ in (2.19). In order to compute the monitoring statistics, the state variables (which span the model space) and the residuals are obtained from the transformed past observations. The T^2 and Q statistics are calculated at each time point as the sum of the squared state variables and residuals respectively. These computations are presented in chapters 4, 5 and 6.

The application of CVA in system identification was pioneered by Akaike (1975) and was adapted to general linear systems by Juricek et al. (2004). Several other successful applications of the CVA approach have also been reported over the years. These studies show that CVA captures process dynamics better and provides superior fault detection and diagnosis than other dynamic approaches that simply introduce time lags into collected measurements (Wang et al., 2010; Stubbs et al., 2012; Chen et al., 2014; Ruiz-Cárcel et al., 2015).

Notably, the MSPM approaches discussed so far are linear techniques. This implies that they do not consider or reveal process nonlinearities. Hence, the use of kernel-based techniques to address nonlinearity is explored in this work. However, some relevant definitions and the theoretical framework of kernels are presented next before the discussion on specific kernel-based monitoring approaches.

2.4 Positive definite kernels

The term *kernel* is used in different branches of mathematics. In linear algebra, it is used as a synonym for the nullspace of a linear operator. It is also used in the theory of integral operators. In conventional statistics, *kernel method* usually suggests kernel density estimation or Parzen window approach discussed in Chapter

3. In the context of this work, a kernel is a real-valued function which takes two arguments (vectors, real numbers, functions, etc.) and outputs a real number. The notation for this is $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$. In particular, the class of kernels used in this work are positive definite kernels. The insights provided here generally follow the expositions given by Berlinet and Thomas-Agnan (2004), Schölkopf and Smola (2002), and Shawe-Taylor and Cristianini (2004).

Definition 2.1 (Positive definite kernel). Let $\mathcal{X}$ be a non-empty set. A kernel is a positive definite (p.d.) kernel on $\mathcal{X}$ if it is symmetric, that is, $k(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_j, \mathbf{x}_i)$, and positive definite, that is,

$$\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) \geq 0, \tag{2.20}$$

for every $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N) \in \mathcal{X}$ and for every $(\alpha_1, \alpha_2 \ldots, \alpha_N) \in \mathbb{R}$, where $\mathbf{x}_i$ is a family of known points and α_i, is a family of real coefficients.

Definition 2.2 (Kernel matrix). Given a kernel k and any set of data points, $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N) \in \mathcal{X}$, the $N \times N$ matrix $\mathbf{K}$ (with elements $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ is the kernel matrix (also called Gram matrix) of k for $i, j = 1, \ldots, N$.

Definition 2.3 (Positive definite matrix). A real-valued $N \times N$ matrix $\mathbf{K}$ is positive definite if

$$\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j K_{ij} \geq 0, \tag{2.21}$$

for all $(\alpha_1, \alpha_2 \ldots, \alpha_N) \in \mathbb{R}$. This condition requires that $\boldsymbol{\alpha}^T \mathbf{K} \boldsymbol{\alpha} \geq 0$ for any $\boldsymbol{\alpha} \in \mathbb{R}^N$, where the superscript T represents transpose. This means that all the eigenvalues of the kernel matrix are non-negative. In practice, a positive definite kernel matrix is derived from a positive definite kernel function. Generally, kernel methods are algorithms that take kernel matrices as input.

2.4.1 Positive semi-definite and positive definiteness

Unfortunately, in the literature, there is no common use of the definitions given above. Some authors refer to functions that give sums that are non-negative as in (2.20) as *positive semi-definite or non-negative definite* functions. On the other hand, functions for which the sum in (2.20) is strictly positive,

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j k(\mathbf{x}_i,\mathbf{x}_j) > 0, \tag{2.22}$$

at least one α_i is non-zero are referred to as *positive definite* functions. The kernel matrix $\mathbf{K}$ derived from such a kernel is also termed a positive definite matrix. It is even argued that the correct mathematical terminology is to say that the kernel matrix associated to a positive definite matrix is positive semi definite (see Mohri et al., 2012, pg. 92). However, in numerical practice, some form of regularisation is carried out on the matrices to explicitly lower their condition number (the ratio of the biggest to the smallest eigenvalue of a matrix) in most estimation procedures. Therefore, definiteness and semi-definiteness will be equivalent. Hence, these two terms are not considered to be different in this work. In other words, the term positive definiteness is used for all kernels that comply with non-negativity.

The requirement for positive definiteness of kernels is important for at least two reasons. First, it is a major assumption in convex programming (Boyd and Vandenberghe, 2004). It ensures that algorithms converge to a unique solution. Second, positive definiteness is a key assumption from the functional analysis viewpoint of kernels in the theory of reproducing kernel Hilbert spaces (RKHSs).

2.4.2 Hilbert spaces

Kernels can also be viewed from the viewpoint of functional analysis because there is a Hilbert space $\mathcal{H}$ of real-valued functions on a set $\mathcal{X}$ to every kernel on $\mathcal{X}$. Hilbert spaces are briefly introduced in this subsection starting with the definition of an

inner product $\langle \cdot, \cdot \rangle$ which is a key concept in kernel methods - the space in which a kernel algorithm is supposedly performed is essentially an inner product space.

Definition 2.4 (Inner product space). Let $\mathcal{V}$ be a vector space over the scalar field $\mathbb{R}$. An inner product (also called scalar product or dot product) on $\mathcal{V}$ is a mapping $\langle \cdot, \cdot \rangle : \mathcal{V} \times \mathcal{V} \to \mathbb{R}$ for all $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{V}$ and for all $\alpha \in \mathbb{R}$. That is, an inner product takes each ordered pair of vectors $\mathbf{x}, \mathbf{y} \in \mathcal{V}$ and outputs a number. An inner product must satisfy the following properties:

(i) $\langle \mathbf{x} + \mathbf{y}, \mathbf{z} \rangle = \langle \mathbf{x}, \mathbf{z} \rangle + \langle \mathbf{y}, \mathbf{z} \rangle$

(ii) $\langle \alpha \mathbf{x}, \mathbf{y} \rangle = \alpha \langle \mathbf{x}, \mathbf{y} \rangle$

(iii) $\langle \mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{y}, \mathbf{z} \rangle$

(iv) $\langle \mathbf{x}, \mathbf{x} \rangle \geq 0$, for all $\mathbf{x} \in \mathcal{V}$ and $\langle \mathbf{x}, \mathbf{x} \rangle = 0$ if and only if $\mathbf{x} = 0$

Generally, the inner product of $\mathbf{x}, \mathbf{y} \in \mathcal{V}^M$ is defined by

$$\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{i=1}^{N} x_i y_i. \tag{2.23}$$

where $x_i, y_i, i, \ldots, N$, are the elements of vectors $\mathbf{x}$ and $\mathbf{y}$.

In a geometric sense, the inner product of two vectors of unit length gives a good notion of the angle between the two vectors by the formula

$$\cos \theta = \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{x}\| \|\mathbf{y}\|}.$$

Furthermore, the inner product defines the length (or norm) of a vector $\mathbf{x}$ as

$$\|\mathbf{x}\| = \langle \mathbf{x}, \mathbf{x} \rangle^{\frac{1}{2}}, \tag{2.24}$$

and the distance d between two vectors $\mathbf{x}$ and $\mathbf{y}$ by

$$d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\| = \langle \mathbf{x} - \mathbf{y}, \mathbf{x} - \mathbf{y} \rangle^{\frac{1}{2}}. \tag{2.25}$$

Therefore, geometric constructions can be formulated in terms of angles, lengths and distances, in an inner product space. An inner product space (also called pre-Hilbert space) is a vector space endowed with an inner product.

Definition 2.5 (Hilbert space). An inner product space that is complete with respect to the metric induced by the inner product is called a Hilbert space $\mathcal{H}$. Completeness means that every Cauchy sequence defined on the space converges to an element of the space (or has a limit which is a point in the space). Completeness is essential for achieving good convergence properties when dealing with infinite-dimensional Euclidean spaces. A sequence $\{x_n\}_{n=1}^{\infty}$ is a Cauchy sequence if for any real number $\epsilon > 0$ there exists a natural number N^* such that $d(x_n - x_m) < \epsilon$ for any $n, m \geq N^*$. Some examples of Hilbert spaces are:

- The vector space $\mathbb{R}^n$ endowed with an inner product $\langle \mathbf{a}, \mathbf{b} \rangle = \mathbf{a}^T\mathbf{b}$.
- The space l_2 of square-summable sequences, with inner product $\langle x, y \rangle = \sum_{i=1}^{\infty} x_i y_i$.
- The space L_2 of square-integrable functions (i.e., $\int f(x)^2 \mathrm{d}x < \infty$), with inner product $\langle f, g \rangle = \int f(x)g(x)\mathrm{d}x$.

2.4.3 Reproducing kernels

Let $\mathcal{H}_0 := f : \mathcal{X} \to \mathbb{R}$ be the space of functions that map $\mathcal{X}$ to $\mathbb{R}$. If $\mathcal{X}$ is a set and k a positive definite kernel, a feature map can be defined as

$$\begin{aligned} \boldsymbol{\phi} : \quad & \mathcal{X} \to \mathcal{H}_0, \\ & \mathbf{x} \mapsto k(\mathbf{x}, \cdot). \end{aligned} \tag{2.26}$$

That is, for any $\mathbf{x} \in \mathcal{X}$, $k(\mathbf{x}, \cdot) \in \mathcal{H}_0$ denotes the function that maps $\mathbf{x}' \in \mathcal{X}$ to $k(\mathbf{x}, \mathbf{x}') \in \mathbb{R}$. In other words, $k(\mathbf{x}, \cdot)$ is a real valued function on $\mathcal{X}$ whose value at an argument $\mathbf{x}'$ is a real number $k(\mathbf{x}, \mathbf{x}')$ which quantifies how similar $\mathbf{x}$ and $\mathbf{x}'$ are.

Let $\mathcal{H}$ be the set of all functions that are finite linear combinations of all possible

$k(\mathbf{x}, \cdot) \in \mathcal{H}_0$. Then, any function $f(\cdot) \in \mathcal{H}$ can be written as

$$f(\cdot) = \sum_{i=1}^{N} \alpha_i k(\mathbf{x}_i, \cdot), \tag{2.27}$$

for some $N \in \mathbb{N}, \mathbf{x}_i \in \mathcal{X}$ and $\alpha_i \in \mathbb{R}$. Given two functions $f, g \in \mathcal{H}$, it is easy to see that $f + g \in \mathcal{H}$ and $\alpha f \in \mathcal{H}$ for $\alpha \in \mathbb{R}$ would also be some finite linear combinations of all the $k(\mathbf{x}, \cdot)$ functions. Since function addition and scalar multiplication can be performed, $\mathcal{H}$ is considered to be a vector space.

Suppose $g(\cdot) = \sum_{j=1}^{N'} \beta_j k(\mathbf{x}_i^{'}, \cdot)$, then the inner product $\langle f, g \rangle_{\mathcal{H}}$ is defined by

$$\langle f, g \rangle_{\mathcal{H}} := \sum_{i=1}^{N} \sum_{j=1}^{N'} \alpha_i \beta_j k(\mathbf{x}_i, \mathbf{x}_j^{'}). \tag{2.28}$$

Notice that (2.28) can be written independent of the representation of f as

$$\langle f, g \rangle_{\mathcal{H}} = \sum_{j=1}^{N'} \beta_j \sum_{i=1}^{N} \alpha_i k(\mathbf{x}_i, \mathbf{x}_j^{'}) = \sum_{j=1}^{N'} \beta_j f(\mathbf{x}_j^{'}), \tag{2.29}$$

or independent of the representation of g as

$$\langle f, g \rangle_{\mathcal{H}} = \sum_{i=1}^{N} \alpha_i \sum_{j=1}^{N'} \beta_j k(\mathbf{x}_i, \mathbf{x}_j^{'}) = \sum_{i=1}^{N} \alpha_i g(\mathbf{x}_i). \tag{2.30}$$

This implies that $\langle f, g \rangle_{\mathcal{H}}$ does not depend on the specific expansion coefficients α and β. Hofmann et al. (2008) has proved that: operator $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ is indeed a valid inner product; is a positive definite kernel; the evaluation of a function at a point is given by the inner product between the function and the kernel centred at the point, that is,

$$f(\mathbf{x}) = \langle f, k(\mathbf{x}, \cdot) \rangle. \tag{2.31}$$

This is called the reproducing kernel property (Aronszajn, 1950). Thus, kernels that admit an inner product representation in a feature space are called reproducing

kernels. In particular, from (2.26), we have that $\boldsymbol{\phi}(\mathbf{x}) = k(\mathbf{x}, \cdot)$. Therefore,

$$\langle k(\mathbf{x}, \cdot), k(\mathbf{x}^{'}, \cdot)\rangle = k(\mathbf{x}, \mathbf{x}^{'}) = \langle \boldsymbol{\phi}(\mathbf{x}), \boldsymbol{\phi}(\mathbf{x}^{'})\rangle. \tag{2.32}$$

Hence, positive definite kernels are also called reproducing kernels (Schölkopf and Smola, 2002; Berg et al., 1984; Saitoh, 1988). The above description also shows that a positive definite kernel corresponds to an inner product in some feature space.

2.4.4 Reproducing kernel Hilbert spaces

A Hilbert space that is endowed with a reproducing kernel is called a reproducing kernel Hilbert space (RKHS) $\mathcal{H}_k$ or a proper Hilbert space. This subject (RKHS) was originally and simultaneously developed by Aronszajn (1950) and Bergman (1950). Aside process monitoring, RKHSs also arise in a number of other areas such as complex analysis, harmonic analysis, quantum mechanics and machine learning. A formal definition of RKHS is given below.

Definition 2.5 (Reproducing kernel Hilbert space). Assume that $\mathcal{X}$ is a non-empty set and $\mathcal{H}$ is a Hilbert space of functions $f : \mathcal{X} \to \mathbb{R}$. Then, $\mathcal{H}$ is a RKHS endowed with an inner product $\langle \cdot, \cdot \rangle$ and a norm $\|f\| := \langle f, f\rangle^{\frac{1}{2}}$, if there exists a function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ for which the following properties are satisfied:

(i) k has a reproducing property

$$\langle f, k(\mathbf{x}, \cdot) = f(\mathbf{x})\rangle, \quad \forall f \in \mathcal{H},\ \forall \mathbf{x} \in \mathcal{X} \tag{2.33}$$

and in particular, $\langle k(\mathbf{x}, \cdot), k(\mathbf{x}^{'}, \cdot)\rangle = k(\mathbf{x}, \mathbf{x}^{'})$.

(ii) $\mathcal{H}$ is spanned by k, that is, every $f \in \mathcal{H}$ can be written in the form of (2.27), $f(\cdot) = \sum_{i=1}^{N} \alpha_i k(\mathbf{x}_i, \cdot)$.

The RKHS representation of positive definite kernels presented here is based on the Moore-Aronszajn theorem (Aronszajn, 1950) which says that every symmetric

positive definite kernel defines a unique RKHS. Mercers's theorem, an alternative way of characterising symmetric p.d. kernels, is discussed in the next sub section.

2.4.5 Mercer's Theorem

Mercer's Theorem provides a solid way for constructing a RKHS. In general terms the theorem states that a positive kernel k, can be expanded in terms of eigenfunctions and eigenvalues of a positive operator that comes from the kernel.

Mercer's Theorem. Given a real-valued symmetric function k and a compact input space X, there exists an inner product space $\mathcal{H}$ and a mapping $\boldsymbol{\phi} : X \to \mathcal{H}$ so that $k(\mathbf{x}, \mathbf{x}^{'}) = \left\langle \boldsymbol{\phi}(\mathbf{x}), \boldsymbol{\phi}(\mathbf{x}') \right\rangle$ if for the set of all square-integrable functions $f(\mathbf{x})$ (that is, $\int f(\mathbf{x})^2 \mathrm{d}\mathbf{x} < \infty$),

$$\int_X k(\mathbf{x}, \mathbf{x}^{'}) f(\mathbf{x}) f(\mathbf{x}^{'}) \mathrm{d}\mathbf{x}\ \mathrm{d}\mathbf{x}^{'} \geq 0. \tag{2.34}$$

Expanding the function $k(\mathbf{x}, \mathbf{x}^{'})$ in its eigenfunctions gives

$$k(\mathbf{x}, \mathbf{x}^{'}) = \sum_{i=1}^{n_{\mathcal{H}}} \lambda_i \psi_i(\mathbf{x}) \psi_i(\mathbf{x}^{'}), \tag{2.35}$$

for $\lambda_i \geq 0$ and $\psi_i : X \to \mathbb{R}$, where $\int_X k(\mathbf{x}, \mathbf{x}^{'}) \psi_i(\mathbf{x}^{'}) \mathrm{d}\mathbf{x}^{'} = \lambda_i \psi_i(\mathbf{x})$; where ψ_i and λ_i are the eigenfunctions and (nonnegative) eigenvalues of k respectively (Mercer, 1909). Equation (2.34) is referred to as Mercer's condition and a kernel satisfying this condition is called a Mercer kernel.

Given, a data point $\mathbf{x}$, a feature map $\boldsymbol{\phi}$ can be defined as

$$\boldsymbol{\phi}(\mathbf{x}) = \left[\sqrt{\lambda_1} \psi_1(\mathbf{x}), \sqrt{\lambda_2} \psi_2(\mathbf{x}_2), \ldots, \sqrt{\lambda_{n_{\mathcal{H}}}} \psi_{n_{\mathcal{H}}}(\mathbf{x}_{n_{\mathcal{H}}}) \right]^T. \tag{2.36}$$

The number of nonnegative eigenvalues $m_{\mathcal{H}}$ determines the dimensionality of this space which is infinite for the Gaussian kernel.

A RKHS of functions defined over X based on Mercer's theorem can then be con-

structed as linear combinations of the eigenfunctions

$$f(\mathbf{x}) = \sum_{i=1}^{\infty} \alpha_i k(\mathbf{x}_i, \mathbf{x}) = \sum_{i=1}^{\infty} \alpha_i \sum_{j=1}^{n_{\mathcal{H}}} \lambda_j \psi_j(\mathbf{x}_j)\psi_j(\mathbf{x}). \tag{2.37}$$

Defining an inner product $\langle \cdot, \cdot \rangle$:

$$\langle f, k(\mathbf{x}, \cdot) \rangle = \sum_{i=1}^{\infty} \alpha_i \sum_{j=1}^{n_{\mathcal{H}}} \sum_{n=1}^{m_{\mathcal{H}}} \lambda_j \psi_j(\mathbf{x}_i) \big\langle \psi_j, \psi_n \big\rangle \lambda_n \psi_n(\mathbf{x}), \tag{2.38}$$

and taking $\big\langle \psi_j, \psi_n \big\rangle = \delta_{jn}/\lambda_j$ makes the Mercer kernel a reproducing kernel for this Hilbert space, that is, $\langle f, k(\mathbf{x}, \cdot)$.

The kernel maps based on Mercer's theorem and Moor-Aronszajn theorem lead to different feature spaces. Nevertheless, since both are Hilbert spaces, it implies that they are isometrically isomorphic.

2.4.6 Feature maps associated with kernels

Apart from constructing a feature space, another interesting point was made in the preceding subsections. It is the notion of finding a feature mapping $\boldsymbol{\phi} : \mathcal{X} \rightarrow \mathcal{F}$ such that:

$$k(\mathbf{x}, \mathbf{x}^{'}) = \langle \boldsymbol{\phi}(\mathbf{x}), \boldsymbol{\phi}(\mathbf{x}^{'}) \rangle_{\mathcal{F}}, \tag{2.39}$$

where $\mathcal{F}$ is some high dimensional feature space and $\boldsymbol{\phi}$ is the feature map associated with the kernel k. It was actually this (feature map) viewpoint that stimulated the initial advocacy for kernel-based learning algorithms (Schölkopf and Smola (2002)). This viewpoint implies that evaluation of a symmetric p.d. kernel k at points $\mathbf{x}$ and $\mathbf{x}^{'}$ in the input space is equivalent to taking the inner products between $\boldsymbol{\phi}(\mathbf{x})$ and $\boldsymbol{\phi}(\mathbf{x}^{'})$ in some (probably unknown) high dimensional feature space. Thus, $\boldsymbol{\phi}$ can be thought of as a mapping from the input space to a high dimensional feature space where inner products can be computed simply by computing k. In other words, there is no need to use (or even know) $\boldsymbol{\phi}$ to embed the data into the feature space and explicit computation of $\boldsymbol{\phi}(\mathbf{x})$ and $\boldsymbol{\phi}(\mathbf{x}^{'})$ is not required. The advantage of this

is that the complexity of the optimisation problem depends on the dimensionality of the input space rather than that of the feature space. Thus, kernel methods make possible the use of feature spaces with exponential or even infinite dimensionality. Without this so called kernel trick, it would not have been possible for kernel-based methods to satisfy the desired efficiency requirements. In this subsection, two common constructions of $\boldsymbol{\phi}$ (based on RKHS and Mercer's theorem) which are more or less equivalent, are presented.

In Section 2.4.3, the feature map was defined by $\boldsymbol{\phi}(\mathbf{x}) = k(\mathbf{x}, \cdot)$ and the feature space by $\mathcal{H}$. Using the reproducing property of the kernel, this gives:

$$\langle \boldsymbol{\phi}(\mathbf{x}), \boldsymbol{\phi}(\mathbf{x}^{'}) \rangle_{\mathcal{H}_k} = \langle k(\mathbf{x}, \cdot), k(\mathbf{x}^{'}, \cdot) \rangle_{\mathcal{H}_k} = k(\mathbf{x}, \mathbf{x}^{'}),$$

which satisfies the requirement for $\boldsymbol{\phi}$ (see Schölkopf et al., 1999).

In Mercers's representation of p.d. kernels, eigenfunctions $\boldsymbol{\psi}_i$ and eigenvalues λ_i were used to define $\boldsymbol{\phi}$ in (2.36). The inner product can therefore be calculated as

$$\begin{aligned} \left\langle \boldsymbol{\phi}(\mathbf{x}), \boldsymbol{\phi}(\mathbf{x}^{'}) \right\rangle &= \left\langle \sqrt{\lambda_i}\psi_i(\mathbf{x}), \sqrt{\lambda_i}\psi_i(\mathbf{x}^{'}) \right\rangle \\ &= \sum_{i=1}^{m_{\mathcal{H}}} \sqrt{\lambda_i}\psi_i(\mathbf{x}) \sqrt{\lambda_i}\psi_i(\mathbf{x}^{'}) \\ &= \sum_{i=1}^{m_{\mathcal{H}}} \lambda_i \psi_i(\mathbf{x}) \lambda_i \psi_i(\mathbf{x}^{'}) \\ &= k(\mathbf{x}, \mathbf{x}^{'}). \end{aligned}$$

An important outcome of feature mapping is that it causes nonlinear relationships in the input space to appear linear in the high dimensional feature space. Consider the toy example of a mapping from two to three dimensions shown below:

$$\begin{aligned} \boldsymbol{\phi} : \mathbb{R}^2 &\rightarrow \mathbb{R}^3, \\ (x_1, x_2) &\mapsto (x_1^2, \sqrt{2}x_1x_2, x_2^2). \end{aligned} \tag{2.40}$$

The inner products between the vectors in the feature space can be computed as

$$\begin{aligned} \langle \boldsymbol{\phi}(\mathbf{x}), \boldsymbol{\phi}(\mathbf{y}) \rangle &= \left\langle (x_1^2, \sqrt{2}x_1x_2, x_2^2), (y_1^2, \sqrt{2}y_1y_2, y_2^2) \right\rangle \\ &= \langle (x_1, x_2), (y_1, y_2) \rangle^2 \\ &= \langle \mathbf{x}, \mathbf{y} \rangle^2 \\ &= k(\mathbf{x}, \mathbf{y}). \end{aligned} \tag{2.41}$$

That is, the function

$$\langle \mathbf{x}, \mathbf{y} \rangle^2 = k(\mathbf{x}, \mathbf{y}),$$

is a kernel function (a polynomial kernel).

Figure 2.3 shows the effect of the above mapping. Data points that are linearly non-separable in the original 2-dimensional (2-D) space are linearly separable in the 3-dimensional (3-D) feature space. Linear relations in the 3-D space corresponds to quadratic relations in the original 2-D space. That is, the feature space is nonlinearly related to the input space. Consequently, linear techniques can be applied to

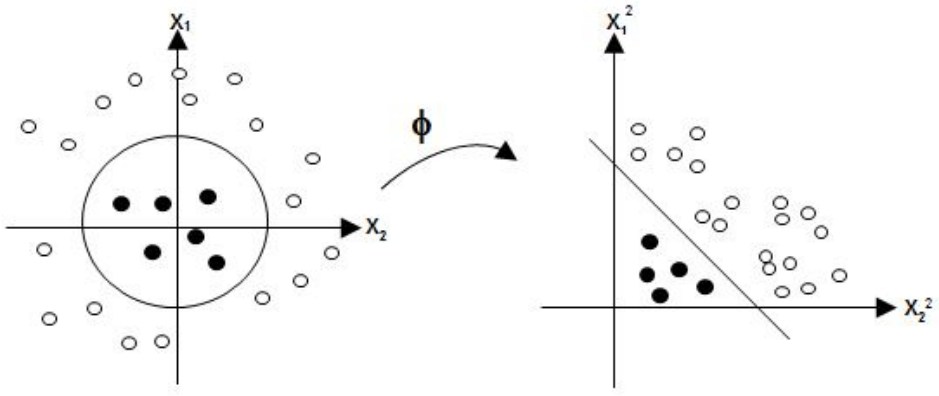


Figure 2.3: An illustration of the effect of mapping data into a high dimensional feature space. The nonlinear function $\boldsymbol{\phi}$ embeds data in the feature space. Data points which were not linearly separable in the input space (left panel) become linearly separable after mapping into higher dimensions (right panel).

detect the relationships in the higher dimensional space. This is desirable because techniques capable of detecting linear relations efficiently are well-researched and well-understood. The same tools can be applied to detect nonlinear relations after mapping the data into a higher dimensional space.

Another interesting point to note in kernel-based feature mapping is that the function $\boldsymbol{\phi}$ (or the dimension of its range) is not unique. The finding in (2.41) can be generalised to mean that (Muller et al., 2001; Schölkopf et al., 1998): the same kernel can compute the inner product that corresponds to d dimensional feature map such that for $\mathbf{x}, \mathbf{y} \in \mathbb{R}^M$ and $d \in \mathbb{N}$, the kernel function can be represented as

$$k(\mathbf{x}, \mathbf{y})^d.$$

Hence, it is possible for more than one $\boldsymbol{\phi}$ to exist for a given kernel function (Shawe-Taylor and Cristianini, 2004).

2.4.7 Examples and properties of kernels

Examples of some commonly used kernels are presented in Table 2.1.

Table 2.1: Examples of commonly used kernels

Type of kernel	**Inner product kernel**	**Comments**
Linear kernel	$k(\mathbf{x}, \mathbf{x}') = \langle \mathbf{x}, \mathbf{x}' \rangle$	Here, $\boldsymbol{\phi}$ is an identity, therefore, no mapping takes place.
Polynomial kernel	$k(\mathbf{x}, \mathbf{x}') = \langle \mathbf{x}, \mathbf{x}' \rangle^d$	$d \in \mathbb{N}$ is specified a priori by user.
Radial basis function (RBF) kernel	$k(\mathbf{x}, \mathbf{x}') = \exp\Big(- \frac{\|\mathbf{x} - \mathbf{x}'\|^2}{c} \Big)$	$c \in \mathbb{R}$ is specified by user.
Sigmoid kernel	$k(\mathbf{x}, \mathbf{x}') = \tanh\left(\kappa \langle \mathbf{x}, \mathbf{x}' \rangle + \theta\right)$	$\kappa, \theta > 0$. Mercer's theorem is only satisfied for some values of κ and θ.

Since kernel functions measure similarity between data objects, intuitively, different similarity measures can also be combined to create new kernels. Two kernel techniques, KPCA and kernel CCA (KCCA), are discussed in the next two subsections.

2.4.8 Kernel principal component analysis

KPCA was proposed by Schölkopf et al. (1998) as a nonlinear generalisation of the PCA. A number of studies adopting the technique for nonlinear process monitoring have also been reported (Ge and Song, 2013; Choi et al., 2005; Cho et al., 2005). KPCA is performed in two steps: (i) mapping the input data into a high-dimensional feature space, and (ii) performing standard PCA in the feature space.

Other PCA-based nonlinear techniques have also been proposed to address the nonlinearity problem. Kramer (1992) proposed a nonlinear PCA based on auto-associative neural network. Dong and McAvoy (1996) suggested a nonlinear PCA that combined principal curves and neural network (NN). Their approach involved: (i) using principal curve method to obtain associated scores and the correlated data, (ii) using an NN model to map the original data into scores, and (iii) mapping the scores into the original variables.

Nonlinear PCA methods have also been proposed by (Jia et al., 2000; Hiden et al., 1999; Cheng and Chiu, 2005; Kruger et al., 2005). However, most of these methods are based on NNs and require the solution of a nonlinear and non-convex optimisation problem. Conversely, KPCA does not involve solving a nonlinear optimisation problem; it only solves an eigenvalue problem. Hence, implementing the KPCA is computational less expensive compared to NN-based methods. Furthermore, KPCA does not require specifying the number of PCs in advance (Choi et al., 2005).

Similar to the PCA, the Hotelling's T^2 and Q statistics are commonly used in KPCA-based process monitoring. In a linear method such as PCA, computation of the upper control limits (UCLs) of T^2 and Q statistics are based on the assumption that random variables included in the data are Gaussian. The actual distribution of T^2 and Q statistics can be analytically derived based on this assumption. Hence, the UCLs can also be derived analytically. However, many real industrial processes are nonlinear. Even though the sources of randomness of these processes could be assumed as Gaussian, variables included in measured data are non-Gaussian due to inherent nonlinearities. Hence, adopting UCLs for fault detection based on the

multivariate Gaussian assumption in such processes is inappropriate and may lead to misleading results (Ge and Song, 2013).

An alternative solution to the non-Gaussian problem, is to derive the UCLs from the underlying probability density functions (PDFs) estimated directly from the T^2 and the Q statistics via a non-parametric technique such as kernel density estimation (KDE). This approach has been applied in various linear fault detection techniques, such as PCA (Chen et al., 2000; Liang, 2005), independent component analysis (ICA) (Xiong et al., 2007), and canonical variate analysis (CVA) (Odiowei and Cao, 2010). It is even more important to adopt this kind of approach to derive meaningful UCLs for a nonlinear technique such as the KPCA. This is because the Gaussian-assumption-based UCLs for latent variables obtained through a nonlinear technique will not be valid at all. Unfortunately, this issue has not attracted enough attention in the literature. Therefore, the KPCA with KDE (KPCA-KDE) technique is investigated in this work.

2.4.9 Kernel canonical correlation analysis

Since CCA is a linear technique, even when strong correlation exists between a pair of multivariate data sets, such a correlation may not be captured by CCA, if the association between the data sets is non-linear. Hence, kernel CCA was developed as a nonlinear generalisation of the CCA to capture nonlinear relations.

Given a pair of input vectors $\mathbf{x}$ and $\mathbf{y}$ drawn from two sets of data, non-linear mappings into high dimensional feature spaces $\boldsymbol{\mathcal{F}}_x$ and $\boldsymbol{\mathcal{F}}_y$ are defined by $\boldsymbol{\phi}_x : \mathbf{x} \mapsto \boldsymbol{\phi}_x(\mathbf{x})$ and $\boldsymbol{\phi}_y : \mathbf{y} \mapsto \boldsymbol{\phi}_y(\mathbf{y})$ respectively Hofmann et al. (2008). Applying CCA to $\boldsymbol{\phi}_x(\mathbf{x}), \boldsymbol{\phi}_y(\mathbf{y})$ in $\boldsymbol{\mathcal{F}}_x$ and $\boldsymbol{\mathcal{F}}_y$ provides linear correlation in $\boldsymbol{\mathcal{F}}_x$ and $\boldsymbol{\mathcal{F}}_y$ which indicates the non-linear correlation in the input spaces. Assuming the data are mean centred,

the covariance and cross-covariance matrices in the feature space are computed as

$$\boldsymbol{\Sigma}_{xx} = E\left\{\boldsymbol{\phi}_x(\mathbf{x}), \boldsymbol{\phi}_x(\mathbf{x})^T\right\} \tag{2.42}$$

$$\boldsymbol{\Sigma}_{yy} = E\left\{\boldsymbol{\phi}_y(\mathbf{y}), \boldsymbol{\phi}_y(\mathbf{y})^T\right\} \tag{2.43}$$

$$\boldsymbol{\Sigma}_{xy} = E\left\{\boldsymbol{\phi}_x(\mathbf{x}), \boldsymbol{\phi}_y(\mathbf{y})^T\right\} \tag{2.44}$$

Kernel CCA seeks to find weights $\mathbf{w}_x$ and $\mathbf{w}_y$ which maximise $\mathbf{w}_x^T\boldsymbol{\Sigma}_{xy}\mathbf{w}_y$ subject to the constraint that $\mathbf{w}_x^T\boldsymbol{\Sigma}_{xx}\mathbf{w}_x = \mathbf{w}_y^T\boldsymbol{\Sigma}_{yy}\mathbf{w}_y = 1$. Expressing the weight vectors as linear combinations of the mapped data in the feature space gives:

$$\mathbf{w}_x = \sum_{i=1}^{N} \alpha_i \boldsymbol{\phi}_x(\mathbf{x}_i) \tag{2.45}$$

$$\mathbf{w}_y = \sum_{i=1}^{N} \beta_i \boldsymbol{\phi}_y(\mathbf{y}_i) \tag{2.46}$$

Defining kernel matrices

$$\mathbf{K}_x = \langle \boldsymbol{\phi}_x(\mathbf{x}_i), \boldsymbol{\phi}_x(\mathbf{x}_j) \rangle \text{ and } \mathbf{K}_y = \left\langle \boldsymbol{\phi}_y(\mathbf{y}_i), \boldsymbol{\phi}_y(\mathbf{y}_j) \right\rangle \tag{2.47}$$

for all $i, j = 1, \ldots, N$ where N is the number of observations, the optimisation problem for kernel CCA is set as (Chu et al., 2013)

$$\max_{\alpha,\beta} \quad \boldsymbol{\alpha}^T\mathbf{K}_x\mathbf{K}_y\boldsymbol{\beta} \tag{2.48}$$

$$\text{s.t.} \quad \boldsymbol{\alpha}^T\mathbf{K}_x\mathbf{K}_x\boldsymbol{\alpha} = \boldsymbol{\beta}^T\mathbf{K}_y\mathbf{K}_y\boldsymbol{\beta} = 1 \tag{2.49}$$

Defining $\boldsymbol{\Gamma}_{xx} = \mathbf{K}_x\mathbf{K}_x^T$, $\boldsymbol{\Gamma}_{yy} = \mathbf{K}_y\mathbf{K}_y^T$, and $\boldsymbol{\Gamma}_{xy} = \mathbf{K}_x\mathbf{K}_y^T$, SVD can be performed on the product matrix $\mathbf{L}_2$ similar to (2.15):

$$\mathbf{L}_2 = \boldsymbol{\Gamma}_{xx}^{-1/2}\boldsymbol{\Gamma}_{xy}\boldsymbol{\Gamma}_{yy}^{-1/2} \tag{2.50}$$

The problem is that the effective rank of a kernel matrix is often lower than its size (Huang et al., 2009). Hence $\mathbf{K}_x$ and $\mathbf{K}_y$ could be singular. Therefore, regu-

larisation is required so that matrix inversion required in CCA can be performed. Thus, $\mathbf{K}_x\mathbf{K}_x^T$ and $\mathbf{K}_y\mathbf{K}_y^T$ become $(\mathbf{K}_x + \gamma\boldsymbol{I})(\mathbf{K}_x + \gamma\boldsymbol{I})^T$ and $(\mathbf{K}_y + \gamma\boldsymbol{I})(\mathbf{K}_y + \gamma\boldsymbol{I})^T$ respectively (Giantomassi et al., 2014), where γ is the regularisation parameter and $\boldsymbol{I}$ is an identity matrix. Denoting $\boldsymbol{\Gamma}_{xx}^* = (\mathbf{K}_x + \gamma\boldsymbol{I})(\mathbf{K}_x + \gamma\boldsymbol{I})^T$ and $\boldsymbol{\Gamma}_{yy}^* = (\mathbf{K}_y + \gamma\boldsymbol{I})(\mathbf{K}_y + \gamma\boldsymbol{I})^T$, a recast of (2.50) is given as

$$\mathbf{L}_2^* = \boldsymbol{\Gamma}_{xx}^{*-1/2}\boldsymbol{\Gamma}_{xy}\boldsymbol{\Gamma}_{yy}^{*-1/2} = \mathbf{U}_2\mathbf{S}_2\mathbf{V}_2^T \tag{2.51}$$

However, the optimum value of γ may not be easily specified. Therefore, carrying out KPCA on the original samples before performing CCA to reduce dimensionality is an alternative approach to attack the singularity problem.

2.4.10 Nonlinear dynamic process monitoring

Some techniques that address nonlinearity and process dynamics separately have been identified in this chapter. In particular, the KPCA addresses nonlinearity but cannot capture dynamic relationships while DPCA, DPLS, CVA, etc. capture process dynamics but cannot deal with nonlinearity. However, many real industrial systems exhibit both nonlinear and dynamic behaviour at the same time. Unfortunately, monitoring techniques aimed at addressing both of these process characteristics simultaneously have received only limited attention in the literature.

One of the few studies that have addressed these two process behaviours simultaneously was carried out by Choi and Lee (2004). They developed the dynamic kernel PCA (DKPCA) and applied it to a simulated nonlinear process and a wastewater treatment facility. This technique captures nonlinearity using the kernel paradigm while process dynamics was addressed using a time lagged version of the original observation data. According to their study, the approach outperformed other techniques such as PCA and KPCA by giving fewer missing alarms and shorter times to detect faults.

As noted in the introduction, kernel methods are efficient techniques for capturing

nonlinear relations. However, in the DKPCA technique proposed by Choi and Lee (2004), the kernel approach is combined with the DPCA which has limitation in capturing process dynamics (Odiowei and Cao, 2010). This is likely to limit the performance of this technique. In other words, although the approach has a good technique for describing nonlinear relations, the method employed to account for process dynamics leaves room for improvement. Furthermore, Odiowei and Cao (2010) developed the CVA with KDE technique to enhance the monitoring performance of nonlinear dynamic processes. To account for nonlinearity, they associated the CVA with KDE-derived control limits. They reported that this approach performed better than the non KDE-based techniques. However, despite using the CVA which is known to perform well in dynamic process monitoring, the approach does not directly address the nonlinear problem.

The preceding discussion suggests that the kernel approach and CVA can be used to effectively address nonlinearity and process dynamics respectively. It is therefore possible to gain from the synergy obtainable from integrating these two approaches to enhance nonlinear dynamic process monitoring. Nevertheless, such studies have not been given the deserved attention in the open literature.

The development of a smart system for monitoring households to minimise energy wastage (Giantomassi et al., 2014) is a good attempt based on the above integration concept. This study involved using kernel canonical variate analysis (KCVA) to detect system malfunctions and occupant bad behaviour (e.g. opening of a window while the heating system is on during the winter). The technique was implemented by performing the CVA on kernel matrices constructed from collected experimental data. They reported that the technique allowed the detection and diagnosis of several faults by simply monitoring the trend of ambient temperature of each room of the apartment. However, kernel matrices are often singular and have effective ranks lower than their sizes (Huang et al., 2009). This also apply to the sample covariance matrices based on the kernel data. Hence, the optimisation problem solved in canonical analysis becomes ill-conditioned. Therefore, regularisation of kernel matrices constructed from observation data is needed. Regularisation, how-

ever, reduces model accuracy and impacts negatively on monitoring performance especially if the optimum regularisation value is not used. This makes the performance of the approach used by Giantomassi et al. (2014) overly dependent on the value of the regularisation parameter selected. Furthermore, the study does not include a comparative analysis of methods. Therefore, it is difficult to compare the performance of the technique with other approaches. The development of two new kernel-based methods with novel implementation strategies for nonlinear dynamic process monitoring including comparative analyses are therefore undertaken in this work.

2.4.11 Concluding remarks

This chapter provided an overview of process monitoring methods. Some equivalent conditions of a kernel on a set $\mathcal{X}$ were also defined. These include:

- Every kernel matrix (or Gram matrix) is positive definite.
- k is the reproducing kernel of a RKHS of functions on $\mathcal{X}$.
- A kernel can be expressed as $k(\mathbf{x}, \mathbf{x}^{'}) = \sum_{i=1}^{n_{\mathcal{H}}} \lambda_i \psi_i(\mathbf{x}) \psi_i(\mathbf{x}^{'})$.
- A kernel has also been considered as a feature map

It was shown in this chapter that a positive definite kernel k on a set $\mathcal{X}$ corresponds to taking the inner products between pairs of data points mapped into the feature space. Furthermore, the dimension of the feature space may be very large or even infinite. Therefore, the mapping might neither be explicitly given nor convenient to work with in practice. Hence, working in the feature space is done implicitly using the kernel trick (that is, formulating algorithms to process finite dimensional vectors expressed in terms of pairwise inner products). That is, the algorithms are applied to potentially infinite-dimensional vectors in the feature space by replacing each inner product evaluation with a kernel evaluation.

The kernel approach is modular and using the kernel trick extends many linear algorithms to nonlinear settings including general data types (for example, non-vectorial data). Modularity means that a given algorithm can be used with any kernel and therefore any data type. There is no need to change an algorithm in order to accommodate a specified kernel function. It is also possible to carry out a different task with the same kernel by simply using another algorithm. Thus, design and analysis of algorithms can be considered separately from the choice of kernel functions. Hence, different modules can be combined together to give complex algorithms.

Finally, many real industrial processes exhibit both nonlinear and dynamic characteristics. However, kernel-based methods for monitoring such processes are limited. Therefore, the development of two new kernel-based algorithms with novel implementation strategies to take into account nonlinearity and process dynamics simultaneously is considered to be a key contribution of this work.

Chapter 3

Nonlinear process fault detection and identification using kernel PCA and KDE

KPCA is an effective and efficient method for monitoring nonlinear processes. However, associating it with upper control limits (UCLs) based on the Gaussian distribution can deteriorate its performance. In this chapter, the mathematical formulation of KPCA is undertaken while the kernel density estimation (KDE) technique is used to estimate UCLs for KPCA-based nonlinear process monitoring. The tehnique is applied to the Tennessee Eastman process and the monitoring performance of the KPCA-KDE approach is compared with KPCA whose UCLs are based on the Gaussian distribution. The robustness of the KPCA-KDE technique is demonstrated and a new fault identification approach is proposed.

3.1 KPCA-KDE-based process monitoring

3.1.1 Kernel PCA algorithm

Given m training samples $\mathbf{x}_k \in \Re^n, k = 1, \ldots, m$, the data can be projected onto a high-dimensional feature space using a nonlinear mapping, $\boldsymbol{\phi} : \mathbf{x}_k \in \Re^n \longrightarrow \boldsymbol{\phi}(\mathbf{x}_k) \in \Re^F$. The covariance matrix in the feature space is then computed as

$$\mathbf{C}_F = \frac{1}{m} \sum_{j=1}^{m} \langle \boldsymbol{\phi}(\mathbf{x}_j), \boldsymbol{\phi}(\mathbf{x}_j) \rangle, \tag{3.1}$$

where $\boldsymbol{\phi}(\mathbf{x}_j)$, for $j = 1, \ldots m$ is assumed to have zero mean and unit variance. To diagonalize the covariance matrix, eigenvalue decomposition is performed in the feature space using

$$\lambda \mathbf{a} = \mathbf{C}_F \mathbf{a}, \tag{3.2}$$

where λ is an eigenvalue of $\mathbf{C}_F$, satisfying $\lambda \geq 0$, and $\mathbf{a} \in \Re^F$ is the corresponding eigenvector ($\mathbf{a} \neq 0$).

The eigenvector can be expressed as a linear combination of the mapped data points as follows:

$$\mathbf{a} = \sum_{i=1}^{m} \alpha_i \boldsymbol{\phi}(\mathbf{x}_i). \tag{3.3}$$

Using $\boldsymbol{\phi}(\mathbf{x}_k)$ to multiply both sides of (3.2) gives

$$\lambda \langle \boldsymbol{\phi}(\mathbf{x}_k), \mathbf{a} \rangle = \langle \boldsymbol{\phi}(\mathbf{x}_k), \mathbf{C}_F \mathbf{a} \rangle. \tag{3.4}$$

Substituting (3.1) and (3.3) in (3.4) gives

$$\begin{aligned} &\lambda \sum_{i=1}^{m} \alpha_i \langle \boldsymbol{\phi}(\mathbf{x}_k), \boldsymbol{\phi}(\mathbf{x}_i) \rangle \\ &\qquad = \frac{1}{m} \sum_{i=1}^{m} \alpha_i \left\langle \boldsymbol{\phi}(\mathbf{x}_k), \sum_{j=1}^{m} \boldsymbol{\phi}(\mathbf{x}_j) \right\rangle \langle \boldsymbol{\phi}(\mathbf{x}_j), \boldsymbol{\phi}(\mathbf{x}_i) \rangle. \end{aligned} \tag{3.5}$$

Instead of performing eigenvalue decomposition directly on $\mathbf{C}_F$ in (3.1) and finding eigenvalues and PCs, we apply the kernel trick by defining an $m \times m$ (kernel) matrix as follows:

$$[\mathbf{K}]_{ij} = K_{ij} = \langle \boldsymbol{\phi}(\mathbf{x}_i), \boldsymbol{\phi}(\mathbf{x}_j) \rangle = k(\mathbf{x}_i, \mathbf{x}_j) \tag{3.6}$$

for all $i, j = 1, \ldots, m$. Introducing the kernel function of the form $k(\mathbf{x}, \mathbf{y}) = \langle \boldsymbol{\phi}(\mathbf{x}), \boldsymbol{\phi}(\mathbf{y}) \rangle$ in (3.5) enables the computation of the inner products $\langle \boldsymbol{\phi}(\mathbf{x}_i), \boldsymbol{\phi}(\mathbf{x}_j) \rangle$ in the feature space as a in terms of the original input data. This precludes the need to carry out the nonlinear mappings and the explicit computation of inner products in the feature space (Schölkopf et al., 1998; Lee et al., 2004). Applying the kernel matrix, (3.5) can be re-written as

$$\lambda \sum_{i=1}^{m} \alpha_i K_{ki} = \frac{1}{m} \sum_{i=1}^{m} \alpha_i \sum_{j=1}^{m} K_{kj} K_{ji}. \tag{3.7}$$

Notice that $k = 1, \ldots, m$, therefore, (3.7) can be represented as

$$\lambda m \mathbf{K} \boldsymbol{\alpha} = \mathbf{K}^2 \boldsymbol{\alpha}. \tag{3.8}$$

Equation (3.8) is equivalent to the eigenvalue problem

$$m\lambda \boldsymbol{\alpha} = \mathbf{K} \boldsymbol{\alpha}. \tag{3.9}$$

Furthermore, the kernel matrix can be mean centred before eigenvalue decomposition as follows

$$\mathbf{K}_{ctr} = \mathbf{K} - \mathbf{I}_n \mathbf{K} - \mathbf{K} \mathbf{I}_n + \mathbf{I}_n \mathbf{K} \mathbf{I}_n, \tag{3.10}$$

where $\mathbf{I}_n$ is an $m \times m$ matrix in which each element is equal to $1/m$. Eigen decomposition of $\mathbf{K}_{ctr}$ is equivalent to performing PCA in $\Re^F$. This, essentially, amounts to solving the eigenvalue decomposition problem in (3.9) to determine the eigenvectors $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \ldots, \boldsymbol{\alpha}_m$ and their corresponding eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_m$.

Since the kernel matrix, $\mathbf{K}_{ctr}$ is symmetric, the derived PCs are orthonormal, that

is,

$$\langle \boldsymbol{\alpha}_i, \boldsymbol{\alpha}_j \rangle = \delta_{i,j}, (i, j = 1, 2, \ldots, m), \tag{3.11}$$

where $\delta_{i,j}$ represents the Dirac delta function.

The score vectors of the nonlinear mapping of mean-centred training observations $\mathbf{x}_j$, $j = 1, \ldots, m$, can then be extracted by projecting $\boldsymbol{\phi}(\mathbf{x}_j)$ onto the PC space spanned by the eigenvectors $\boldsymbol{\alpha}_k$, $k = 1, \ldots, m$,

$$z_{k,j} = \langle \boldsymbol{\alpha}_k, \mathbf{k}_{ctr} \rangle = \sum_{i=1}^{m} \alpha_{k,i} \langle \boldsymbol{\phi}(\mathbf{x}_i), \boldsymbol{\phi}(\mathbf{x}_j) \rangle . \tag{3.12}$$

Applying the kernel trick, this can be expressed as

$$z_{k,j} = \sum_{i=1}^{m} \alpha_{k,i} \left[\mathbf{K}_{ctr}\right]_{i,j} . \tag{3.13}$$

3.1.2 Fault detection metrics

The Hotelling's T^2 of the j^{th} samples in the feature space used for KPCA fault detection is computed as

$$T_j^2 = [z_{1,j}, \ldots, z_{q,j}] \, \Omega^{-1} \, [z_{1,j}, \ldots, z_{q,j}]^T \tag{3.14}$$

where $z_{i,j}$, $i = 1, \ldots, q$ represents the PC scores of the j^{th} samples, q is the number of nonlinear PCs retained and Ω^{-1} represents the inverse of the matrix of eigenvalues corresponding to the retained PCs. A simplified computation of the Q-statistic has been proposed by Lee et al. (2004). For the the j^{th} samples,

$$Q_j = ||\phi(\mathbf{x}_j) - \hat{\phi}_q(\mathbf{x}_j)||^2 = \sum_{i=1}^{m} z_{i,j}^2 - \sum_{i=1}^{q} z_{i,j}^2 \tag{3.15}$$

The method of computing control limits directly from the PDFs of the T^2 and Q statistics is explained in the next section.

3.1.3 Kernel density estimation

KDE is a procedure for fitting a data set with a suitable smooth PDF from a set of random samples. It is used widely for estimating PDFs, especially for univariate random data (Bowman and Azzalini, 1997). The KDE is applicable for the T^2 and Q statistics since both are univariate although the process characterized by these statistics is multivariate.

Given a random variable y, its PDF $g(y)$ can be estimated from its N samples, y_j, $j = 1, \ldots, N$, as follows:

$$g(y) = \frac{1}{Nh} \sum_{j=1}^{N} K(\frac{y - y_j}{h}) \tag{3.16}$$

where K is a kernel function while h is the window width (also known as bandwidth or smoothing parameter). Integrating the density function over a continuous range gives the probability. Thus, the probability of y to be less than the control limit c based on a given significance level, α is given by:

$$P(y < c) = \int_0^c g(y)\,\mathrm{d}y = \alpha. \tag{3.17}$$

where $g(y)$ is the PDF. Consequently, the control limits of the monitoring measures (i.e., T^2 and Q) can be calculated from their respective PDF estimates as

$$\int_0^{T_\alpha^2} g(T^2)\,\mathrm{d}T^2 = \alpha \tag{3.18}$$

$$\int_0^{Q_\alpha} g(Q)\,\mathrm{d}Q = \alpha. \tag{3.19}$$

3.1.4 On-line monitoring

For a mean-centred test observation, $\mathbf{x}_{tt}$, the corresponding kernel vector, $\mathbf{k}_{tt}$ is calculated with the training samples, $\mathbf{x}_j$, $j = 1, \ldots, m$ as follows:

$$[\mathbf{k}_{tt}]_j = k(\mathbf{x}_j, \mathbf{x}_{tt}). \tag{3.20}$$

The test kernel vector is then centred as shown below:

$$\mathbf{k}_{ctt} = \mathbf{k}_{tt} - \mathbf{K}\mathbf{I}_t - \mathbf{I}_n\mathbf{k}_{tt} + \mathbf{I}_n\mathbf{K}\mathbf{I}_t. \tag{3.21}$$

where $\mathbf{I}_t = \frac{1}{m}[1, \ldots, 1]^T \in \Re^m$. The corresponding test score vector, $\mathbf{z}_{tt}$ is calculated using

$$z_{tt,k} = \langle \boldsymbol{\alpha}_k, \mathbf{k}_{ctt} \rangle = \sum_{i=1}^{m} \alpha_{k,i} \langle \boldsymbol{\phi}(\mathbf{x}_i), \boldsymbol{\phi}(\mathbf{x}_{tt}) \rangle. \tag{3.22}$$

This can be re-written as

$$z_{tt,k} = \sum_{i=1}^{m} \alpha_{k,i} [\mathbf{k}_{ctt}]_i. \tag{3.23}$$

In vector form,

$$\mathbf{z}_{tt} = \mathbf{A}\mathbf{k}_{ctt}. \tag{3.24}$$

where $\mathbf{A} = [\boldsymbol{\alpha}_1, \cdots, \boldsymbol{\alpha}_m]$.

3.1.5 Outline of KPCA-KDE fault detection procedure

Tables 3.1 and 3.2 show the outline of KPCA-KDE-based fault detection procedure.

Table 3.1: Off-line training

TR1.	Obtain data under normal operating conditions (NOC) and scale the data using the mean and standard deviation of the columns of the data set which represent the different variables.
TR2.	Decide on the type of kernel function to use and determine the kernel parameter
TR3.	Construct the kernel matrix of the NOC data and centre it using (3.10)
TR4.	Obtain eigenvalues and their corresponding eigenvectors and rearrange them in a descending order
TR5.	Orthonormalise the eigenvectors using (3.11)
TR6.	Obtain nonlinear components using (3.13)
TR7.	Compute monitoring indices (T^2 and Q) based on the kernelised NOC data using (3.14) and (3.15)
TR8.	Determine control limits of T^2 and Q using (3.18) and (3.19)

To provide a more intuitive picture, a flowchart of the procedure is presented in Figure 3.1.

Table 3.2: On-line monitoring

TT1.	Acquire test sample $\mathbf{x}_{tt}$ and normalize using the mean and standard deviation values used in step 1 of the off-line stage
TT2.	Compute the kernel vector of the test sample using (3.20)
TT3.	Centre the kernel vector according to (3.21)
TT4.	Obtain the principal component of the test sample from (3.23)
TT5.	Compare the T^2 and Q of the test sample with their respective control limits obtained in the model development stage
TT6.	If T^2 or Q is less than its monitoring statistic, the process is in-control. If both exceed their control limits, the process is out-of-control and therefore fault identification is carried out to identify the source of the fault

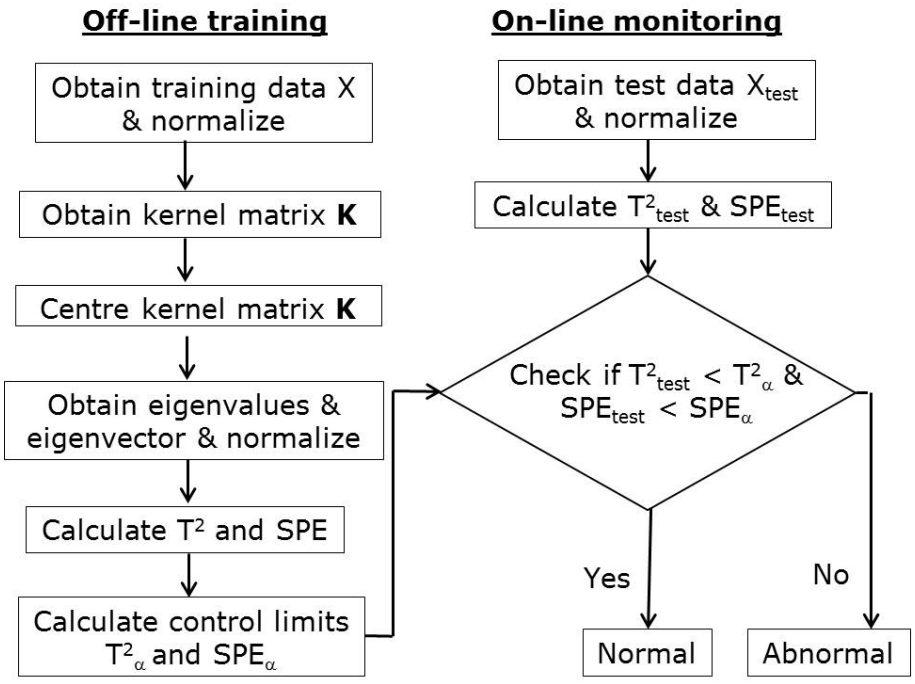


Figure 3.1: KPCA-KDE fault detection procedure.

3.2 KPCA-KDE based fault identification

Detecting a fault is just one aspect of monitoring a process. It is also important that the variables responsible for the fault are identified, after a detecting the fault. This step makes it possible for the root cause(s) of the fault to be located and removed in order to restore the process to its normal operating conditions.

Contribution plots have been used as a common means of fault identification in previous studies (Westerhuis et al., 2000; Miller et al., 1998). Essentially, contribution plots show the contributions of different process variables to the monitoring statistics. A high contribution of a given variable to the monitoring indices in a fault region shows that such a variable is strongly connected with the fault. A scheme using the Hotelling's T^2 and Q-statistics based on the PCA has been reported (Chiang et al., 2001; He et al., 2006). A combined index of the T^2 and Q statistics based on fault reconstruction in PCA and PLS have also been reported which showed to achieve better fault identification than using the T^2 and Q statistics singly (Yue and Qin, 2001; Dunia and Qin, 1998). Limited studies have also been carried out using linear dynamic systems like canonical variate analysis (Chiang et al., 2001; Jiang, Huang, Zhu, Fan and Braatz, 2015) for successful fault identification. Nevertheless, the drawback of these techniques is their inability to address process nonlinearities.

Fault identification based on kernel methods is not as straightforward as that of linear techniques and cannot be generalised from the linear approaches due the nonlinear relationship between the transformed and the original process variables(Lee et al., 2004). Efforts have been made to perform nonlinear fault identification using kernel PCA by calculating reconstruction error (Cho et al., 2005) and virtual scale factor (Choi et al., 2005). Kernel PCA fault identification strategy has also been developed based on approximating kernel functions using power series approximation. Although this approach makes possible the use of contribution plots similar to linear PCA, the complexity of the calculations become more involved as the problem size increases due to the calculation of kernel matrices. Thus, despite the scholarly efforts that have been made in the past, identification of faulty variables and source diagnosis is a problem that is still open to be solved in data-driven fault diagnosis studies involving nonlinear systems (Deng et al., 2013).

A recent study by Deng et al. (2013) identified fault variables using a sensitivity analysis approach (Petzold et al., 2006). The method is based on calculating the rate of change in system output variables resulting from changes in the problem causing parameters. Given an observation vector $\mathbf{x}_i \in \Re^n$ with n variables, the

contribution of the i^{th} variable to a monitoring metric using

$$T^2_{i,con} = x_i a_i \quad \text{and} \quad Q_{i,con} = x_i b_i, \tag{3.25}$$

where $a_i = \partial T^2/\partial x_i$ and $b_i = \partial Q/\partial x_i$.

In this work, the partial derivatives are obtained by differentiating the functions defining T^2 and Q at a given reference fault instance using complex step differentiation method. Let f be a function. The complex step differentiation (Martins et al., 2003) of f at a value $\mathbf{x}$ is given by the expression

$$\frac{\partial f}{\partial \mathbf{x}} \approx \frac{\text{lm}[f(\mathbf{x} + ih)]}{h}. \tag{3.26}$$

where h is the step size, hence ih is an imaginary step. Essentially, evaluating the function with a complex argument gives both the function value (the real part) and the derivative (the imaginary part). Unlike, the finite difference method, complex step differential does not involve a difference operation. Hence, very small step sizes can be used without losing accuracy.

Lyness (1967) and Lyness and Moler (1967) were the first to adopt the use of complex variables to develop estimates of derivatives in their work. The potential of this technique is now well recognised. Squire and Trapp (1998) used this theory to obtain an expression for the estimation of derivatives. It has been used by Newman et al. (1998) in a multidisciplinary environment. Work on the approach have also been presented in the works of Martins et al. (2000) and Martins et al. (2003). Although, using the approach for evaluating routines with complex arguments may require a high runtime, it is still a good generalisable approach for fault identification studies.

3.3 Application study

3.3.1 Overview of Tennessee Eastman process

The Tennessee Eastman (TE) process is a simulation of an actual industrial plant developed by Downs and Vogel of the Eastman Chemical Company (Downs and Vogel, 1993). It is probably the most popular benchmark process for assessing new monitoring and control techniques. The evidence of this can be seen in the many benchmark studies based on this process (Russell et al., 2000; Liu et al., 2012; McAvoy and Ye, 1994; Chiang et al., 2001; Yin et al., 2012; Kano et al., 2002). A schematic diagram of the process is shown in Figure 3.2.

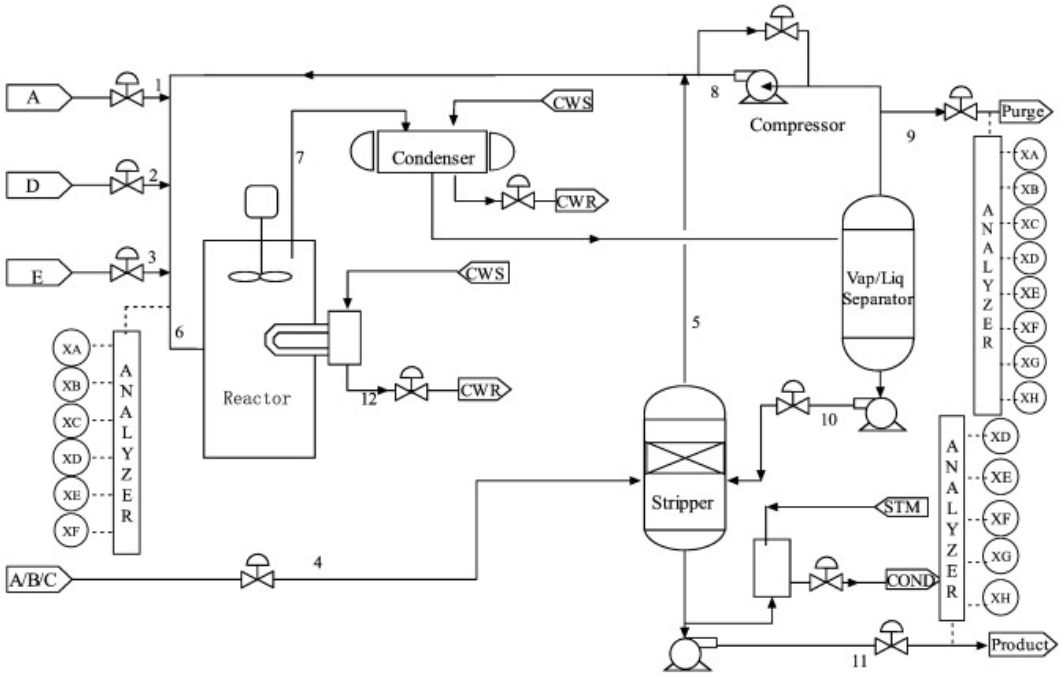


Figure 3.2: Schematic diagram of the TE process. Reaction products (Stream 7) are cooled in the condenser and sent to the separator where the vapour phase is cooled, partially purged (Stream 9) and recycled. Stream 4 strips unreacted reactants from Stream 10 and feeds them to the recycle stream while the products are collected from the exit.

The process has five main units - separator, compressor, reactor, stripper and condenser. There are eight components (A to H) in the process. The following equations

represents the of the process.

$$\begin{aligned} A(g) + C(g) + D(g) &\rightarrow G(liq) \qquad \text{Product} \\ A(g) + C(g) + E(g) &\rightarrow H(liq) \qquad \text{Product} \\ A(g) + E(g) &\rightarrow F(liq) \qquad \text{By product} \\ 3D(g) &\rightarrow 2F(liq) \qquad \text{By product} \end{aligned} \tag{3.27}$$

A, C, D and E are gaseous reactants which enter the reactor via streams 1, 4, 2 and 3 respectively. B is an inert component that feeds the reactor via stream 4. G and H are liquid products while F is a bye product. The cooled reactor product enters the separator where the vapour phase is condensed and recycled. Stream 9 is partially purged to avoid the accumulation of B and F in the recycle stream. The liquid phase in the separator enters the stripper. Stream 4 strips the reactants that have not reacted from stream 10 and the stripped components enter the recycle stream via stream 5. The products, G and H, are collected at the exit of the stripper.

The TE process consists of 960 observations which are sampled every 3 minutes. It also has 53 variables out of which 12 are continuous, 19 are composition and 12 are manipulated variables. The initial version of Downs and Vogel defined 20 faults but a 21^{st} fault was added by Chiang et al. (2001). All fault conditions were induced in the process after a 8 hour period of normal operation.

3.3.2 Application procedure

Five hundred samples collected during normal operation were used as training data and all 960 samples collected in each of the faulty conditions were used as test data. All 22 continuous and the 11 manipulated variables were used in this work. The 12^{th} manipulated variable (that is, agitation speed of the reactor's stirrer) which is constant, was excluded. A total of 20 faults in the process were studied. A description of the variables and faults studied are presented in Tables 3.3 and 3.4 respectively.

Table 3.3: TE process monitoring variables.

No.	Description	No.	Description
1	A feed (stream 1)	18	Stripper temperature
2	D feed (stream 2)	19	Stripper stream flow
3	E feed (stream 4)	20	Compressor work
4	Total feed (stream 4)	21	Reactor cooling water outlet temp.
5	Recycle flow (stream 8)	22	Condenser cooling water outlet temp.
6	Reactor feed rate (stream 6)	23	D feed flow (stream 2)
7	Reactor pressure	24	E feed flow (stream 3)
8	Reactor level	25	A feed flow (stream 1)
9	Reactor temperature	26	Total feed flow (stream 4)
10	Purge rate	27	Compressor recycle valve
11	Separator temperature	28	Purge valve
12	Separator level	29	Separator pot liquid flow (stream 10)
13	Separator pressure	30	Stripper liquid product flow
14	Separator under flow (stream 10)	31	Stripper steam valve
15	Stripper stream valve	32	Reactor cooling water flow
16	Stripper pressure	33	Condenser cooling water flow
17	Stripper under flow (stream 11)		

In this study, the number of PCs that explained over 90% of the total variance were retained. Based on this approach, 16 and 17 PCs were selected for PCA and KPCA respectively.

Another important parameter for kernel-based modelling is the choice of kernel and its width. The RBF kernel commonly used in previous studies (Stefatos and Hamza, 2007; Lee et al., 2004) was also used in this work. The kernel parameter value c was computed using the formula $c = Wn\sigma^2$, where W is a constant which depends on the monitored process, n is the dimension of the input data while σ^2 is the variance (Lee et al., 2004; Mika, Schölkopf, Smola, Müller, Scholz and Rätsch, 1999). The value of W was set at 40 with validation from the training data. The T^2 and Q statistics were jointly used for detecting faults. In other words, a successful detection of fault was deemed to have occurred when either the T^2 or Q or both statistics detects the fault. This is because the faults may not always manifest in the model space and the residual space simultaneously to the same degree due the different ways the abnormality may affect these spaces.

Table 3.4: Fault descriptions in the TE process.

Fault	Description	Type
1	A/C feed ratio, B composition constant	Step
2	B composition, A/C ratio constant	Step
3	D feed temperature	Step
4	Reactor cooling water inlet temperature	Step
5	Condenser cooling water inlet temperature	Step
6	A feed loss	Step
7	C header pressure loss-reduced availability	Step
8	A, B, C feed composition	Random variation
9	D feed temperature	Random variation
10	C feed temperature	Random variation
11	Reactor cooling water inlet temperature	Random variation
12	Condenser cooling water inlet temperature	Random variation
13	Reaction kinetics	Slow drift
14	Reactor cooling water valve	Sticking
15	Condenser cooling water valve	Sticking
16	Unknown	
17	Unknown	
18	Unknown	
19	Unknown	
20	Unknown	

3.3.3 Fault detection rule

Since measurements obtained from chemical processes are usually noisy, monitoring indices may exceed their thresholds randomly. This amounts to announcing the presence of a fault when no disturbance has actually occurred, that is, a false alarm. In other words, a monitoring index may exceed its threshold once but if no fault is present, the monitoring index may not stay above its threshold in subsequent measurements. Conversely, a fault has likely occurred if the monitoring index stays above its threshold in several consecutive measurements. A fault detection rule

is used to address the problem of spurious alarms (Tien et al., 2004; van Sprang et al., 2002; Choi and Lee, 2004). A detection rule also provides a uniform basis for comparing different monitoring methods. In this study, successful fault detection was counted when a monitoring measure stays above its control limit in at least two observations consecutively. All algorithms recorded a false alarm rate (FAR) of zero when tested with the training data based on this criterion. Computation of the metrics for evaluating the monitoring performance of the different techniques was therefore based on this criterion.

3.3.4 Computation of monitoring performance metrics

Three metrics were used to measure the performance of the different monitoring techniques: fault detection rate (FDR), false alarm rate (FAR) and detection delay. FDR is the percentage of samples that are faulty which are correctly identified. This was calculated using

$$FDR = \frac{n_{\text{fc}}}{n_{\text{tf}}} \times 100, \tag{3.28}$$

where n_{fc} denotes the number of fault samples detected correctly and n_{tf} is the total number of samples that are faulty. FAR is the percentage of samples that are normal which are detected as faulty or abnormal. It was calculated as

$$FAR = \frac{n_{\text{nf}}}{n_{\text{tn}}} \times 100, \tag{3.29}$$

where n_{nf} is the number of normal samples incorrectly detected as abnormal and n_{tn} is the total number of samples that are normal. Detection delay is the difference between the time of occurrence of the fault and the time of detection.

3.3.5 Results and discussion

KPCA-KDE-based fault detection

Fault detection based on KPCA-KDE is demonstrated using Faults 11 and 12 of the TE process. Fault 11 is a random variation in the reactor cooling water inlet temperature while Fault 12 is a random variation in the condenser cooling water inlet temperature. The monitoring charts for the two faults are shown in Figures. 3.3(a) and 3.3(b) respectively. The solid curves represent the fault signals while the dash-dot and dash lines represent the control limits at 99% confidence level based on Gaussian distribution and KDE respectively. It can be seen that in both cases, especially in the T^2 control charts, the KDE-based control limits are below the Gaussian distribution-based control limits. That is, the monitoring indices exceed the KDE-based control limits to a greater extent compared to the Gaussian distribution-based control limits. In other words, using the KDE-based control limits with the KPCA technique gives higher monitoring performance compared to using the Gaussian distribution-based control limits.

Table 3.5 shows the detection rates for PCA, PCA-KDE, KPCA, and KPCA-KDE for all 20 faults studied. The results show that the KDE versions have overall higher FDRs compared to the corresponding Gaussian distribution-based versions. Furthermore, in Table 3.6, it can be seen that the detection delays of the KDE-based versions are either equal to or lower than the non-KDE-based techniques. This implies that the approaches based on KDE-derived UCLs detected faults earlier than their Gaussian distribution-based counterparts. Thus, associating the control limits derive via KDE with the KPCA technique enhances the performance of the KPCA approach.

KPCA-KDE fault identification

KPCA-KDE-based fault identification is demonstrated using Fault 11 as an example. The occurrence of Fault 11 induces change in the reactor cooling water flow rate

Table 3.5: Fault Detection Rates (%).

Fault	PCA	PCA-KDE	KPCA	KPCA-KDE
1	99.75	99.75	99.75	99.75
2	98.25	98.75	98.63	98.63
3	0.13	0.88	1.63	1.75
4	99.88	99.88	99.88	99.88
5	23.63	25.75	26.38	26.88
6	99.88	99.88	99.88	99.88
7	99.88	99.88	99.88	99.88
8	96.88	97.38	98.00	98.00
9	0.25	1.13	1.63	2.25
10	35.75	41.63	51.13	53.50
11	74.75	77.50	78.13	79.88
12	97.50	97.63	97.50	97.63
13	95.50	95.75	95.38	95.63
14	99.75	99.75	99.75	99.75
15	0	1.13	2.13	2.88
16	27.50	36.13	39.75	44.62
17	92.50	93.88	93.00	93.50
19	5.50	9.88	10.13	13.50
20	49.25	53.00	57.13	57.75

which causes fluctuation in the reactor temperature. Both the T^2- and SPE-based contribution plots at sample 300 shown in Figure 3.4 identified the two fault variables correctly. Variables 9 and 32 are the reactor's temperature 32 and cooling water flow rate respectively. Although, it is possible for the control loops to compensate the change in the reactor temperature after a longer time has elapsed, the fluctuations in both variables affected early after the introduction of the fault were correctly identified by the contribution plots.

Table 3.6: Detection Delay, DD (min).

Fault	PCA	PCA KDE	KPCA	KPCA KDE
1	6	6	6	6
2	39	30	33	33
3	2346	1656	1725	1725
4	3	3	3	3
5	3	3	3	3
6	3	3	3	3
7	3	3	3	3
8	48	48	48	48
9	2346	1665	1725	1725
10	186	186	180	180
11	15	15	15	15
12	42	30	60	42
13	102	102	111	105
14	6	6	6	6
15	ND	1656	1725	1725
16	81	78	81	81
17	45	45	45	45
18	24	24	24	24
19	213	24	213	36
20	105	102	105	105

Note: ND, Not Detected.

Test of robustness

The robustness of the techniques was tested by varying two parameters: bandwidth and the number of PCs retained. Fig. 3.5 shows the monitoring charts for KPCA and KPCA-KDE with $W = 40$ and $W = 10$ for Fault 14. This fault represents sticking of the reactor cooling water valve, which is quit easily detected by most statistical process monitoring approaches. At $W = 40$, both KPCA and KPCA-KDE recorded zero false alarms (Fig. 3.5(a)). However, at $W = 10$, KPCA recorded a false alarm rate of 8.13% while the FAR for KPCA-KDE was still zero (Fig. 3.5(b)). Also, Table 3.7 shows that the KPCA recorded a similar high FAR when 25 principal

components were retained. Conversely, the KPCA-KDE approach still recorded zero false alarms.

Table 3.7: Monitoring results at different number of PCs retained.

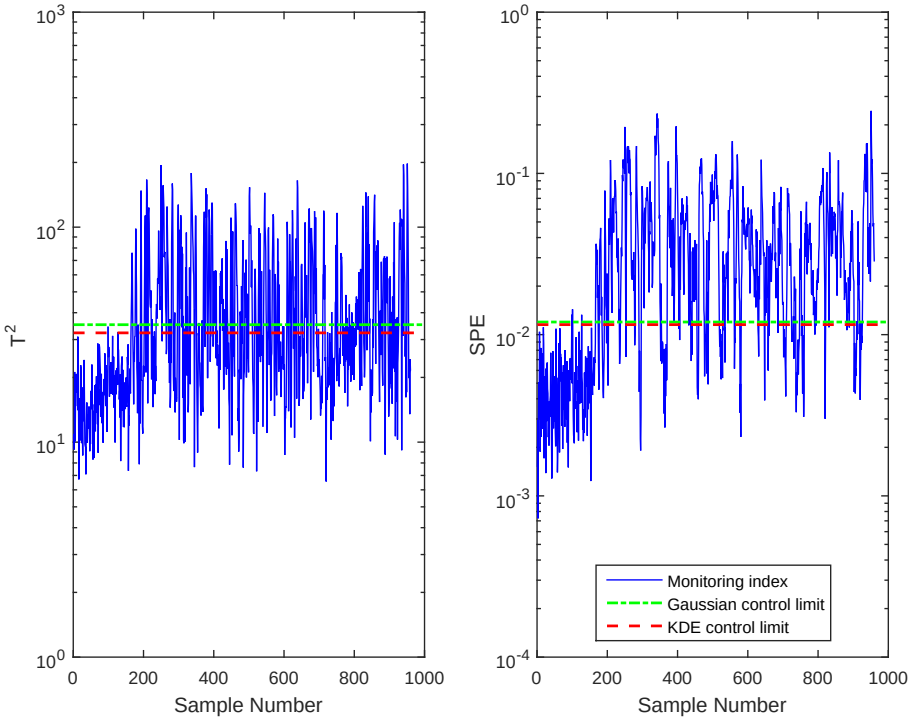
	KPCA			KPCA-KDE		
PCs	FDR	FAR	DD	FDR	FAR	DD
10	99.88	0	3	99.88	0	3
15	99.75	0	6	99.75	0	6
20	96.75	0	6	99.88	0	3
25	99.88	8.13	3	99.75	0	6

Thus, apart from generally providing higher FDRs and earlier detections, the KPCA-KDE is more robust than the KPCA technique with control limits based on the Gaussian assumption. A more sensitive technique is better for process operators since less faults will be missed. Secondly, when a fault is detected early, operators will have more time to establish the cause(s) that induced the fault and take remedial actions while the process can still be controlled. Thirdly, although methods are available for obtaining optimum design parameters for developing process monitoring models, there is no guarantee that the optimum values are used all the time. The reason for this may range from lack of experience of personnel to lack of or limited understanding of the process itself. Therefore, the more robust a technique is the better it is for process operations.

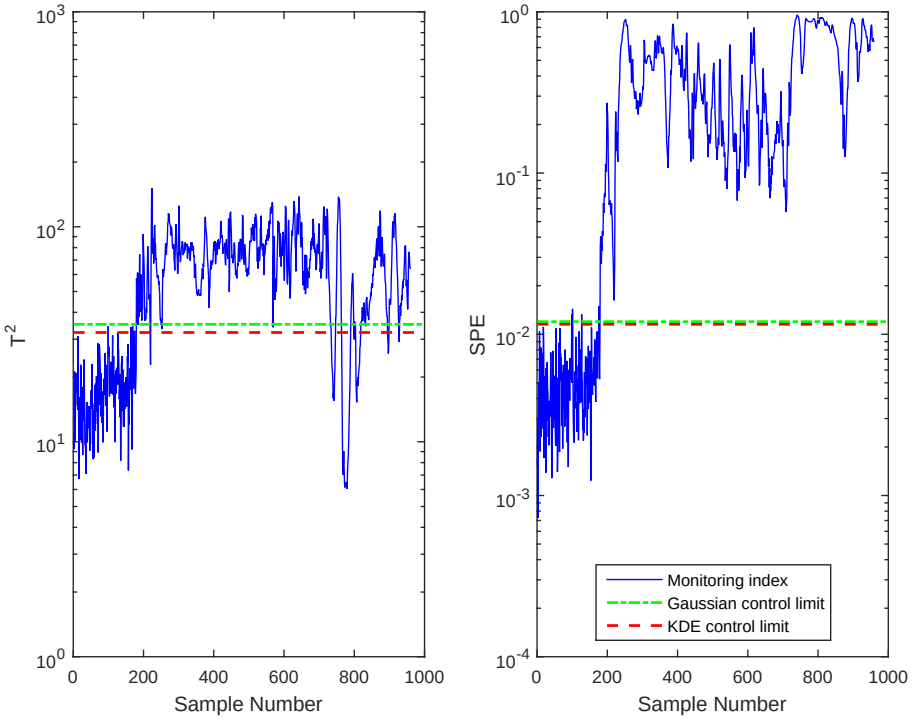
3.4 Concluding remarks

This chapter investigated nonlinear process fault detection and identification using the KPCA-KDE technique. In this approach, the thresholds used for constructing control charts were derived directly from the PDFs of the monitoring indices instead of using thresholds based on the Gaussian distribution. The technique was applied to the benchmark TE process and its fault detection performance was compared with the KPCA technique based on the assumption of Gaussian distributed variables.

The overall results show that KPCA-KDE detected faults more and at shorter times from when faults occur compared to the KPCA approach whose control limits were based on the Gaussian distribution. The study also shows that the control limits based on KDE are more robust than those based on the Gaussian assumption because the former follow the actual distribution of the monitoring statistics more closely. In general, the work corroborates the claim that using KDE-based control limits give better monitoring results in nonlinear processes than using control limits based on the Gaussian assumption. A generalisable approach for computing variable contributions in fault identification studies that centre on multivariate statistical methods was also demonstrated. However, the KPCA approach does not capture process dynamics. This will be addressed in Chapter 4.

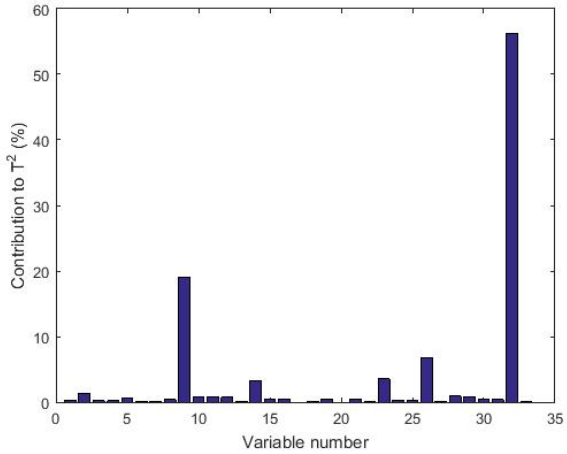


(a)

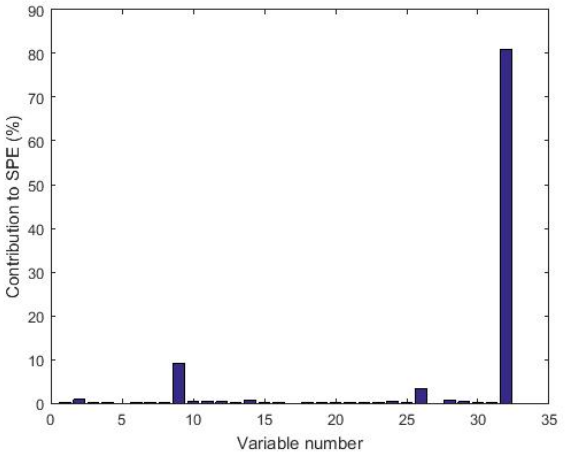


(b)

Figure 3.3: KPCA-based control charts showing monitoring indices and Gaussian assumption/KDE-based control limits for (a) Fault 11 (b) Fault 12. The KDE-based control limits are below the Gaussian assumption-based thresholds in both faults and give higher fault detection rates.

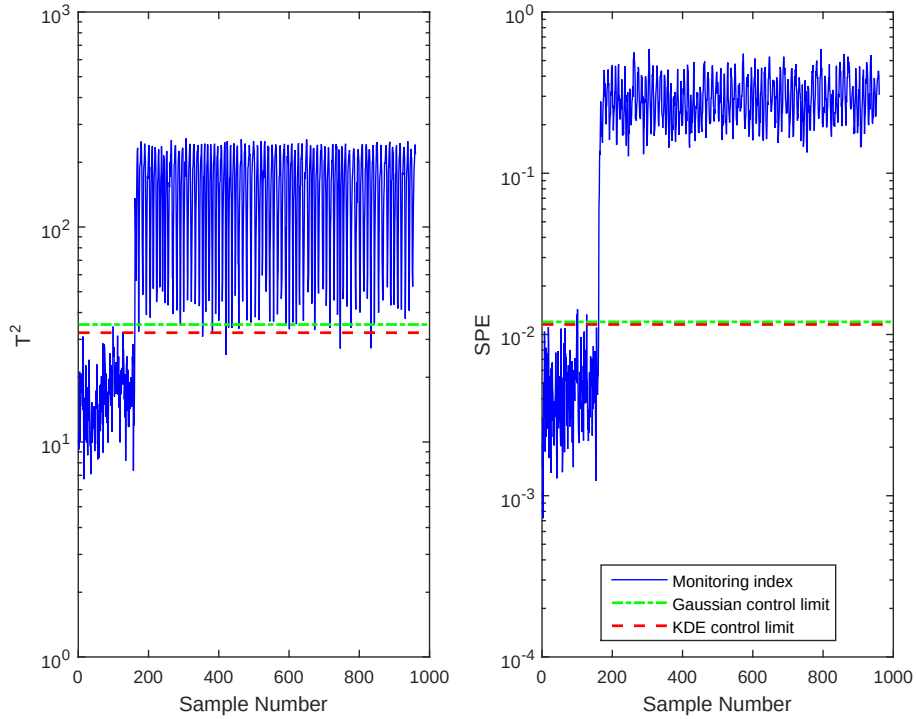


(a)

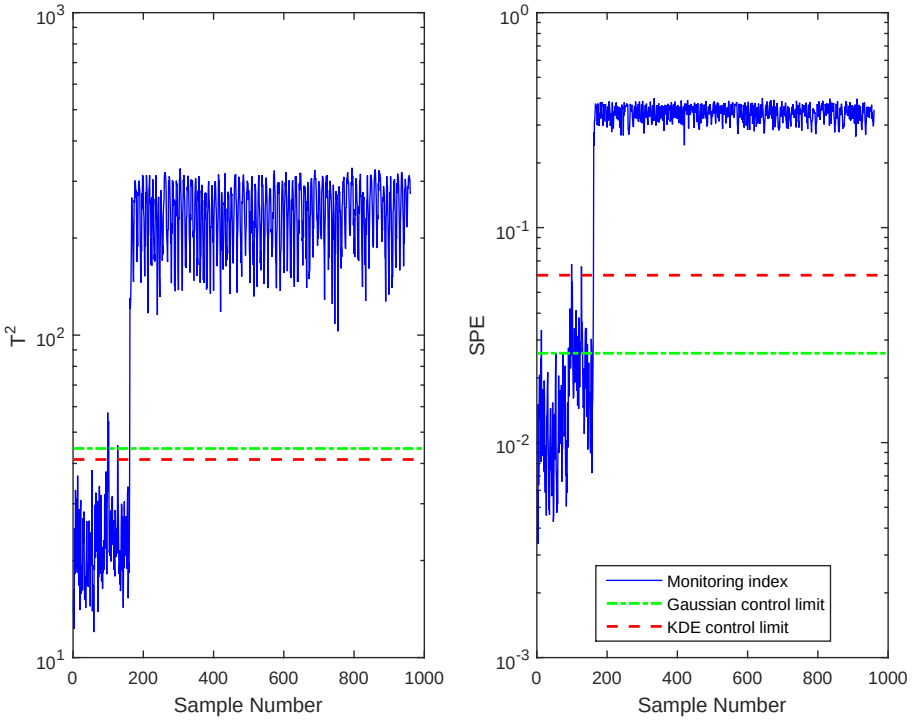


(b)

Figure 3.4: Plot showing KPCA-KDE-based contributions to T^2 and SPE for Fault 11 of the TE process at sample number 300; (a) T^2-based contribution plot (b) SPE-based contribution plot. Both plots correctly identified variables 9 and 32 mostly responsible for the faulty condition.



(a)



(b)

Figure 3.5: KPCA-KDE control charts for Fault 14 of the TE process; (a) control chart using $W = 40$ in the formula $c = Wn\sigma^2$ (b) control chart using $W = 10$ in the formula $c = Wn\sigma^2$. The KPCA-KDE FARs do not change drastically with changing operating parameters which makes it more robust than the KPCA approach based on Gaussian assumption control limits.

Chapter 4

Statistical process monitoring using linear latent variable CVA

It is not impossible for process measurements to have both linear and dynamic variable relationships. Therefore, in this chapter, the linear latent variable CVA (LLV-CVA) technique is proposed as a dynamic latent variable method to capture linear variable cross-correlations and dynamic autocorrelations simultaneously. In this approach, PCA is used to capture linear cross-correlations while CVA is used to capture dynamic autocorrelations. Application results of the technique on the TE process are compared with results obtained with DPCA and CVA.

4.1 Introduction

Dynamic monitoring techniques extract only dynamic relationships. On the other hand, steady state models focus on static relationships without considering any time lags or delays. However, a complex industrial process may exhibit more than one data characteristic. This could mean that data generated from the process have both linear and dynamic features. Hence, developing a monitoring technique which simultaneously accounts for cross-correlations arising from steady state behaviour as well as autocorrelations due to process dynamics is a good idea.

Li et al. (2011) suggested a dynamic latent variable model using dynamic factor analysis to restrict the dynamic variation in a reduced subspace in the prediction of time series. This approach involves, firstly, extracting the dynamics in a variable space into autocorrelated latent variables. Secondly, the autocorrelated PCA is used for extracting dynamic principal factors according to their autocovariance. However, this approach is not still able to extract dynamic relations exhaustively, due to the limitations of using time lags to capture process dynamics (Jiang, Zhu, Huang, Paulson and Braatz, 2015).

In this chapter, PCA is used to extract static correlations in the variable space to obtain linear latent variables (LLVs). Then, CVA is applied to the LLVs to extract dynamic autocorrelations. The proposed approach is referred to as linear latent variable-CVA (LLV-CVA). Modelling and fault detection based on the LLV-CVA is presented in this chapter and its performance is compared with DPCA and CVA. The chapter ends with concluding remarks drawn from the application study.

4.2 Linear latent variable-CVA modelling

Recall that from (2.3), the transformed data (that is, the matrix of scores or linear latent variables) derived from the observation dataset $\mathbf{X}$ is given by $\mathbf{Z} = \mathbf{XP}$. If $\mathbf{z}_l$ is the l^{th} column vector of $\mathbf{Z}$, computation of the past (p) and future (f) observation vectors in the CVA step is obtained as follows:

$$\mathbf{z}_{p,l} = \begin{bmatrix} \mathbf{z}_{l-1} \\ \mathbf{z}_{l-2} \\ \vdots \\ \mathbf{z}_{l-p} \end{bmatrix} \in \Re^{rp} \text{ and } \mathbf{z}_{f,l} = \begin{bmatrix} \mathbf{z}_{l} \\ \mathbf{z}_{l+1} \\ \vdots \\ \mathbf{z}_{l+f-1} \end{bmatrix} \in \Re^{rf}, \tag{4.1}$$

where r is the dimension of the linear latent variables retained which contains eigenvectors corresponding to the first r eigenvalues of $\mathbf{Z}$. The mean centred past and

future observation vectors, $\hat{\mathbf{z}}_{p,l}$ and $\hat{\mathbf{z}}_{f,l}$ respectively are calculated as

$$\hat{\mathbf{z}}_{p,l} = \mathbf{z}_{p,l} - \bar{\mathbf{z}}_{p,l} \quad \text{and} \quad \hat{\mathbf{z}}_{f,l} = \mathbf{z}_{f,l} - \bar{\mathbf{z}}_{f,l}, \tag{4.2}$$

where $\bar{\mathbf{z}}_{p,l}$ and $\bar{\mathbf{z}}_{f,l}$ denote the sample means of $\mathbf{z}_{p,l}$ and $\mathbf{z}_{f,l}$ respectively. The past $\mathbf{A}_p$ and future $\mathbf{A}_f$ matrices were obtained by arranging the corresponding past and future vectors together in columns as follows:

$$\mathbf{A}_p = [\hat{\mathbf{z}}_{p,p+1},\ \hat{\mathbf{z}}_{p,p+2}, \ldots \hat{\mathbf{z}}_{p,p+W}] \in \Re^{rp \times W}, \tag{4.3}$$

$$\mathbf{A}_f = [\hat{\mathbf{z}}_{f,p+1},\ \hat{\mathbf{z}}_{f,p+2}, \ldots \hat{\mathbf{z}}_{f,p+W}] \in \Re^{rf \times W}, \tag{4.4}$$

where $\mathbf{A}_p$ and $\mathbf{A}_f$ are past and future truncated Hankel matrices for N observations. To avoid singularity of $\mathbf{\Sigma}_{pp}$ and $\mathbf{\Sigma}_{ff}$, parameters r, p and f have to satisfy $\{rp, rf\} < N - p - f + 1$. The number of columns W for the truncated Hankel matrices were computed using $W = N - f - p + 1$.

The sample covariances and cross-covariances of the past and future matrices were estimated as

$$\mathbf{\Sigma}_{pp} = \mathbf{A}_p \mathbf{A}_p^T \left(W - 1\right)^{-1}, \tag{4.5}$$

$$\mathbf{\Sigma}_{ff} = \mathbf{A}_f \mathbf{A}_f^T \left(W - 1\right)^{-1}, \tag{4.6}$$

$$\mathbf{\Sigma}_{fp} = \mathbf{A}_f \mathbf{A}_p^T \left(W - 1\right)^{-1}. \tag{4.7}$$

The correlation between the linear combinations of the future observation vectors $\mathbf{a}^T\hat{\mathbf{z}}_{f,l}$ and those of the past observation vectors $\mathbf{b}^T\hat{\mathbf{z}}_{p,l}$ were expressed as

$$\rho = \max_{\mathbf{a},\mathbf{b}} \frac{\mathbf{a}^T \mathbf{\Sigma}_{fp} \mathbf{b}}{\left(\mathbf{a}^T \mathbf{\Sigma}_{ff} \mathbf{a}\right)^{1/2} \left(\mathbf{b}^T \mathbf{\Sigma}_{pp} \mathbf{b}\right)^{1/2}}. \tag{4.8}$$

Assuming that $\mathbf{u} = \mathbf{\Sigma}_{ff}^{1/2}\mathbf{a}$ and $\mathbf{v} = \mathbf{\Sigma}_{pp}^{1/2}\mathbf{b}$, the CVA optimisation problem can be

represented as

$$\max_{\mathbf{u},\mathbf{v}} \quad \mathbf{u}^T \left(\mathbf{\Sigma}_{ff}^{-1/2} \mathbf{\Sigma}_{fp} \mathbf{\Sigma}_{pp}^{-1/2} \right) \mathbf{v}, \tag{4.9}$$

$$\text{s.t.} \quad \mathbf{u}^T \mathbf{u} = \mathbf{v}^T \mathbf{v} = 1, \tag{4.10}$$

where the solution, $\mathbf{u}$ and $\mathbf{v}$ are the left and right singular vectors of the Hankel matrix $\mathbf{H}_l = \mathbf{\Sigma}_{ff}^{-1/2} \mathbf{\Sigma}_{fp} \mathbf{\Sigma}_{pp}^{-1/2}$. Singular value decomposition (SVD) of $\mathbf{H}_l$ gives:

$$\mathbf{H}_l = \mathbf{\Sigma}_{ff}^{-1/2} \mathbf{\Sigma}_{fp} \mathbf{\Sigma}_{pp}^{-1/2} = \mathbf{U} \mathbf{\Omega} \mathbf{V}^T. \tag{4.11}$$

where $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices of the left and right singular vectors and $\mathbf{\Omega}$ is a diagonal matrix whose elements are the singular values of $\mathbf{H}_k$. Reordering the singular values in descending order and rearranging the columns of the associated singular vectors makes the first q columns of $\mathbf{V}$ the ones with top largest correlations with those of $\mathbf{U}$. This gives a reduced dimensional matrix $\mathbf{V}_q$ such that $(q < rp)$.

The transformation matrices $\mathbf{C}$ and $\mathbf{D}$ for converting the rp-dimensional past matrices to the q-dimensional state variables and the residuals respectively were computed as

$$\mathbf{C} = \mathbf{V}_q^T \Sigma_{pp}^{-1/2} \in \Re^{q \times rp}, \tag{4.12}$$

$$\mathbf{D} = \left(I - \mathbf{V}_q \mathbf{V}_q^T \right) \Sigma_{pp}^{-1/2} \in \Re^{rp \times rp}, \tag{4.13}$$

For a given latent variable vector, the states and residuals are defined as

$$\mathbf{z}_l^* = \mathbf{C}.\hat{\mathbf{z}}_{p,l} \quad \text{and} \quad \mathbf{e}_l^* = \mathbf{D}.\hat{\mathbf{z}}_{p,l}. \tag{4.14}$$

4.3 Method of fault detection

The Hotellings T^2 and the Q statistic or squared prediction error (SPE) and their control limits are used for process monitoring in LLV-CVA. They are computed as

follows:

$$T_l^2 = \mathbf{z}_l^{*T}\mathbf{z}_l^* \quad \text{and} \quad Q_l = \mathbf{e}_l^{*T}\mathbf{e}_l^*. \tag{4.15}$$

To avoid the Gaussian distribution assumption, the control limits of T^2 and Q are computed through kernel density estimation (Odiowei and Cao, 2010).

Given a test observation $\mathbf{x}_t$, the corresponding latent variable is computed as

$$\mathbf{z}_t = \mathbf{x}_t\mathbf{P}, \tag{4.16}$$

and the past vector $\mathbf{z}_{p,l}^{(t)}$ is computed similar to (4.1) and (4.2). The corresponding state and residual test vectors are calculated using

$$\mathbf{z}_l^{*(t)} = \mathbf{J}.\hat{\mathbf{z}}_{p,l}^{(t)} \quad \text{and} \quad \mathbf{e}_l^{*(t)} = \mathbf{L}.\hat{\mathbf{z}}_{p,l}^{(t)}. \tag{4.17}$$

4.3.1 LLV-CVA-based process monitoring steps

Off-line training

A6–1 Obtain normal operating data.

A6–2 Mean centre data and normalize to unit variance.

A6–3 Compute covariance matrix of the pre-processed data using (2.1) and perform eigen decomposition using (2.2).

A6–4 Compute latent variables using (2.3).

A6–5 Expand latent variable vector at each time point l to obtain past (p) and future (f) measurements using (4.1).

A6–6 Perform SVD on the scaled Hankel matrix using (4.11) and determine number of states to retain.

A6–7 Determine state variables and residuals using (4.14).

A6–8 Compute monitoring indices using (4.15) and their KDE-based thresholds similar to (3.18) and (3.19) .

On-line monitoring

B6–1 Obtain and pre-process test data with same mean and standard deviation used for training data.

B6–2 Compute latent variable of test data using (4.16).

B6–3 Compute the past vector similar to (4.1) and (4.2) .

B6–4 Calculate state and residual using (4.17).

B6–5 Compute T_l^2 and Q_l of the test data using (4.15).

B6–6 Monitor process by comparing values of T_l^2 and Q_l of the test data against their thresholds. A fault is detected if either or both indices exceed their thresholds.

4.4 Application study

In this section the LLV-CVA approach was tested using the TE challenge process (Downs and Vogel, 1993; Lyman and Georgakis, 1995; Wilson and Irwin, 2000; Liu et al., 2012; Jing et al., 2014) and the results obtained are compared with DPCA and CVA-based results.

4.4.1 Parameters selection

Two parameters are required for implementing the CVA. These are the lag length (i.e. number of past and future lags used) and the number of states retained. The lag length of a time series can be estimated from its sample autocorrelation function

(ACF). Assuming an observed series $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_N$. The autocorrelation coefficient at a given time interval or lag l ($\hat{r}_l$) is given by

$$\hat{r}_l = \frac{\sum_{i=1}^{N-l} (\mathbf{x}_i - \bar{\mathbf{x}}) (\bar{\mathbf{x}}_{i+l} - \bar{\mathbf{x}})}{\sum_{i=1}^{N} (\mathbf{x}_i - \bar{\mathbf{x}})^2}, \tag{4.18}$$

where $\bar{\mathbf{x}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i$ is the sample mean of the N observations. Notice that N is used instead of the usual $N - 1$ in calculating the sample mean. This convention is common in time series applications and also because N is usually not small. Therefore, the difference between using N and $N - 1$ will be little.

Autocorrelation coefficient measures the correlation between values of an observation at two different time periods. A plot of autocorrelation coefficients for several time periods (i.e. $l = 1, 2, 3, \ldots$) is referred to as the autocorrelation function (ACF) or correlogram. In other words, the sample ACF summarises the autocorrelation of a time series at different lags. Therefore, the number of past lags that correlate significantly with the time series can be determined from the sample ACF. The autocorrelation function of the training data with 95% confidence level (represented by the solid horizontal lines) is shown in Figure 4.1. Based on the ACF, a lag length of 15 is used in this study. Hence, the values of p and f were set at 15.

It is proposed that determining the states to retain be based on the number of largest singular values in matrix $\mathbf{H}_l$ in (4.11) (Negiz and Cinarl, 1998). However, Figure 4.2 shows that the normalised singular values decrease very slowly in this particular case. Hence, such an approach may not give a realistic model (Odiowei and Cao, 2010). Furthermore, the choice of states to retain is not critical because both indices are used for monitoring. A fault that does not manifest in the model phase may manifest in residual phased and will be detected by the Q statistic. Similarly, a fault that does not have significant impact on the residual space may manifest significantly in the model phase and be detected by the T^2 statistic. Consequently, 26 states were retained to minimise the rate of false alarms under normal operating conditions. All

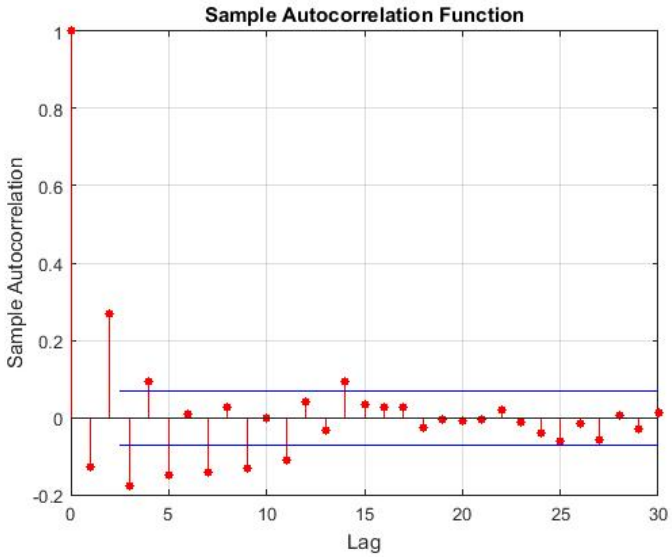


Figure 4.1: Plot showing sample autocorrelation function of the training data with 95% confidence level. Autocorrelation died out at the 15^{th} time period. Hence the length of future f and past g time lag was fixed at 15.

control charts for monitoring the process were constructed at 99% confidence level.

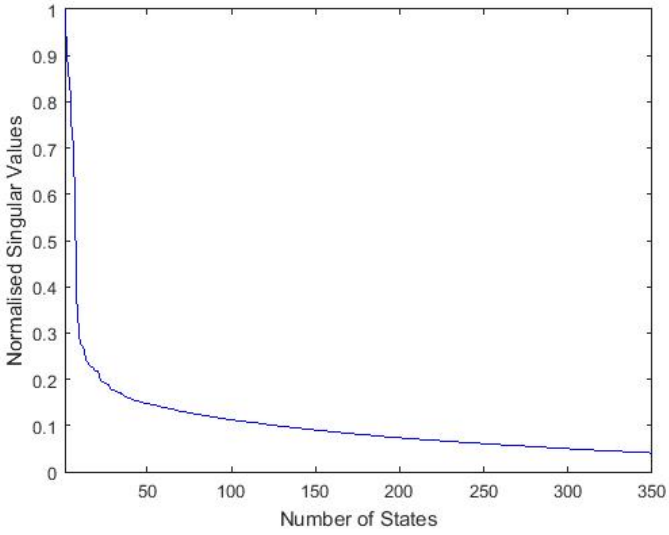


Figure 4.2: Plot of normalised singular values of the training data used for determining number of states to retain. Since the singular values decreased very slowly, 26 states were retained to minimise false alarms.

The DPCA model was constructed with 2 lags. This agrees with the view that one or two lags are appropriate for DPCA-based process monitoring even though a higher lag-order may be required in nonlinear processes (Chiang et al., 2001). Also,

a total of 28 principal components (PCs) were retained in the DPCA technique based on the cumulative variance of the eigenvalues that accounted for over 90% of the total variance. The design parameters used for model development in this study are presented in Table 4.1. Successful fault detection was assumed when a

Table 4.1: Table showing summary of design parameters for the different monitoring methods.

Design Parameter	Value
Order of time lag (DPCA)	2
Number of PCs (DPCA), (90% of total variance)	28
Number of states (CVA, LLV-CVA)	16
Number of time lags (CVA, LLV-CVA)	15

monitoring statistic exceeds its threshold in at least three consecutive observations. This fault detection rule provided the common basis for comparing the monitoring techniques investigated.

4.4.2 Results and discussion

Tables 4.2 and 4.3 show the monitoring performances of DPCA, CVA, and LLV-CVA. Table 4.2 compares the fault detection rates (FDRs) of the techniques investigated while Table 4.3 compares the fault detection times (DTs) of all three approaches. FDR is the percentage of faulty observations that are correctly detected. FAR represents the observations recorded as faults when no faults were actually present, while detection time is the time that passes before a fault is detected.

In general, the results showed that the CVA-based methods (i.e. CVA and LLV-CVA) gave higher FDRs and lower detection times compared to the DPCA. In other words, faults were detected more and earlier by the CVA-based techniques compared to the DPCA approach. This is attributable to the inability of the DPCA to adequately capture the dynamics of the process as was highlighted in the introduction. The difference in performance between the DPCA and the CVA-based techniques is particularly significant in Faults 3, 9, and 15. These faults are more difficult

to detect in the TE process because they cause little variation in the measured variables.

In Fault 3, the FDR of the DPCA, CVA and LLV-CVA are 0%, 65.13% and 65.88% respectively. The equivalent results for Fault 9 are 0%, 88.63% and 90.13% respectively. In Fault 3, both CVA and LLV-CVA records a detection delay of 15 minutes while the DPCA did not detect the fault at all. However, results of the CVA and the LLV-CVA were largely similar, although, the LLV-CVA gave slightly higher FDRs than the CVA in Faults 3, 9, 13, 15, 17 and 18. The LLV-CVA also recorded slightly lower detection times in Faults 9, 13, 15, 17 and 18. Based on the fault detection rule of three consecutive readings above the control limit employed, all the techniques recorded zero false alarms. Evidently, the LLV-CVA enhanced both the level of fault detected and time of detection in a number of faults compared to the DPCA. However, incorporating a linear step in CVA to extract static correlations in addition to dynamic autocorrelations seem to have improved performance only very slightly. Since the techniques investigated in this chapter are linear methods, they do not consider process nonlinearities. Therefore, a nonlinear dynamic approach is considered in the next chapter.

Figs. 4.3(a), 4.3(b), and 4.3(c) show the control charts for Fault 9 for DPCA, CVA, and LLV-CVA respectively. The solid signal is the monitoring statistic while the dash line is the threshold (control limit). The difference between the DPCA and CVA-based methods can be seen clearly. The control charts show that the DPCA did not detect Fault 9. The solid signal is below the control line which erroneously implies a normal condition. The T^2 statistic of both the CVA and LLV-CVA also did not detect the fault. However, their charts based on the Q-statistic clearly show an abnormal deviation from about the 172^{nd} sample. This is the instance when the monitoring statistic exceeds the control limit.

Table 4.2: Table comparing FDRs of DPCA, CVA and LLV-CVA for Faults 1 to 20 of the TE process. LLV-CVA outperformed the DPCA and give FDRs comparable with the CVA. **Note:** All techniques gave zero false alarms.

	FDR (%)		
Fault	DPCA	CVA	LLV-CVA
1	99.50	99.63	99.63
2	98.13	99.50	99.50
3	0	65.13	65.88
4	99.75	99.75	99.75
5	21.63	99.75	99.75
6	99.75	99.75	99.75
7	99.75	99.75	99.75
8	96.75	98.75	98.75
9	0	88.63	90.13
10	32.38	96.38	96.38
11	86.75	99.25	99.25
12	97.38	99.38	99.38
13	95.25	96.00	96.13
14	99.63	99.75	99.75
15	0	99.50	99.63
16	28.75	99.13	99.13
17	95.63	98.00	98.13
18	98.88	99.13	99.25
19	8.38	99.75	99.75
20	48.63	97.38	97.38

4.5 Concluding remarks

In this chapter, the latent variable-CVA technique (LLV-CVA) was proposed. It involved performing PCA on process measurements to extract latent variables. The latent variables were then used as input for canonical variate analysis. The objective of the technique was to capture both static cross-correlations and dynamic serial correlations. The results show that the LLV-CVA method performed better than the DPCA technique and gave results that were comparable with the CVA approach. The techniques considered in this chapter capture static and dynamic relations but

Table 4.3: Table comparing fault detection times (min) of DPCA, CVA and LLV-CVA for Faults 1 to 20 of the TE process. LLV-CVA performs better than the DPCA. However, detection times of the LLV-CVA and CVA are not significantly different.

Fault	DPCA	CVA	LLV-CVA
1	12	9	9
2	45	12	12
3	ND	15	15
4	6	6	6
5	6	6	6
6	6	6	6
7	6	6	6
8	57	30	30
9	ND	39	36
10	180	87	87
11	18	18	18
12	63	15	15
13	114	96	93
14	9	6	6
15	ND	12	9
16	84	21	21
17	48	48	45
18	27	21	18
19	132	6	6
20	108	63	63

Note: ND, not detected

do not address nonlinearities. However, many real processes are inherently nonlinear and dynamic. Therefore, nonlinear dynamic processes will be presented in the next chapter.

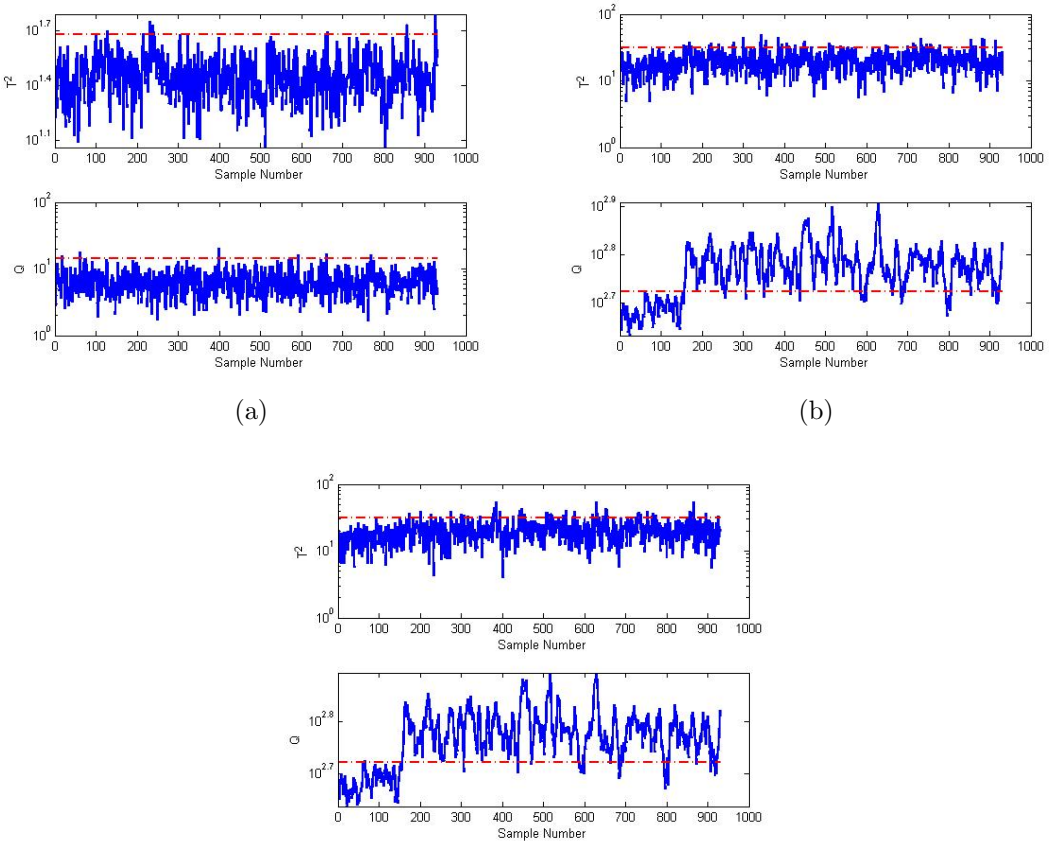


Figure 4.3: Control charts for Fault 9 (a) DPCA (b) CVA (c) LLV-CVA.

Chapter 5

Kernel canonical variate analysis using QR decomposition

This chapter presents the kernel CVA (KCVA) technique based on QR decomposition (KCVA-QRD). This algorithm is the first of two strategies proposed in this thesis for monitoring nonlinear dynamic processes. The technique is also intended to deal with the formation of singular matrices associated with implementing canonical analysis on kernel data. Details of the proposed approach and its mathematical formulation are presented. The technique is applied to the TE process and the results obtained are discussed in comparison with two other approaches: performing KCVA on regularised kernel data (KCVA-REG) and DKPCA.

5.1 Introduction

In Chapter 3, the KPCA-KDE technique was developed for nonlinear process monitoring. In Chapter 4, process dynamics and linear variable relationships were addressed simultaneously using a dynamic linear modelling approach (LLV-CVA). In this chapter, the past and future kernel matrices constructed from process measurements are factorised using QR decomposition. This avoids the need for regularisation in kernel-based canonical analysis techniques. The performance of the proposed ap-

proach is tested using the TE process. The results obtained are then compared with DKPCA and the kernel CVA technique based on regularised kernel data. A brief discussion on QR decomposition is presented in the next section.

5.2 QR decomposition

Basically, QR decomposition is used to factor a matrix as a product of two matrices. Given a matrix $\mathbf{A}$, its QR decomposition yields

$$\mathbf{A} = \mathbf{Q} \times \mathbf{R} \tag{5.1}$$

where $\mathbf{Q}$ is a unitary matrix and $\mathbf{R}$ is an upper triangular matrix (that is, $\mathbf{A} = 0 \quad \text{for} \quad j < i$). If $\mathbf{A} \in \Re^{m \times n}$, then $\mathbf{Q} \in \Re^{m \times m}$ and $\mathbf{R} \in \Re^{m \times n}$.

QR decomposition can be used for matrix inversion because if $\mathbf{A} = \mathbf{QR}$, then,

$$\mathbf{A}^{-1} = \mathbf{Q}^{-1}\mathbf{R}^{-1} = \mathbf{R}^{-1}\mathbf{Q}^{T}. \tag{5.2}$$

Since $\mathbf{R}$ is triangular, it is easy to invert.

Essentially, if a matrix $\mathbf{A}$ is invertible, its inverse is given by $\mathbf{A}^{-1}$. However, if it is not invertible, its inverse can be approximate by obtaining its pseudo-inverse as

$$\mathbf{A}^{+} = \left(\mathbf{A}^{T}\mathbf{A}\right)^{-1}\mathbf{A}^{T}, \tag{5.3}$$

where the superscript + represents pseudo-inverse. When $\mathbf{A}$ is invertible, the pseudo-inverse is in theory, the same as the inverse but when $\mathbf{A}$ is not invertible, the pseudo-inverse is much more robust than the inverse.

5.3 Kernel CVA with QR decomposition

Kernel-based CVA seeks to extract state variables which also capture nonlinear characteristics in the observation dataed data using nonlinear kernel transformation and CVA. The description of the technique is presented in this section.

To address time correlations, an observation vector $\mathbf{x}$ having a dimension of d is expanded at every time instant l to generate past $\mathbf{x}_{(p,l)}$ and future $\mathbf{x}_{(f,l)}$ observation vectors

$$\mathbf{x}_{(p,l)} = \begin{bmatrix} \mathbf{x}_{(l-1)} \\ \mathbf{x}_{(l-2)} \\ \vdots \\ \mathbf{x}_{(l-p)} \end{bmatrix} \in \Re^{dp} \quad \text{and} \quad \mathbf{x}_{(f,l)} = \begin{bmatrix} \mathbf{x}_{(l)} \\ \mathbf{x}_{(l+1)} \\ \vdots \\ \mathbf{x}_{(l+f-1)} \end{bmatrix} \in \Re^{df}. \tag{5.4}$$

These p and f vectors are then normalised to have a mean of 0 so that variables having greater absolute values do not have a dominant effect

$$\hat{\mathbf{x}}_{(p,l)} = \mathbf{x}_{(p,l)} - \bar{\mathbf{x}}_{(p,l)} \quad \text{and} \quad \hat{\mathbf{x}}_{(f,l)} = \mathbf{x}_{(f,l)} - \bar{\mathbf{x}}_{(f,l)}. \tag{5.5}$$

where $\bar{\mathbf{x}}_{(p,l)}$ and $\bar{\mathbf{x}}_{(f,l)}$ are the sample means of $\mathbf{x}_{(p,l)}$ and $\mathbf{x}_{(f,l)}$ respectively. The past and future vectors are then arranged together in columns to obtain the corresponding past and and future matrices, $\mathbf{X}_p$ and $\mathbf{X}_f$ respectively

$$\mathbf{X}_p = \left[\hat{\mathbf{x}}_{(p,p+1)},\ \hat{\mathbf{x}}_{(p,p+2)}, \ldots \hat{\mathbf{x}}_{(p,p+W)}\right] \in \Re^{dp \times W}, \tag{5.6}$$

$$\mathbf{X}_f = \left[\hat{\mathbf{x}}_{(f,p+1)},\ \hat{\mathbf{x}}_{(f,p+2)}, \ldots \hat{\mathbf{x}}_{(f,p+W)}\right] \in \Re^{df \times W}, \tag{5.7}$$

where the columns of the truncated Hankel matrices for N observations is given by $W = N - f - p + 1$. To capture nonlinear relations, a nonlinear mapping of the generated p and f vectors is performed, $\Phi_1 : \Re^{dp} \rightarrow F$ and $\Phi_2 : \Re^{df} \rightarrow F$ respectively, where, Φ_1 and Φ_2 are the nonlinear maps. The kernel trick is then used to construct

kernel matrices ($\mathbf{K}_p$ and $\mathbf{K}_f$) (Schölkopf et al., 1998; Lee et al., 2004):

$$\mathbf{K}_p = \langle \Phi_1 (\mathbf{X}_p), \Phi_1 (\mathbf{X}_p) \rangle , \tag{5.8}$$

$$\mathbf{K}_f = \langle \Phi_2 (\mathbf{X}_f), \Phi_2 (\mathbf{X}_f) \rangle , \tag{5.9}$$

where the elements of these kernel matrices are defined as

$$(\mathbf{K}_p)_{ji} = \left\langle \Phi_1 \left(\hat{\mathbf{x}}_{(p,p+j)}\right), \Phi_1 \left(\hat{\mathbf{x}}_{(p,p+i)}\right) \right\rangle ,$$
$$(\mathbf{K}_f)_{ji} = \left\langle \Phi_2 \left(\hat{\mathbf{x}}_{(f,p+j)}\right), \Phi_1 \left(\hat{\mathbf{x}}_{(f,p+i)}\right) \right\rangle ,$$

for all $j, i = 1 \ldots, W$. The kernel matrices are mean-centred using

$$\mathbf{K}_{cp} = \mathbf{K}_p - \mathbf{I}_w \mathbf{K}_p - \mathbf{K}_p \mathbf{I}_w + \mathbf{I}_w \mathbf{K}_p \mathbf{I}_w, \tag{5.10}$$

$$\mathbf{K}_{cf} = \mathbf{K}_f - \mathbf{I}_w \mathbf{K}_f - \mathbf{K}_f \mathbf{I}_w + \mathbf{I}_w \mathbf{K}_f \mathbf{I}_w. \tag{5.11}$$

where $\mathbf{K}_{cp}$ and $\mathbf{K}_{cf}$ are the past and future mean-centred kernel matrices, $\mathbf{I}_w$ is an $W \times W$ matrix in which each element is equal to $1/W$.

The p and f kernel matrices are then factorised using QR decomposition in order to avoid the need for regularisation

$$\mathbf{K}_{cp} = \mathbf{Q}_p \mathbf{R}_p \quad \text{and} \quad \mathbf{K}_{cf} = \mathbf{Q}_f \mathbf{R}_f, \tag{5.12}$$

where $\mathbf{Q}_p$ and $\mathbf{Q}_f$ are orthogonal matrices and $\mathbf{R}_p$ and $\mathbf{R}_f$ are upper triangular matrices. Although, $\mathbf{K}_{cp}$ and $\mathbf{K}_{cf}$ are not full rank, they can be managed by using the MATLAB backlash operator which makes them equivalent to pseudo-inverses. The product of the orthogonal matrix pair is then calculated and canonical variates are determined using singular value decomposition (SVD):

$$\mathbf{H} = \mathbf{Q}_f^T \mathbf{Q}_p = \mathbf{U} \mathbf{S} \mathbf{V}^T, \tag{5.13}$$

where $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices and $\mathbf{S}$ is a diagonal matrix. The singu-

lar values of $\mathbf{H}$ (its elements on the main diagonal) show the degree of correlation between pairs of $\mathbf{U}$ and $\mathbf{V}$. This approach avoids the computational problems associated with generating the scaled Hankel for the SVD operation when covariances are based on kernel matrices constructed from the observation data as reported by Giantomassi et al. (2014). This is a key advantage of the proposed technique.

The left and right singular vectors, $(\mathbf{U}^*)$ and $(\mathbf{V}^*)$ respectively are normalised as follows::

$$\mathbf{U}^* = \mathbf{R}_f^+ \mathbf{U} \left(W-1\right)^{\frac{1}{2}} \quad \text{and} \quad \mathbf{V}^* = \mathbf{R}_p^+ \mathbf{V} \left(W-1\right)^{\frac{1}{2}}, \tag{5.14}$$

where $\mathbf{R}_f^+$ and $\mathbf{R}_p^+$ are pseudo-inverse of $\mathbf{R}_f$ and $\mathbf{R}_p$ respectively. Re-arranging the columns of the eigenvectors according to the sizes of their corresponding eigenvalues in descending order makes $\mathbf{V}_n^*$ (i.e. the first n columns of $\mathbf{V}^*$) the ones having the most dominant pairwise correlations with those of $\mathbf{U}^*$. The transformation matrices for determining the n-dimensional state variables and residuals are given by

$$\mathbf{J} = \mathbf{V}_n^* \in \Re^{W \times n} \quad \text{and} \quad \mathbf{L} = \left(I - \mathbf{J}\mathbf{J}^T\right) \in \Re^{W \times W}, \tag{5.15}$$

The state and residual spaces are computed as shown below:

$$\mathbf{Z}^* = \mathbf{J} \cdot \mathbf{K}_{cp} \in \Re^{n \times W}. \quad \text{and} \quad \mathbf{E}^* = \mathbf{L} \cdot \mathbf{K}_{cp} \in \Re^{W \times W}. \tag{5.16}$$

5.4 Fault detection using KCVA-QRD

Similar to other multivariate statistical process monitoring methodologies, the fault detection strategy of KCVA-QRD involves two phases: off-line training and on-line monitoring or testing. The off-line training phase involves developing the process model, and calculating the monitoring indices and their control limits using the normal operation data. Conversely, on-line monitoring involves computing the monitoring indices using faulty or test data and comparing their values with the control limits obtained in the off-line training phase to determine the status of the process. The Hotellings T^2 and the Q statistic or squared prediction error (SPE) are used as

the indices for KCVA-QRD-based process monitoring. The Hotellings T^2 monitors the changes in the state space while the Q statistic monitors the changes in the residual space. The monitoring indices are computed using

$$T^2 = \sum_{i=1}^{n} z_{i,l}^{*2} \quad \text{and} \quad Q = \sum_{i=1}^{W} e_{i,l}^{*2}, \tag{5.17}$$

where n is the number of states retained, $z_{i,l}$ and $e_{i,l}$ are $(i,l)^{\text{th}}$ the entries of $\mathbf{Z}^*$ and $\mathbf{E}^*$ matrices respectively. To correct the assumption of Gaussian distributed variables, the KDE method is also adopted for calculating the control limits from the PDFs of T^2 and Q as discussed in Section 3.1.3.

For a test observation vector $\mathbf{x}_l$ at a given time point l, the past vector $\mathbf{x}_{p,l}^{(t)}$ is computed similar to (5.4) and (5.5). The states and residuals of the test data are computed as

$$\mathbf{z}_l^{*(t)} = \mathbf{J} \cdot \mathbf{k}_{cp}^{(t)} \in \Re^{n \times 1} \quad \text{and} \quad \mathbf{e}_l^{*(t)} = \mathbf{L} \cdot \mathbf{k}_{cp}^{(t)} \in \Re^{W \times 1}, \tag{5.18}$$

where $\mathbf{k}_{cp}^{(t)}$ is the kerenel matrix of the test observation vector.

5.5 Summary of KCVA-QRD-based fault detection procedure

The steps involved in the proposed kernel CVA technique for the training and monitoring phases are outlined below:

Off-line training

c–1 Obtain observation vector.

c–2 Expand observation vector at each time point t to obtain information from the past (p) and future (f) measurements.

c–3 Form kernel matrices of the past and future measurements.

c–4 Mean-centre the past and future kernel matrices. Factorise the mean-centred past and future kernel matrices using QR decomposition to obtain pairs of upper triangular and orthogonal matrices.

c–5 Compute the product of the orthogonal matrix pair from step 4 and perform singular value decomposition. Normalise the canonical coefficients.

c–6 Determine states and residuals.

c–7 Compute monitoring indices T^2 and Q at each time point as the sum of the squared state variables and residuals respectively.

On-line monitoring

d–1 Acquire test data and define past and future matrices and arrange data similar to training data.

d–2 Form kernel matrices of the past and future measurements using the same function and parameters used in the training stage and mean-centre.

d–3 Calculate states and residuals of test data.

d–4 Compute T^2 and Q of test data.

d–5 Monitor process by comparing value of T^2 and Q against their control limits. A fault is detected if both monitoring indices exceed their control limits.

5.6 Application study

The kernel CVA based on QR decomposition (KCVA-QRD) and kernel CVA on regularised kernel data (KCVA-REG) were tested on the TE process to assess their performance. Three faults (3, 9, and 15) of the TE process were considered for this application study. Faults 3 and 9 are step change and random variation in D feed

temperature respectively, while Fault 15 is a sticking valve problem involving the condenser cooling water valve. These three faults are usually more difficult to detect in the TE process because they cause very little variation in the variables that are measured.

5.6.1 Results and discussion

The performance of the techniques were assessed based on the three metrics used in Chapter 3 (FDR, FAR and detection delay): fault detection rate, fault detection time and false alarm rate. Fault detection was calculated as the percentage of observations with a value higher than the value of the control limit in the fault region of the data signal. Detection rate is the difference between when a fault was introduced and when it was detected. False alarm rates were calculated as the percentage of observations having values greater than the value of the control limits outside the fault region. An overview of the process including number of observation, types of variables and the types of faults are as explained in Chapter 3 where a schematic of the process is also presented.

Table 5.1 shows the rate of fault detection for Faults 3, 9 and 15 based on the KCVA-QRD approach and results obtained with KCVA-REG at different values of regularisation. The detection rates at a regularisation value of 10^{-2} were the lowest (51.25, 78.13, and 85.38 percent) for Faults 3, 9 and 15 respectively. At a very small regularisation value of 10^{-8}, the detection rates improved but the values were still lower than the results obtained via QR decomposition.

The detection times for the KCVA-QRD approach for all three faults was 15 minutes while the corresponding rates for the KCVA-REG technique were 54/45, 84/66, and 54/45 minutes for Faults 3, 9 and 15 respectively for the worst and best detection time delays. A higher detection time means that it takes a longer time to detect a fault that has occurred which is not desired in process monitoring. Thus, in all faults considered, the KCVA-QRD based detection times were better than the best detection times obtained via the KCVA-REG approach. The regularisation

approach also had higher FARs (which again makes it relatively poorer) than the KCVA-QRD approach except for the smallest regularisation value.

Table 5.1: Comparison of monitoring results (DKPCA, KCVA-QRD and KCVA-REG). The DKPCA results are very poor in all faults while the performance of KCVA-REG depends on the regularisation parameter value. The proposed KCVA-QRD is generally better without regularisation.

FDR (%)					
Fault	DKPCA	KCVA-QRD	KCVA-REG		
			10^{-2}	10^{-5}	10^{-8}
3	9.50	98.25	51.25	98.13	98.13
9	9.13	97.50	78.13	97.38	97.38
15	11.50	98.25	85.35	98.13	98.13
FAR (%)					
	DKPCA	KCVA-QRD	KCVA-REG		
			10^{-2}	10^{-5}	10^{-8}
3	0.0316	0.0382	0.0458	0.0840	0.0076
9	0.0316	0.0382	0.0458	0.0840	0
15	0.0316	0.0382	0.0458	0.0840	0
Detection delay (min)					
	DKPCA	KCVA-QRD	KCVA-REG		
			10^{-2}	10^{-5}	10^{-8}
3	23	15	54	45	45
9	23	15	84	63	66
15	23	15	54	45	45

Fig. 5.1 shows the monitoring results for Fault 15. Figs. 5.1(a) represents the monitoring result for KCVA-QRD while Figs. 5.1(b) and 5.1(c) are the monitoring results for KCVA-REG at regularisation values of 10^{-2} and 10^{-8} respectively. The solid signals represent the monitoring indices while the dash-dot horizontal lines are the control limits. Unlike the other two figures, the monitoring index did not fully go above the control limit most of the time in Fig. 5.1(b). This shows that its fault detection performance is poor. The reason for this poor detection performance is the inappropriate value of the regularisation parameter used. This heavy dependence on regularisation is a major weakness of the KCVA-REG technique.

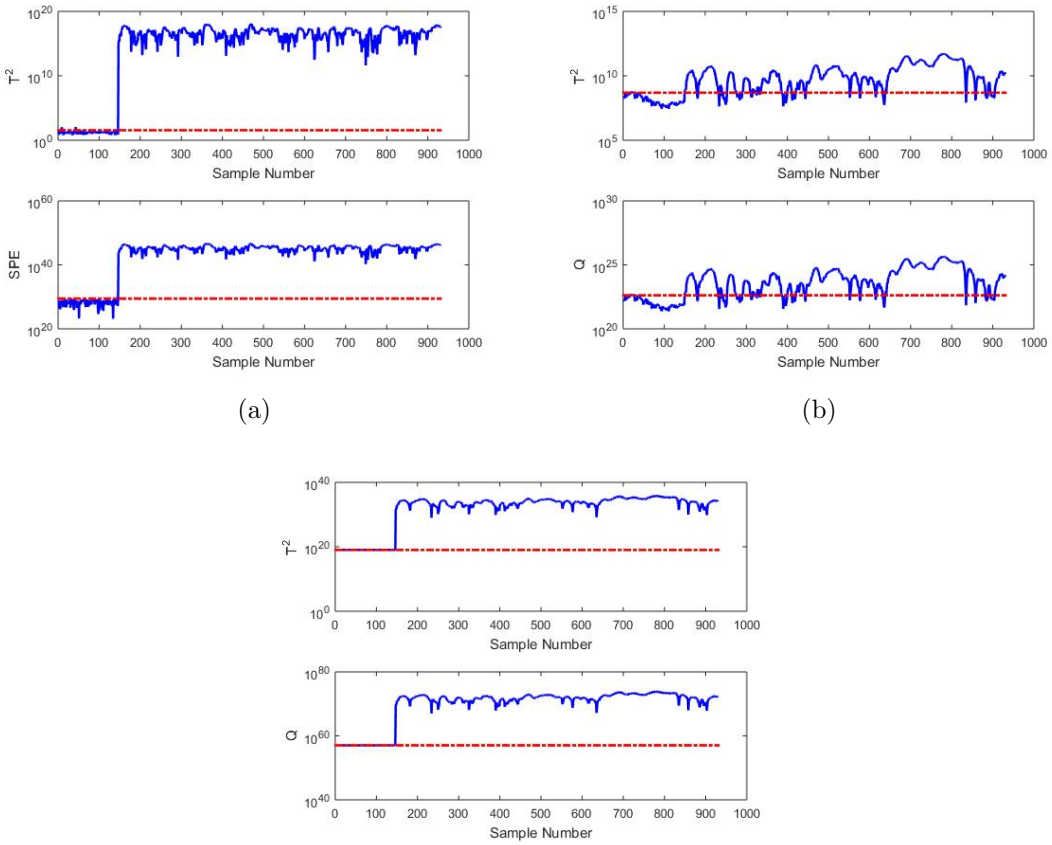


Figure 5.1: Monitoring statistics of Fault 15. (a) KCVA with QRD, (b) KCVA with regularisation (10^{-2}), (c) KCVA with regularisation (10^{-8}).

5.7 Concluding remarks

In this chapter, the KCVA-QRD technique is proposed to account for nonlinearity and process dynamics simultaneously by combining the CVA and the kernel paradigm. In the proposed technique, the product matrix computed from the past and future kernel matrices which is normally subjected to SVD was computed using QR decomposition. The aim of this implementation strategy is to avoid singularity problems encountered in kernel-based canonical modelling.

Results of the application study of the proposed KCVA-QRD approach using the TE process were compared with DKPCA and KCVA-REG-based results. The study shows that the proposed KCVA-QRD technique performed better than the both DKPCA and KCVA-REG in both monitoring rate and the time taken to detect

faults. This supports the effectiveness of the proposed approach. Furthermore, the technique proposed does not require selecting an optimum regularisation value to perform kernel-based CVA. This is desirable because choosing the wrong regularisation parameter value generates poor monitoring results. A second nonlinear dynamic/kernelisation technique, the kernel latent variable-CVA) will be the subject of the next chapter.

Chapter 6

Kernel latent variable CVA for nonlinear dynamic process monitoring

This chapter focuses on the development of kernel latent variable canonical variate analysis (KLV-CVA) for nonlinear dynamic process monitoring. The KLV-CVA algorithm is the second of two strategies proposed in this thesis to: (i) effectively monitor nonlinear dynamic processes, and (ii) address the problem of ill-conditioned kernel matrices and regularisation encountered if CVA is performed directly on symmetric kernel matrices constructed from observation data. The first approach based on kernel CVA with QR decomposition was discussed in Chapter 5. Details of the KLV-CVA approach and its mathematical formulation are presented in this chapter. In the application study, the proposed KLV-CVA approach is tested on Faults 3, 9 and 15 of the TE process which are the most difficult to detect. The results obtained are discussed and compared with results obtained from two related techniques.

6.1 Introduction

The popular procedure for developing kernel versions of linear algorithms (kernelisation) described in the open literature is the two-step approach of mapping data implicitly from the input space to a feature space using a kernel function, and applying the original algorithm on the corresponding feature space (Honeine and Richard, 2011b; Schölkopf et al., 1998). Figure 6.1 summarises this process.

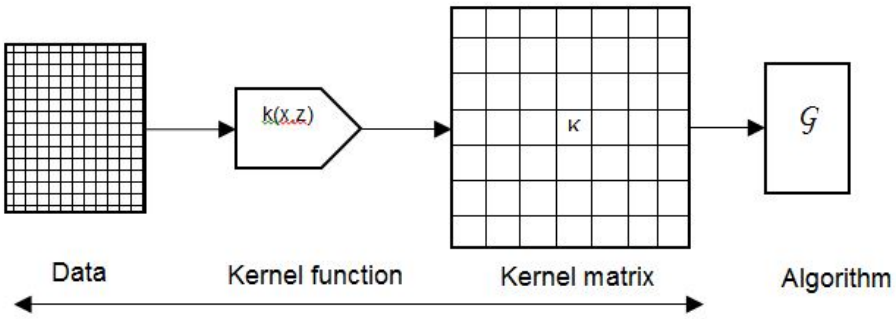


Figure 6.1: Flowchart showing the regular kernelisation process. In general, it involves constructing a kernel matrix using a kernel function and performing the required algorithm directly on the kernel matrix.

This implies taking the kernel matrix as the input of the given algorithm instead of the observation data. This amounts to modifying the input interface of the algorithm or inner processing in its kernel version. This view of kernelisation suggests that algorithms that can be performed in terms of inner products can be made nonlinear by kernel substitution - i.e. the kernel trick (Schölkopf and Smola, 2002; Schölkopf et al., 1999).

Assuming that $\mathbf{X}$ is the input data matrix with each row as data points and each column as variables. The output $\mathbf{Y}$ of an algorithm $\mathcal{G}$ is obtained by taking the inner product of the observation data as input. This can be represented as follows:

$$\mathbf{Y} = \mathcal{G}(\mathbf{X}^T\mathbf{X}). \tag{6.1}$$

On the contrary, the kernelised algorithm $\mathcal{G}_k$ takes the inner product matrix (or kernel matrix) $\mathbf{K}$ in the feature space as input instead of the original inner product matrix of (6.1). Consequently, the relationship between the kernelised algorithm $\mathcal{G}_k$

and the original algorithm $\mathcal{G}$ can be represented as

$$\mathbf{Y}_k = \mathcal{G}_k(\mathbf{X}) = \mathcal{G}(\mathbf{K}). \tag{6.2}$$

where $\mathbf{Y}_k$ is the output of $\mathcal{G}_k$.

Alternatively, Yang et al. (2004) performed kernel fisher discriminant analysis (FDA) by combining KPCA and linear discriminant analysis (LDA). Lee et al. (2007) also carried out kernel independent component analysis (KICA) by integrating KPCA and ICA. In line with the works of Yang et al. and Lee et al., Chen and Zhang (2007) suggested that kernel methods can essentially be viewed as performing kernel PCA on the input data to extract nonlinear principal components (PCs) in the feature space, and then performing the original algorithm, taking the PCs as input. Figure 6.2 shows the computing process of this stance.

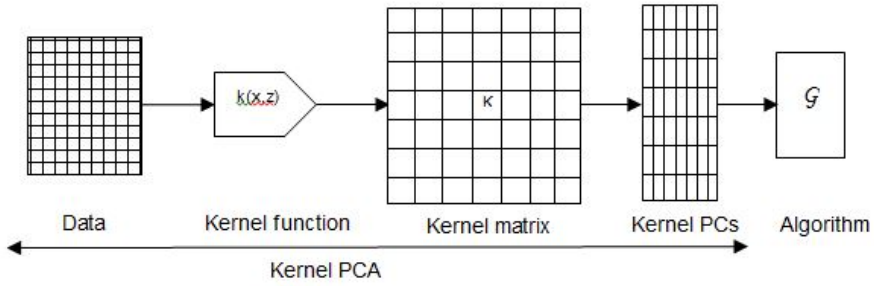


Figure 6.2: Flowchart showing an alternate kernelisation approach. It is essentially carrying out KPCA followed by performing any other specified algorithm on the kernel latent variables

Chen and Zhang hold the view that the process of kernelising an algorithm $\mathcal{G}$ and the process of performing KPCA ($\mathcal{G}_{\text{KPCA}}$) to obtain PCs followed by carrying out the algorithm $\mathcal{G}$ on the PCs (i.e. $\mathcal{G}_{\text{KPCA}} + \mathcal{G}$) are equivalent. An expression of the process is represented as

$$\mathbf{F} = \mathcal{G}_{\text{KPCA}}(\mathbf{X}), \tag{6.3}$$

$$\mathbf{Y}_{k,f} = \mathcal{G}(\mathbf{F}^T\mathbf{F}), \tag{6.4}$$

where $\mathbf{F}$ are the PCs in the feature space, and $\mathbf{Y}_{k,f}$ denotes output of the algorithm

in the feature space. Then, PCA and KPCA can be related as

$$\mathbf{F} = \mathcal{G}_{\mathrm{KPCA}}(\mathbf{X}) = \mathcal{G}_{\mathrm{PCA}}(\mathbf{K}). \tag{6.5}$$

Notice (6.2) and (6.4) give the relation

$$\mathbf{Y}_{k,f} = \mathcal{G}(\mathbf{F}^T\mathbf{F}) = \mathcal{G}(\mathbf{K}) = \mathbf{Y}_k. \tag{6.6}$$

Equation (6.6) proves that the kernelisation of $\mathcal{G}$ to obtain $\mathcal{G}_k$ and the process of $\mathcal{G}_{\mathrm{KPCA}} + \mathcal{G}$ are equivalent. However, the latter kernelisation process requires only changing the input of the original algorithm. It does not require changing the original algorithm by formulating it in inner product format which could be quite problematic for complex algorithms.

Inspired by the works of Yang et al. (2004), Lee et al. (2007), Chen and Zhang (2007), and the idea of using dynamic latent variable models for extracting dynamics in a variable space into autocorrelated latent factors discussed by Li et al. (2011), a new kernel latent variable-CVA (KLV-CVA) technique is proposed for nonlinear dynamic process monitoring. The proposed algorithm involves performing CVA on the kernel component space obtained from KPCA instead of performing the CVA directly on the kernel matrix constructed from process measurements. This technique transforms the higher dimensional inputs into lower dimensions while retaining the important characteristics of the inputs. It also simplifies the algorithm and enhances its generalisation ability. Furthermore, the singularity problem associated with the KCCA approach is avoided. The proposed algorithm is derived and tested on the TE benchmark process in this chapter.

6.2 Kernel latent variable CVA

Suppose $\mathbf{x_k} \in \Re^m$ is a set of training data, where $k = 1, \ldots, N$ are the number of observations while m are variables. Using the derivation presented in Section 3.1.1,

the $N \times N$ centred symmetric kernel matrix, $\mathbf{K}_c$ has N eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_N$ and N orthogonal eigenvectors, $\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, \ldots, \boldsymbol{\alpha}^N$ satisfying

$$\left\langle \boldsymbol{\alpha}^i, \boldsymbol{\alpha}^j \right\rangle = \delta_{i,j}, (i, j = 1, 2, \ldots, N)\,. \tag{6.7}$$

Let $\mathbf{G} \in \Re^{N \times r}$ contain eigenvectors corresponding to the largest r eigenvalues. A kernel latent variable matrix $\mathbf{Z}$ is defined as

$$\mathbf{Z} = \mathbf{G}^T \mathbf{K}_c \in \Re^{r \times N}. \tag{6.8}$$

To capture both nonlinear and dynamic relations, CVA is performed on $\mathbf{Z}$, treating it as a collected $r \times N$ data. State variables and residuals are then extracted for computing the monitoring statistics.

Assuming that $\mathbf{z}_k$ is the k^{th} column vector of $\mathbf{Z}$, information from the past (p) and future (f) data series are defined as

$$\mathbf{z}_{p,k} = \begin{bmatrix} \mathbf{z}_{k-1} \\ \mathbf{z}_{k-2} \\ \vdots \\ \mathbf{z}_{k-p} \end{bmatrix} \in \Re^{rp} \text{ and } \mathbf{z}_{f,k} = \begin{bmatrix} \mathbf{z}_{k} \\ \mathbf{z}_{k+1} \\ \vdots \\ \mathbf{z}_{k+f-1} \end{bmatrix} \in \Re^{rf}. \tag{6.9}$$

Each component is then normalised to have a mean of 0 as follows:

$$\hat{\mathbf{z}}_{p,k} = \mathbf{z}_{p,k} - \bar{\mathbf{z}}_{p,k} \quad \text{and} \quad \hat{\mathbf{z}}_{f,k} = \mathbf{z}_{f,k} - \bar{\mathbf{z}}_{f,k}, \tag{6.10}$$

where $\bar{\mathbf{z}}_{p,k}$ and $\bar{\mathbf{z}}_{f,k}$ denote the sample means of $\mathbf{z}_{p,k}$ and $\mathbf{z}_{f,k}$ respectively. To obtain the past and and future matrices, $\mathbf{A}_p$ and $\mathbf{A}_f$ respectively, the corresponding past and future vectors are arranged together in columns

$$\mathbf{A}_p = [\hat{\mathbf{z}}_{p,p+1},\ \hat{\mathbf{z}}_{p,p+2}, \ldots \hat{\mathbf{z}}_{p,p+B}] \in \Re^{rp \times W}, \tag{6.11}$$

$$\mathbf{A}_f = [\hat{\mathbf{z}}_{f,p+1},\ \hat{\mathbf{z}}_{f,p+2}, \ldots \hat{\mathbf{z}}_{f,p+B}] \in \Re^{rf \times W}, \tag{6.12}$$

where the columns of the truncated Hankel matrices for N observations is given by $W = N - f - p + 1$. The sample covariances and cross-covariances of the past and future matrices are estimated as shown below:

$$\mathbf{\Sigma}_{pp} = \frac{1}{W-1}\mathbf{A}_p\mathbf{A}_p^T, \tag{6.13}$$

$$\mathbf{\Sigma}_{ff} = \frac{1}{W-1}\mathbf{A}_f\mathbf{A}_f^T, \tag{6.14}$$

$$\mathbf{\Sigma}_{fp} = \frac{1}{W-1}\mathbf{A}_f\mathbf{A}_p^T. \tag{6.15}$$

In order for $\mathbf{\Sigma}_{pp}$ and $\mathbf{\Sigma}_{ff}$, not to be singular, parameters r, p and f have to satisfy $\{rp, rf\} < N - p - f + 1$. Canonical variates are then computed via SVD of the scaled Hankel matrix, $\mathbf{H}_k$

$$\mathbf{H} = \mathbf{\Sigma}_{ff}^{-1/2}\mathbf{\Sigma}_{fp}\mathbf{\Sigma}_{pp}^{-1/2} = \mathbf{U}\mathbf{L}\mathbf{V}^T. \tag{6.16}$$

Rearranging the singular values in descending order and reordering the columns of the associated singular vectors makes the first q columns of $\mathbf{V}$ the top dominant correlations with those of $\mathbf{U}$. This generates a new matrix $\mathbf{V}_q$ of a smaller dimension such that $(q < rp)$.

Transformation matrices $\mathbf{C}$ and $\mathbf{D}$ for converting the rp-dimensional past matrices to the q-dimensional state variables and the residuals respectively are calculated as

$$\mathbf{C} = \mathbf{V}_q^T\Sigma_{pp}^{-1/2} \in \Re^{q\times rp}, \tag{6.17}$$

$$\mathbf{D} = \left(I - \mathbf{V}_q\mathbf{V}_q^T\right)\Sigma_{pp}^{-1/2} \in \Re^{rp\times rp}. \tag{6.18}$$

The canonical state variables denotes linear combinations of the past data series that most explain the future variability. The state variables $\mathbf{Z}^*$ and residuals $\mathbf{E}^*$ for training data are computed using

$$\mathbf{Z}^* = \mathbf{C}\cdot\mathbf{A}_p \in \Re^{q\times W} \quad \text{and} \quad \mathbf{E}^* = \mathbf{D}\cdot\mathbf{A}_p \in \Re^{rp\times W}. \tag{6.19}$$

6.3 KLV-CVA-based fault detection

Like CVA, the Hotellings T^2 and the Q statistic or squared prediction error (SPE) and their control limits are employed for process monitoring in KLV-CVA. Hotellings T^2 is used for monitoring changes in the model space while Q statistic is used for monitoring changes in the residual space. Their values are calculated using

$$T_k^2 = \sum_{i=1}^{q} \mathbf{z}_{i,k}^{*2} \quad \text{and} \quad Q_k = \sum_{i=1}^{rp} \mathbf{e}_{i,k}^{*2} \tag{6.20}$$

where q is the number of states retained, $z_{i,k}^*$ and $e_{i,k}$ are $(i,k)^{\text{th}}$ elements of $\mathbf{Z}^*$ and $\mathbf{E}^*$ matrices respectively. Since measurements in a nonlinear process do not follow the Gaussian distribution, the control limits of T^2 and Q are determined using the KDE approach.

To carry out online monitoring, the test observation vector $\mathbf{x}_t$ is preprocessed and arranged similar to the training data and latent variables are computed. The past observation vector $\hat{\mathbf{z}}_{p,k}^{(t)}$ is then computed similar to (6.9) and (6.10). The state variables and residuals for test data are computed as

$$\mathbf{z}_k^* = \mathbf{C}.\hat{\mathbf{z}}_{p,k}^{(t)} \in \Re^{q\times 1} \quad \text{and} \quad \mathbf{e}_k^* = \mathbf{D}.\hat{\mathbf{z}}_{p,k}^{(t)} \in \Re^{rp\times 1}. \tag{6.21}$$

6.3.1 Summary of KLV-CVA fault detection procedure

Off-line training

a–1 Acquire normal operating data, construct kernel matrix and obtain kernel latent variables.

a–2 Compute past and future data series from the kernel latent variables using (6.11) and (6.12) .

a–3 Compute covariances and cross-covariance from (6.13) to (6.15).

a–4 Perform SVD on scaled Hankel matrix using (6.16) and determine number of states to retain.

a–5 Determine state variables and residuals using (6.19).

a–6 Compute monitoring indices using (6.20) and their control limits using (**??**).

On-line monitoring

b–1 Acquire test data and construct kernel matrix.

b–2 Calculate state and residual of test data using (6.21).

b–3 Calculate T^2 and Q of test data and monitor process by comparing their values against their control limits. A fault is detected if either of the monitoring indices or both exceed their control limits.

6.4 Application study

Application of the KLV-CVA technique on the TE process, details of the implementation procedure as well as selection of design parameters are presented in this section. The results obtained are also presented and discussed.

6.4.1 Implementation details

Normal operation data were used as the training set and data collected under fault conditions were used for monitoring. The radial basis function (RBF) kernel, a recommended representative kernel function for KPCA-based studies (Lee et al., 2004) was adopted in this work. All confidence limits were determined from the probability density functions of the monitoring indices using KDE approach. Control charts based on the T^2 and and SPE were constructed for process monitoring at 99% confidence level. Monitoring performance indices used are fault detection rate (FDR), false alarm rate (FAR) and detection time.

6.4.2 Parameter selection

A kernel window width of 1660 was used in this study based on cross-validation with the training data. The dimension of the kernel component space was set based on cumulative variance percentage (CVP) (Deng et al., 2013). The number of kernel latent variables whose cumulative variance accounted for more than 99% of the total variance was adopted to ensure little loss of information. This sets the kernel latent variables used to 62.

To successfully develop the CVA and kernel CVA models to characterize the variability of the off-line data requires that the number of time lags for the past and future measurements are determined. The lag order represents the number of past measurements that are significantly correlated with a measurement at a given time point. A lag order of 15 (i.e. $p = f = 15$) was used based on the autocorrelation function analysis discussed in chapter 4. This satisfies $rp = rf = 900 < N - p - f + 1 = 931$. A tabulation of the design parameters is presented in Table 6.1.

Table 6.1: Summary of design parameters

Parameter	ID	Value
Kernel window width	c	1660
Regularisation parameter	ra	$10^{-2}, 10^{-5}$ & 10^{-8}
No. of states retained	a	16
Length of past and future lag	p,f	15
Kernel latent variable	KLV	62

6.4.3 Results and discussion

Monitoring results showing the FDRs and FARs of all three techniques investigated for Faults 3, 9 and 15 are presented in Table 6.2. The results show that KLV-CVA has higher FDRs than CVA-KDE. Its FDRs are 98.13, 97.25, and 98.13 percent for Faults 3, 9, and 15 respectively as against 75.25, 91.25, and 97.88 percent recorded by CVA-KDE. The KCVA-REG fault detection at different regularisation sizes show that its performance is affected by the size of regularisation used. At a regularisation

value of 10^{-2}, it recorded FDRs of 51.25, 78.18 and 85.35 percents for Faults 3, 9 and 15 respectively. These results are poorer than results obtained using CVA-KDE and KLV-CVA. The KCVA-REG FDR results became comparable with KLV-CVA only at highr regularisation values. Also, both CVA-KDE and KLV-CVA recorded 0 false alarms while the KCVA-REG approach gave false alarms.

Furthermore, KLV-CVA also gave better results in terms of detection times (i.e. 45, 66 and 45 minutes) compared to 54, 72 and 51 minutes recorded by CVA-KDE for the three faults (Table 6.3). Conversely, KCVA-REG detection delay results were poorer or equal to the CVA-KDE results at a regularisation value of 10^{-2} but were better than CVA-KDE and comparable with the KLV-CVA approach at higher regularisation values. It is worthy of note that although the KCVA-REG approach gave results comparable in FDR and detection delay at higher regularisation values, its high false alarm rate still makes it less effective than the proposed KLV-CVA approach.

Table 6.2: Comparison of FDR and FAR. KLV-CVA performs better than CVA-KDE. KCVA-REG values depend on regularisation value used and compares with KLV-CVA values only at high regularisation.

FDR (%)					
Fault	CVA-KDE	KLV-CVA	KCVA-REG		
			10^{-2}	10^{-5}	10^{-8}
3	75.25	98.13	51.25	98.13	98.13
9	91.25	97.25	78.13	97.38	97.38
15	97.88	98.13	85.35	98.13	98.13
FAR (%)					
Fault	CVA-KDE	KLV-CVA	KCVA-REG		
			10^{-2}	10^{-5}	10^{-8}
3	0	0	0.0458	0.0840	0.0076
9	0	0	0.0458	0.0840	0
15	0	0	0.0458	0.0840	0

The superior performance of KLV-CVA over CVA-KDE can be clearly observed in Figure 6.3 showing their monitoring charts for Fault 3. It can be seen that in Figure 6.3(a), the CVA-based T^2 represented by the solid signal is below the horizontal dash-dot line (i.e., the control limit), virtually all through the duration

Table 6.3: Comparison of fault detection delay. KLV-KCVA detected faults earlier than CVA-KDE. KCVA-REG values depend on regularisation parameter used.

Fault	CVA-KDE	KLV-CVA	KCVA-REG		
			10^{-2}	10^{-5}	10^{-8}
3	54	45	54	45	45
9	72	66	84	63	66
15	51	45	54	45	45

of the process. This suggests that the process is in-control. However, the Q statistic is above the control limit at some points showing the presence of a fault. The presence of a malfunction is confirmed by both the T^2 and the Q statistic based on KLV-CVA. The monitoring indices of KLV-CVA (Figure 6.3(b)) are above the control limits prominently, indicating the presence of a fault. Furthermore, the poor performance of KCVA-REG is clearly seen when a regularisation size of 10^{-2} was used (Figure 6.3(c)) compared to Figure 6.3(d) which is comparable to the KLV-CVA result but with a higher false alarm rate as mentioned earlier. This affirms that the performance of the KCVA-REG approach is affected by the size of the regularisation parameter selected. Selecting a wrong regularisation parameter makes it perform dismally, while the proposed KLV-CVA approach does not require the use of a regularisation parameter.

6.5 Concluding remarks

In this chapter, the CVA with KDE technique was generalised to a non-linear case to give a novel kernel latent variable-based CVA approach (KLV-CVA). The proposed technique draws from the synergy of combining kernel KPCA and CVA. The kernel PCA accounts for non-linearities while the CVA captures the process dynamics. Applying the proposed technique to the TE challenge problem and comparing the result with those of CVA with KDE, and kernel CVA with regularisation, showed that the proposed method recorded better overall fault detection performance. Furthermore, the strategy of using kernel latent variables as input to the CVA made

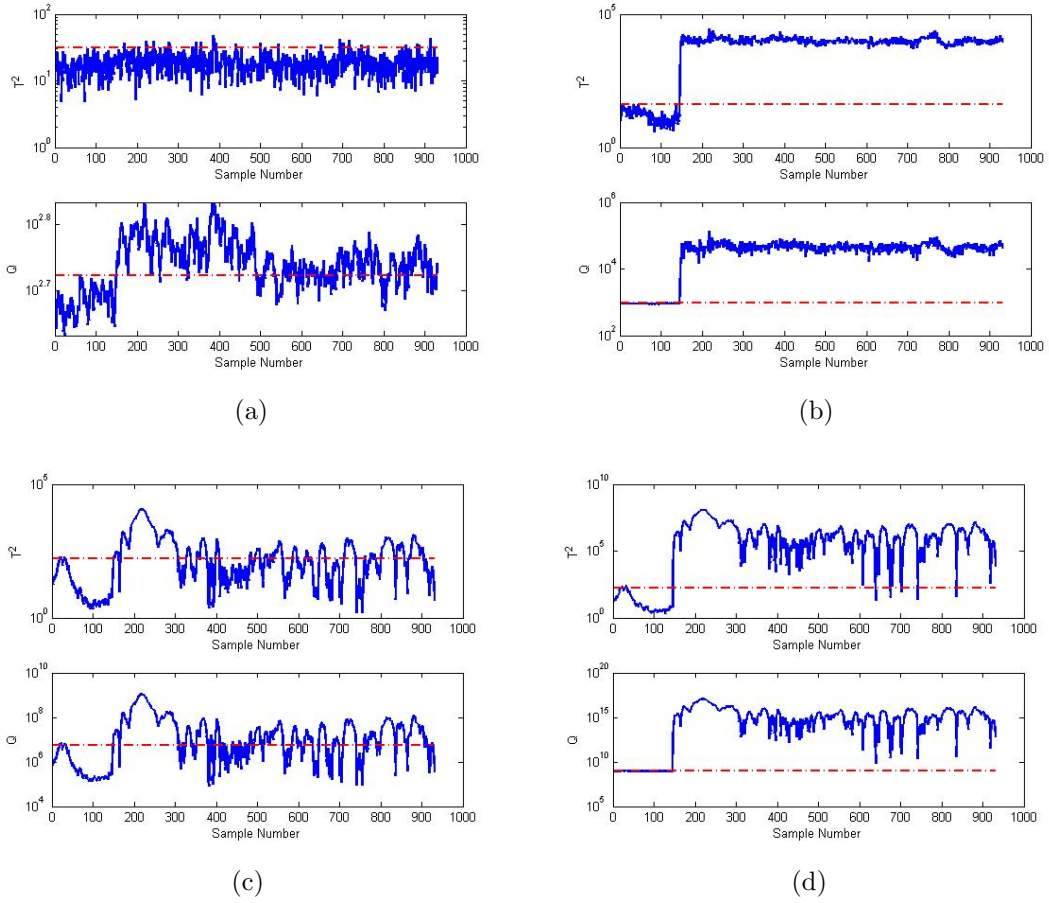


Figure 6.3: Monitoring statistics for Fault 3 using KLV-CVA and KCVA-REG using 2 regularisation sizes (a) CVA-KDE, (b) KLV-CVA, (c) 10^{-2} regularisation, (d) 10^{-8} regularisation

.

the implementation of the proposed technique more convenient. Unlike KCVA-REG, KLV-CVA does not need regularisation to perform well. In addition to nonlinearity and dynamics, another common feature in industrial systems is the use of multiple operating modes. This will is addressed in Chapter 7.

References

Akaike, H. (1975). Markovian Representation of Stochastic Processes by Canonical Variables, *SIAM J. Control* **13**(1): 162–173.

Aronszajn, N. (1950). Theory of Reproducing Kernels, *Trans. Am. Math. Soc.* **68**(3): 337–404.

Berg, C., Christensen, J. P. R. and Ressel, P. (1984). *Harmonic analysis on semigroups: theory of positive definite and related functions*, Vol. 53, Springer-Verlag, New York, NY, USA.

Bergman, S. (1950). *The Kernel Function And Conformal Mapping*, The American Mathematical Society, New York.

Berlinet, A. and Thomas-Agnan, C. (2004). *Reproducing Kernel Hilbert Spaces in Probability and Statistics*, Springer Science Business Media, New York.

Bowman, A. W. and Azzalini, A. (1997). *Applied smoothing techniques for data analysis*, Clarendon Press, Oxford.

Boyd, S. and Vandenberghe, L. (2004). *Convex optimization*, Cambridge University Press, Cambridge.

Campos-Delgado, D. U. and Espinoza-Trejo, D. R. (2011). An observer-based diagnosis scheme for single and simultaneous open-switch faults in induction motor drives, *IEEE Trans. Ind. Electron.* **58**(2): 671–679.

Camps-Valls, G., Rojo-Alvarez, J. L. and Martinez-Ramon, M. (2006). *Kernel meth-*

ods in bioengineering, signal and image processing, Idear Group Publishing, London.

Chen, J. and Liu, K.-c. (2002). On-line batch process monitoring using dynamic PCA and dynamic PLS models, *Chem. Eng. Sci.* **57**: 63–75.

Chen, Q., Wynne, R., Goulding, P. and Sandoz, D. (2000). The application of principal component analysis and kernel density estimation to enhance process monitoring, *Control Eng. Pract.* **8**(5): 531–543.

Chen, R. Q. (2013). Advances in data-driven monitoring methods for complex process, *3rd Int. Conf. Appl. Mech. Mater. Manuf. ICAMMM 2013;* **423-426**: 2448–2451.

Chen, W. and Zhang, H. (2007). The Condition of kernelizing an algorithm and an equivalence between kernel methods, *in* J. Marti, J. Benedi, A. Mendonca and J. Serrat (eds), *Third Iber. Conf. Pattern Recognit. Image Anal. Part I*, Girona, Spain, pp. 338–345.

Chen, Z., Zhang, K., Hao, H., Ding, S. X., Krueger, M. and He, Z. (2014). A canonical variate analysis based process monitoring scheme and benchmark study, *19th World Congr. Int. Fed. Autom. Control. Cape Town, South Africa*, pp. 10634–10639.

Cheng, C. and Chiu, M.-S. (2005). Nonlinear process monitoring using JITL-PCA, *Chemom. Intell. Lab. Syst.* **76**(1): 1–13.

Chiang, L. H., Russell, E. L. and Braatz, R. D. (2001). *Fault detection and diagnosis in industrial systems*, Springer-Verlag, London.

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, *Chem. Eng. Sci.* **60**(1): 279–288.

Choi, S. W., Lee, C., Lee, J. M., Park, J. H. and Lee, I. B. (2005). Fault detection and identification of nonlinear processes based on kernel PCA, *Chemom. Intell. Lab. Syst.* **75**(1): 55–67.

Choi, S. W. and Lee, I.-B. (2004). Nonlinear dynamic process monitoring based on dynamic kernel PCA, *Chem. Eng. Sci.* **59**(24): 5897–5908.

Chu, D., Liao, L.-Z., Ng, Michael, K. and Zhang, X. (2013). Sparse kernel canonical correlation analysis, *Proc. Int. MultiConference Eng. Comput. Sci. IMECS*, Vol. I, Hong Kong, pp. 1–6.

Dai, X. and Gao, Z. (2013). From model, signal to knowledge : a data-driven perspective of fault detection and diagnosis, *IEEE Trans. Ind. Informatics* **9**(4): 2226–2238.

Deng, X., Tian, X. and Chen, S. (2013). Modified kernel principal component analysis based on local structure analysis and its application to nonlinear process fault diagnosis, *Chemom. Intell. Lab. Syst.* **127**: 195–209.

Ding, S. X., Yin, S., Peng, K., Hao, H. and Shen, B. (2013). A novel scheme for key performance indicator prediction and diagnosis with application to an industrial hot strip mill, *IEEE Trans. Ind. Informatics* **9**(4): 2239–2247.

Dong, D. and McAvoy, T. J. (1996). Nonlinear principal component analysis–Based on principal curves and neural networks, *Comput. Chem. Eng.* **20**(1): 65–78.

Downs, J. J. and Vogel, E. F. (1993). A plant-wide industrial process control problem, *Comput. Chem. Eng.* **17**(3): 245–255.

Duin, R. P. W., De Ridder, D. and Tax, D. M. J. (1998). Featureless pattern classification, *Kybernetika* **34**(4): 399–404.

Dunia, R. and Qin, S. J. (1998). Subspace approach to multidimensional fault identification and reconstruction, *AIChE J.* **44**(8): 1813–1831.

Frank, P. M. (1990). Fault diagnosis in dynamic systems using analytical knowledge-based redundancy A survey and some new results, *Automatica* **26**(3): 459–474.

Ge, Z. and Song, Z. (2013). *Multivariate statistical process control: process monitoring methods and applications*, Springer-Verlag, London.

Ge, Z., Song, Z. and Gao, F. (2013). Review of recent research on data-based process monitoring, *Ind. Eng. Chem. Res.* **52**: 3543–3562.

Gertler, J. (1997). Fault detection and isolation using parity relations, *Control Eng. Pract.* **5**(5): 653–661.

Gertler, J. J. (1991). Analytical redundancy methods in fault detection and isolation-survey and synthesis, *Proc. IFAC/IAMCS Symp. safe Process*, Vol. 1, Baden-Baden Germany, pp. 9–21.

Gertler, J. J. (1998). *Fault detection and diagnosis in Engineering Systems*, Marcel Dekker, New York.

Giantomassi, A., Ferracuti, F., Iarlori, S., Longhi, S., Fonti, A. and Comodi, G. (2014). Kernel canonical variate analysis based management system for monitoring and diagnosing smart homes, *Int. Jt. Conf. Neural Netw.*, Beijing, China, pp. 1432–1439.

Gonzalez, J. P. N. and Castanon, L. E. G. (2011). Fault detection and dagnosis with statistical and soft computing methods, *in* L. M. Simon (ed.), *Fault Detect. theory, methods Syst.*, Nova Science Publishers Inc., chapter Chapter 3, pp. 97–176.

Gou, Z. and Fyfe, C. (2004). A canonical correlation neural network for multicollinearity and functional data, *Neural Networks* **17**: 285–293.

Haykin, S. (1999). *Neural Networks: A comprehensive foundation*, 2nd. edn, Prentice Hall International, Inc., New Jersey.

He, T., Xie, W.-R., Wu, Q.-H. and Shi, T.-L. (2006). Process fault detection and diagnosis based on principal component analysis, *Fifth Int. Conf. Mach. Learn. Cybern.*, Dalian, pp. 3551–3556.

Hiden, H. G., Willis, M. J., Tham, M. T. and Montague, G. A. (1999). Non-linear principal components analysis using genetic programming, *Comput. Chem. Eng.* **23**: 413–425.

Hofmann, T., Schölkopf, B. and Smola, A. J. (2008). Kernel methods in machine learning, *Ann. Stat.* **36**(3): 1171–1220.

Honeine, P. and Richard, C. (2011a). Preimage problem in kernel-based machine learning, *IEEE Signal Process. Mag.* **28**(2): 77–88.

Honeine, P. and Richard, C. (2011b). Preimage problem in kernel-based machine learning [an intimate connection with the dimensionality-reduction problem], *IEEE Signal Process. Mag.* pp. 77–88.

Hotelling, H. (1936). Relation between two sets of variables, *Biometrika* **28**(3/4): 321–377.

Huang, S.-Y., Lee, M.-H. and Hsiao, C. K. (2009). Nonlinear measures of association with kernel canonical correlation analysis and applications, *J. Stat. Plan. Inference* **139**: 2162–2174.

Hwang, I., Kim, S., Kim, Y. and Eng, C. (2010). A Survey of Fault Detection, Isolation, and Reconguration Methods, *Ieee Trans. Control Syst. Technol.* **18**(3): 18.

Iri, M., Aoki, K., O'Shima, E. and Matsuyama, H. (1979). An improved algorithm for diagnosis of system failures in the chemical process, *Comput. Chem. Eng.* **3**(1-4): 489–493.

Isermann, R. (1984). Process fault detection based on modeling and estimation methods-A survey, *Automatica* **20**(4): 387–404.

Isermann, R. (1993). Fault diagnosis of machines via parameter estimation and knowledge processing-Tutorial paper, *Automatica* **29**(4): 815–835.

Isermann, R. (2005). Model-based fault-detection and diagnosis - Status and applications, *Annu. Rev. Control* **29**(1): 71–85.

Isermann, R. and Ballé, P. (1997). Trends in the application of model based fault detection and diagnosis of technical processes, *Control Eng. Pract.* **5**(5): 709–719.

Jackson, J. E. (1991). *A User's guide to principal components*, Wiley, New York, NY, USA.

Jia, F., Martina, E. B. and Morris, A. J. (2000). Non-linear principal components analysis with application to process fault detection, *Int. J. Syst. Sci.* **31**(11): 1473–1487.

Jiang, B., Huang, D., Zhu, X., Fan, Y. and Braatz, R. D. (2015). Canonical variate analysis-based contributions for fault identification, *J. Process Control* **26**: 17–25.

Jiang, B., Zhu, X., Huang, D., Paulson, J. and Braatz, R. D. (2015). A combined canonical variate analysis and Fisher discriminant analysis (CVAFDA) approach for fault diagnosis, *Comput. Chem. Eng.* **77**: 1–9.

Jing, C., Gao, X. and Zhu, X. (2014). Fault classification on Tennessee Eastman process : PCA and SVM, *Int. Conf. Mechatronics Control*, Jinzhou, China, pp. 2194–2197.

Jolliffe, I. (2002). *Principal Component Analysis*, 2nd edn, Springer-Verlag, New York, NY, USA.

Juricek, B. C., Seborg, D. E. and Larimore, W. E. (2004). Fault detection using canonical variate analysis, *Ind. Eng. Chem. Res.* **43**: 458–474.

Kano, M., Nagao, K., Hasebe, S., Hashimoto, I., Ohno, H., Strauss, R. and Bakshi, B. R. (2002). Comparison of multivariate statistical process monitoring methods with applications to the Eastman challenge problem, *Comput. Chem. Eng.* **26**(2): 161–174.

Katipamula, S. and Brambley, M. (2005). Methods for fault detection, diagnostics, and prognostics for building systemsA review, Part I, *HVAC&R Res.* **11**(1): 3–25.

Komulainen, T., Sourander, M. and Jamsa-Jounela, S.-L. (2004). An online application of dynamic PLS to a dearomatization process, *Comput. Chem. Eng.* **28**(12): 2611–2619.

Kramer, M. A. (1992). Autoassociative neural networks, *Comput. Chem. Eng.* **16**(4): 313–328.

Kruger, U., Antory, D., Hahn, J., Irwin, G. W. and McCullough, G. (2005). Introduction of a nonlinearity measure for principal component models, *Comput. Chem. Eng.* **29**(11-12 SPEC. ISS.): 2355–2362.

Kruger, U., Zhou, Y. and Irwin, G. W. (2004). Improved principal component monitoring of large-scale processes, *J. Process Control* **14**: 879–888.

Ku, W., Storer, R. H. and Georgakis, C. (1995). Disturbance detection and isolation by dynamic principal component analysis, *Chemom. Intell. Lab. Syst.* **30**(1): 179–196.

Lee, J.-M., Yoo, C., Choi, S. W., Vanrolleghem, P. A. and Lee, I.-B. (2004). Nonlinear process monitoring using kernel principal component analysis, *Chem. Eng. Sci.* **59**(1): 223–234.

Lee, J., Qin, S. and Lee, I. (2007). Fault detection of non-linear processes using kernel independent component analysis, *Can. J. Chem. Eng.* **85**(4): 526–536.

Lees, F. P. (2005). *Lees' Loss Prevention in the Process Industries: hazard Identification, Assessment and Control*, Vol. Volume 1, 3rd edn, Elsevier, Oxford.

Li, G., Liu, B., Qin, S. J. and Zhou, D. (2011). Dynamic latent variable modeling for statistical process monitoring, *18th World Congr. Int. Fed. Autom. Control*, Milano, Italy, pp. 12886–12891.

Liang, L. (2005). Multivariate statistical process monitoring using kernel density estimation, *Dev. Chem. Eng. Miner. Process.* **13**: 185–192.

Liu, Y., Xu, C. and Shi, J. (2012). Tennessee Eastman Process Monitoring Based on Support Vector Data Description, *Int. Conf. Intell. Syst. Des. Eng. Appl.*, pp. 553–555.

Lyman, P. and Georgakis, C. (1995). Plant-wide control of the Tennessee Eastman problem, *Comput. Chem. Eng.* **19**(3): 321–331.

Lyness, J. N. (1967). Numerical algorithms based on the theory of complex variable, *Proc. 1967 22nd Natl. Conf.* -, pp. 125–133.

Lyness, J. N. and Moler, C. B. (1967). Numerical Differentiation of Analytic Functions, *SIAM J. Numer. Anal.* **4**(2): 202–210.

MacGregor, J. and Cinar, A. (2012). Monitoring, fault diagnosis, fault-tolerant control and optimization: Data driven methods, *Comput. Chem. Eng.* **47**: 111–120.

Martins, J. R. R. A., Kroo, I. M. and Alonso, J. J. (2000). An automated method for sensitivity analysis using complex variables, *AIAA Pap. 2000-0689* pp. 1–12.

Martins, J. R. R. A., Sturdza, P. and Alonso, J. J. (2003). The complex-step derivative approximation, *ACM Trans. Mathmatical Softw.* **29**(3): 245–262.

Mathews, V. J. (1991). Adaptive Polynomial Filters, *IEEE Signal Process. Mag.* **8**(3): 10–26.

Maurya, M. R., Rengaswamy, R. and Venkatasubramanian, V. (2003). A systematic framework for the development and analysis of signed digraphs for chemical processes. 1. Algorithms and analysis, *Ind. Eng. Chem. Res.* **42**(20): 4789–4810.

McAvoy, T. and Ye, N. (1994). Base control for the Tennessee Eastman problem, *Comput. Chem. Eng.* **18**(5): 383–413.

Mehra, R. K. R. and Peschon, J. (1971). An innovations approach to fault detection and diagnosis in dynamic systems, *Automatica* **7**(5): 637–640.

Mercer, J. (1909). Functions of positive and negative type and their connection with the theory of integral equations, *Philos. Trans. R. Soc. London* **A 209**: 415–446.

Mika, S., Ratsch, G., Weston, J., Schölkopf, B. and Muller, K.-R. (1999). Fisher discriminant analysis with kernels, *Neural Networks Signal Process. IX*, pp. 41–48.

Mika, S., Schölkopf, B., Smola, A., Müller, K.-r., Scholz, M. and Rätsch, G. (1999). Kernel PCA and De-Noising in Feature Spaces, *Adv. Neural Inf. Process. Syst. II*, Cambridge, pp. 536–542.

Miller, P., Swanson, R. E. and Heckler, C. E. (1998). Contribution plots: A missing link in multivariate quality control, *Appl. Math. Comput. Sci.* **8**(4): 775–792.

Mohri, M., Rostamizzadeh, A. and Talwalkar, A. (2012). *Foundations of machine learning*, Vol. 1, MIT Press Cambridge, Massachusetts London, England, Cambridge.

Muller, K.-R., Mika, S., Ratsch, G., Tsuda, K. and Scholkopf, B. (2001). An introduction to kernel-based learning algorithms, *IEEE Trans. Neural Newtworks* **12**(2): 181–201.

Muradore, R. and Fiorini, P. (2012). A PLS-Based Statistical Approach for Fault Detection and Isolation of Robotic Manipulators, *IEEE Trans. Ind. Electron.* **59**(8): 3167–3175.

Negiz, A. and Cinarl, A. (1998). Monitoring of multivariable dynamic processes and sensor auditing, *J. Process Control* **8**(5-6): 375–380.

Newman, J. C., Anderson, W. K. and Whitfield, D. L. (1998). Multidisciplinary sensitivity derivatives using complex variables, *Technical report*, Computational Fluid Dynamics Laboratory.

Odiowei, P. E. P. and Cao, Y. (2010). Nonlinear dynamic process monitoring using canonical variate analysis and kernel density estimations, *IEEE Trans. Ind. Informatics* **6**(1): 36–45.

Petzold, L., Barbara, S., Tech, V., Li, S., Cao, Y. and Serban, R. (2006). Sensitivity analysis of differential-algebraic equations and partial differential equations, *Comput. Chem. Eng.* **30**(10-12): 1553–1559.

Qin, S. J. (2012). Survey on data-driven industrial process monitoring and diagnosis, *Annu. Rev. Control* **36**(2): 220–234.

Raich, A. and Cinar, A. (1996). Statistical Process Monitoring and Disturbance Diagnosis in Multivariable Continuous Processes, *AIChE J.* **42**(4): 995–1009.

Rato, T. J. and Reis, M. S. (2011). Statistical Process Control of Multivariate Systems with Autocorrelation, *21st Eur. Symp. Comput. Aided Process Eng.*, pp. 497–501.

Rato, T. J. and Reis, M. S. (2013). Advantage of Using Decorrelated Residuals in Dynamic Principal Component Analysis for Monitoring Large-Scale Systems, *Ind. Eng. Chem. Res.* **52**: 13685–13698.

Richard, C., Bermudez, J. and Honeine, P. (2009). Online Prediction of Time Series Data With Kernels, *IEEE Trans. Signal Process.* **57**(3): 1058–1067.

Rosipal, R. and Trejo, L. J. (2002). Kernel partial least squares regression in reproducing kernel hilbert space, *J. Mach. Learn. Res.* **2**: 97–123.

Rothenhagen, K. and Fuchs, F. W. (2009). Doubly fed induction generator model-based sensor fault detection and control loop reconfiguration, *IEEE Trans. Ind. Electron.* **56**(10): 4229–4238.

Ruiz-Cárcel, C., Cao, Y., Mba, D., Lao, L. and Samuel, R. T. (2015). Statistical process monitoring of a multiphase flow facility, *Control Eng. Pract.* **42**: 74–88.

Russel, E., Chiang, L. H. and Braatz, R. D. (2000). *Data-driven techniques for fault detection and diagnosis in chemical processes*, Springer, London.

Russell, E., Chiang, L. and Braatz, R. (2000). Fault detection in industrial processes using canonical variate analysis and dynamic principal component analysis, *Chemom. Intell. Lab. Syst.* **51**: 81–93.

Saitoh, S. (1988). *Theory of repoducing kernels and its applications*, Longman Scientific and Technical, Essex England.

Saxen, H., Gao, C. and Gao, Z. (2013). Data-driven time discrete models for dynamic prediction of the hot metal silicon content in the blast furnace - A review, *IEEE Trans. Ind. Informatics* **9**(4): 2213–2225.

Schölkopf, B. and Smola, A. J. (2002). *Learning with Kernels, Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press London, England.

Schölkopf, B., Smola, A. J. and Muller, K.-R. (1998). Non-linear component analysis as a kernel eigenvalue problem, *Neural Comput.* **10**: 1299–1399.

Schölkopf, B., Williamson, R., Smola, A., Shawe-Taylor, J. and Platt, J. (1999). Support Vector Method for Novelty Detection, *Adv. Neural Inf. Process. Syst. 12* pp. 582–588.

Shawe-Taylor, J. and Cristianini, N. (2004). *Kernel Methods for Pattern Analysis*, Cambridge University Press, Cambridge.

Squire, W. and Trapp, G. (1998). Using Complex Variables to Estimate Derivatives of Real Functions, *SIAM Rev.*, Vol. 40, pp. 110–112.

Stefatos, G. and Hamza, A. B. (2007). Statistical process control using kernel pca, *Proc. 15th Mediterr. Conf. Control Autom.*, IEEE, Anthens, Greece, pp. 1418–1423.

Stubbs, S., Zhang, J. and Morris, J. (2012). Fault detection in dynamic processes using a simplified monitoring-specific CVA state space modelling approach, *Comput. Chem. Eng.* **41**: 77–87.

Tan, S., Wang, F., Chang, Y., Chen, W. and Xu, J. (2010). Fault detection and diagnosis of nonlinear processes based on kernel ICA-KCCA, *Chinese Control Decis. Conf. (CCDC), 2010*, IEEE, Xuzhou, pp. 3869–3874.

Tien, D. X., Lim, K.-W. and Jun, L. (2004). Comparative Study of PCA Approaches in Process Monitoring and Fault Detection, *30th Annu. Conf. IEEE Ind. Electron. Soc.*, Busan, Korea, pp. 2594–2599.

Tsung, F. (2000). Statistical monitoring and diagnosis of automatic controlled processes using dynamic PCA, *Int. J. Prod. Res.* **38**(3): 625–637.

van Sprang, E. N., Ramaker, H.-J., Westerhuis, J. a., Gurden, S. P. and Smilde, A. K. (2002). Critical evaluation of approaches for on-line batch process monitoring, *Chem. Eng. Sci.* **57**(18): 3979–3991.

Vanhatalo, E. and Kulahci, M. (2015). The Effect of Autocorrelation on the Hotelling T2 Control Chart, *Qual. Reliab. Eng. Int.* **31**: 1779–1796.

Vapnik, V. N. (2000). *The Nature of Statistical Learning Theory*, Vol. 8, Springer-Verlag.

Venkatasubramanian, V., Rengaswamy, R. and Kavuri, S. N. (2003). A review of process fault detection and diagnosis: Part II: Qualitative models and search strategies, *Comput. Chem. Eng.* **27**(3): 313–326.

Venkatasubramanian, V., Rengaswamy, R., Kavuri, S. N. and Yin, K. (2003). A review of process fault detection and diagnosis Part III: Quantitative model based methods, *Comput. Chem. Eng.* **27**: 293–311.

Venkatasubramanian, V., Rengaswamy, R., Yin, K. and Kavuri, S. N. (2003). A review of process fault detection and diagnosis Part I: Quantitative model-based methods, *Comput. Chem. Eng.* **27**: 293–311.

Wang, Y., Cheng, G. and Tang, J. (2010). Statistical process monitoring of continuous catalytic reforming heat exchangers using canonical variate analysis, *29th Chinese Control Conf.*, Beijing, China, pp. 5072–5075.

Westerhuis, J. A., Gurden, S. P. and Smilde, A. K. (2000). Generalized contribution plots in statistical process monitoring, *Chemom. Intell. Lab. Syst.* **51**: 95–114.

Wilson, D. J. H. and Irwin, G. W. (2000). PLS modelling and fault detection on the Tennessee Eastman benchmark, *Int. J. Syst. Sci.* **31**(11): 1449–1457.

Wold, S. (1992). Nonlinear partial least squares modelling II. Spline inner relation, *Chemom. Intell. Lab. Syst.* **14**(1-3): 71–84.

Wold, S., Esbensen, K. and Geladi, P. (1987). Principal component analysis, *Chemom. Intell. Lab. Syst.* **2**(1-3): 37–52.

Xiong, C., Zhao, Y. and Liu, W. (2006). Fault detection method based on artificial immune system for complicated process, *Int. Conf. Intell. Comput.*, Kunming, China, pp. 625–630.

Xiong, L., Liang, J. and Qian, J. (2007). Multivariate statistical process monitoring of an industrial polypropylene catalyzer reactor with component analysis and kernel density estimation, *Chinese J. Eng.* **15**(4): 524–532.

Yang, J., Jin, Z., Yang, J. Y., Zhang, D. and Frangi, A. F. (2004). Essence of kernel Fisher discriminant: KPCA plus LDA, *Pattern Recognit.* **37**(10): 2097–2100.

Yang, Y., Chen, Y., Chen, X. and Liu, X. (2012). Multivariate industrial process monitoring based on the integration method of canonical variate analysis and independent component analysis, *Chemom. Intell. Lab. Syst.* **116**: 94–101.

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, *J. 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 Trans. Ind. Electron.* **61**(11): 6414–6428.

Yue, H. H. and Qin, S. J. (2001). Reconstruction-Based Fault Identification Using a Combined Index, *Ind. Eng. Chem. Res.* **40**(20): 4403–4414.